ADMINISTRACION

DE SISTEMAS

GNU/LINUX

GUIA DE ESTUDIO HACIA UNA CAPACITACION SEGURA

FUNDACION

Código Libre Dominicano

Antonio Perpinan

ADMINISTRACION
DE SISTEMAS
GNU/LINUX

GUIA DE ESTUDIO HACIA UNA CAPACITACION SEGURA

FUNDACION
Código Libre Dominicano

FUNDAMENTOS DE SISTEMA GNU/LINUX

GUIA DE AUTO ESTUDIO HACIA UNA CAPACITACION SEGURA

LOS PROPÓSITOS DEL CURSO

Los profesionales de la tecnología de la información (TI) son críticos hoy día para el ambiente de negocio. Adquirir las herramientas y conocimiento disponible en la tecnología de hoy es vital. GNU/Linux y el Código Libre y Abierto han colocado un nuevo estándar en lo que es desarrollo e implementación de aplicaciones nuevas y personalizables. GNU/Linux continúa ganando espacio de reconocimiento entre los profesionales y administradores del TI debido a su flexibilidad, estabilidad, y su poderosa funcionalidad. A medida que más empresas utilizan GNU/Linux, crece la necesidad de soporte y planificación sobre la integración de GNU/Linux en infraestructuras nuevas y/o existentes. El rol del administrador es guiar la implementación y desarrollo de soluciones basadas en GNU/Linux. Su éxito o derrota dependerán de su conocimiento y experiencia de esta fantástica arquitectura.

Este curso es un repaso comprensivo de las características y funcionalidad de GNU/Linux, orientada a preparar al estudiante con las herramientas necesaria para la certificación. Explicación detallada se provee de los conceptos claves, muchos conceptos y utilidades de GNU/Linux son idénticos sin importar la distribución específica siendo utilizada. Algunas características están disponibles en algunas distribuciones, y otras son añadidas durante la instalación. La naturaleza de GNU/Linux y el Software Open Source, es tal, que cambios al fuente y cambio a funcionalidad de cualquier componente debe ser incluido en la distribución específica . Los conceptos sublimes de las capacidades de GNU/Linux se mantienen consistentes a través de cada distribución, kernel y cambio de Software.

Estos libros han sido desarrollados de acuerdo con los estándares de la industria de la certificación de GNU/Linux. Los objetivos de la certificación GNU han sido elementos claves en el desarrollo de este material. La secuencia de los exámenes de certificación GNU/Linux provee la gama más amplia de los conceptos necesarios para dominar GNU/Linux. Los objetivos de las certificaciones LPI y RHCE también son incluidos. El CD interactivo y la página Web con el curso contiene videos digitales y pequeñas prácticas de selección múltiple igual a los del exámen. En el libro LA GUIA DEL ESTUDIANTE se provee una guía específica para la preparación de la certificación.

Este libro provee los conceptos y principios fundamentales necesarios para administrar un sistema GNU/Linux. Los conceptos y las tareas de administración pueden ser un poco amplios. Se le dará una explicación del rol del administrador, estructura y función detallada del kernel, y cubriremos tópicos administrativos claves del manejo de paquetes, procesos, espacio de disco, Backups y los usuarios así como las tareas programáticas, y los Logs/Registros del sistema. Este conjunto de herramientas te permitirán apropiadamente administrar un sistema GNU/Linux sea este de unos cuantos hasta miles de usuarios. Estos capítulos también te proveerán la información que necesitas para Certificarte.

Fundamentos de GNU/Linux proporciona una introducción a profundidad de los conceptos y de los principios que son necesarios para instalar un sistema GNU/Linux y desenvolverse en los ambientes de ventana del X y de la línea de comandos. Este manual da la dirección paso a paso para las distribuciones importantes de GNU/Linux y su instalación, incluyendo RedHat, Debian, Mandrake y Slackware. Se enfatizan los conceptos de instalación, las utilidades, y la funcionalidad de GNU/Linux común a todas las distribuciones y estas se explican en detalle adicional. Un principiante o un experto pueden aprender o repasar los conceptos de particionar discos y localizar los archivos de configuración, usando el shell y las consolas, crear los scripts, y editar archivos de texto que permanecen dominantes, sin importar la nuevas herramientas gráficas, para los ajustes de configuración. Este conjunto de tópicos permitirá que usted instale y configure correcta-

mente un sistema GNU/Linux. En estos capítulos también se le provee la información necesaria para certificar sus habilidades en GNU/Linux

METAS DEL CURSO

Este curso le proveerá la información necesaria para completar los siguientes tópicos:

- Describir los componentes estructurales y distinguir entre una distribución de GNU/Linux y otra.
- Describir software de fuente abierta (Software Open Source) y diferenciar entre GNU/GPL.
- Crear los disquetes de arranque de instalación.
- Instalar las principales distribuciones de GNU/Linux: RedHat (RPM), Debian (DPKG) y Slackware (tar.gz).
- Utilizar los ambientes de escritorio KDE y GNOME.
- Instalar y configurar XFree86.
- Localizar y utilizar la ayuda en línea.
- Configurar el hardware del sistema.
- El uso de fdisk o el cfdisk para crear, corregir, y suprimir particiones del disco.
- Utilizar el LILO/GRUB para manejar opciones para cargar el sistema.
- Arrancar el sistema, cambiar los runlevels, y cerrar o re-iniciar el sistema.
- Utilizar los disquetes de rescate para iniciar un sistema que se ha dañado.
- Describir el sistema de archivos jerárquico de GNU/Linux y el papel de los directorios y archivos claves en la organización del sistema. Trabajar con eficacia en la línea de comando de Linux usando comandos comunes del shell, streams, tuberías, filtros, y cambio de dirección.
- Usar scripts del shell para realizar tareas repetitivas rápidamente.
- Abrir, corregir, y almacenar documentos de texto usando el editor 'vi'.
- Manejar los sistemas de impresión locales.
- Describir algunas aplicaciones comunes disponibles al usuario para sus tareas, tales como: navegar en Internet y acceso a E-mail, procesamiento de textos, presentaciones, hojas de cálculo, y manejo de gráficos.

EJERCICIOS

Los ejercicios en este manual son diseñados para dar practicas reales en los ambientes de redes y aislados (stand-alone o networking) al usuario. Es altamente recomendado que usted complete todos los ejercicios en cada capítulo antes de continuar al próximo. Entendemos que en raros casos tal vez esto no sea conveniente cuando estudia fuera del taller. Si por alguna razón no puedes completar un ejercicio por circunstancias ajenas, debes planificar completarlo tan pronto sea posible.

Existirán ejercicios que no podrás completar por el limitante de equipo ó software. No permita que esto le impida completar los otros ejercicios que resten en el capítulo ó modulo.

TOME NOTA

Los ejercicios en este libro fueron diseñados para ser ejecutados en un equipo de prueba y nunca deben ser llevados a cabo en uno trabajando y donde se ejecuten aplicaciones importantes. Instalar GNU/Linux, reparticionar para instalar GNU/Linux, o practicando los ejercicios en una LAN u ordenador de trabajo puede causar problemas de configuración, lo cual puede conllevar a perdidas irreparable de data y dispositivos periféricos. Por favor siempre recuerde esta advertencia. Es preferible que dediques una estación de trabajo para practicar estos ejercicios. Instalar GNU/Linux en una situación dual-boot es una alternativa razonable, pero aún así conlleva ciertos riesgos.

WEB y CD

Una parte muy clave de esta serie de auto-aprendizaje es el portal de soporte. Las lecciones que le indiquen visitar la página web o el CD-ROM que le acompaña, a menudo, es para ayuda con los conceptos que son mejor entendidos después de una descripción visual. Los segmentos video Digital proporcionan una ilustración gráfica acompañada por una narración de los instructores. Estas lecciones son ideales, ambos como introducciones para afinar conceptos y para ayudar el refuerzo.

RECUERDE

Como herramienta de soporte les ofrecemos el CD interactivo, incluido en este libro, y nuestra página web http://www.abiertos.org y allí accesar hacia la sección Linux-Certificación, estos contienen exámenes de prueba de las diferentes certificaciones. Recreamos el escenario de las preguntas de selección múltiples, multi-selección y falso verdadero. Es muy importante que tome muchas horas de practicas antes de intentar pasar el exámen de certificación que le corresponda ya sea LPI ó RHCE.

CONTENIIDO

CAPITULO 1

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

INTRODUCCIÓN A LA ADMINISTRACIÓN

OBJETIVOS

Al completar este capítulo, usted podrá:

- Detallar los permisos necesarios para usar el comando su
- Definir el uso de comando sudo en relación a la seguridad del sistema.
- Identificar recursos claves del sistema
- Proveer soporte al usuario
- Localizar y reposicionar archivos del sistema.

Preguntas Pre-Exámen

1.- ¿Qué tipo de acceso le permite el comando sudo a los usuarios?

2.- ¿Cuál argumento de opción se le pasa al comando uname para ver el tipo de procesador?

INTRODUCCION

En este capítulo analizamos el Rol del Administrador del Sistema de GNU/Linux. Como el administrador del sistema a menudo usted necesitará accesar la cuenta del superusuario, root. Es de absoluta necesidad que entienda la importancia del poder involucrado al utilizar la cuenta de root, porque si es utilizada inadecuadamente puede ser desastroso. El rol del superusuario será utilizado en las operaciones del día a día.

Rol del Administrador del Sistema

Esta sección se concentrará en el rol del administrador del sistema GNU/Linux. Una descripción del rol del administrador del sistema también es incluida. El Administrador del Sistema es quien vela por el buen estado del sistema. Esto incluye desde las operaciones normales del día a día, a cosas como backups, agregar y remover usuarios, instalar, remover y configurar aplicaciones. En sistemas grandes de multiusuarios, las tareas administrativas son diferenciadas entre las que se realizan diariamente y las menos frecuentes. Los operadores del sistema son involucrados en tareas comunes de administración, con supervisión del administrador; mientras que el Administrador lleva a cabo las de mayor nivel de seguridad y las menos frecuentes.

Tareas más complejas, como instalación de sistemas y actualizaciones, muchas veces son clasificadas como tarea de soporte técnico. Aunque en compañías pequeñas, se espera que el administrador del sistema lleve acabo estas rutinas.

El administrador del sistema es responsable de asegurar que el sistema GNU/Linux brinde los servicios necesarios para que los usuarios puedan cumplir con sus tareas. Esto involucra una serie de actividades donde las siguientes son de mayor importancia:

- Agregar nuevos usuarios al sistema y configurar sus directorios home y los privilegios básicos.
- Instalar software nuevo, incluyendo aplicaciones, nuevas versiones del sistema y corregir errores.
- Monitorear el uso del sistema de archivos, asegurándose que nadie esta mal utilizando los recursos y asegurarse que las políticas de seguridad y backups están siendo implementadas.
- Responder a problemas enfrentados por los usuarios y dar seguimiento a los casos de errores reportados, mantenerse informado de cuestiones informáticas que le competen en su total desempeño.
- Instalar nuevos componentes de hardware con sus módulos respectivos.
- Asegurarse de que los servicios básicos para la operación de la empresa estén disponibles, tales como email, acceso remoto y servicios de redes internas o intranet, etc.

Los siguientes tópicos son discutidos en esta sección:

- Responsabilidad General
- Soporte de Usuarios
- El Uso de la Cuenta root
- Shells de Administración
- Identificar el Sistema GNU/Linux
- Identificar los Usuarios Activos
- Sistema de Archivos GNU/Linux
- Los Archivos Por Defecto del Sistema

Responsabilidad General

Identifique su rol como administrador del sistema. Pregunte a su superior que espera de usted y notifique a los usuarios de su responsabilidad y sus privilegios. Aclare cualquier área en la cual usted no está

totalmente claro si le compete. Investigue si puedes delegar algunas tareas al equipo administrativo. Tareas como cambiar el toner de las impresoras o añadir resmas de papel no son tan difíciles como entender y configurar sistemas de archivos GNU/Linux.

Usted debe tener un agenda con fechas para servicios de mantenimiento y reposición de equipos, si la compañía tiene departamento de hardware, o una compañía externa para ese servicio, por citar un caso, cada 2 meses para de esta forma dar mantenimiento preventivo y eficaz. Compañías grandes (de más de 100 terminales) pueden efectuar contratos de mantenimiento cuales incluyen desde la reposición de equipos hasta que éste sea reparado.

Listamos algunas cosas adicionales que debe identificar en el rol de administrador del sistemas:

- ¿Quién está a cargo del soporte de hardware?
- ¿Quién usa el sistema?
- ¿Habrá una actualización del sistema pronto?
- ¿Puedo como administrador pedir más hardware?
- ¿Cómo puedo conseguir recursos adicionales?

Soporte Para los Usuarios

Existen muchas instalaciones en las cuales un usuario necesita contactar al Administrador del Sistema. El e-mail es la forma más común de intercomunicación entre los usuarios y el Administrador, pero esto es muy lento, si se desea respuestas inmediatas de los usuarios. En este caso podemos emplear el comando write para mandar un mensaje a un administrador que se encuentra en línea en el sistema. Este comando envía un mensaje sencillo a otros usuarios del sistema. La disponibilidad de los usuarios para recibir mensaje esta controlado por el comando mesg. Una vez el mensaje es recibido el administrador puede responderlo de la misma forma con el comando write. Si una sesión mucho más interactiva es requerida se puede crear una sesión de chat con el comando talk, esta función permite a dos o más usuarios del sistema comunicarse en tiempo real. Los usuarios pueden impedir el despliegue de los mensajes con el comando mesg n. A menudo los administradores colocan este comando en el archivo perfil del usuario (/etc/profile o /home/usuario/.bash_profile). Si un administrador desea enviar un mensaje a todos los usuarios del sistema actualmente conectados, puede utilizar el comando wall (write all).

$ write usuario-tal
Sabes dónde esta el CD de instalación de OpenOffice?
ctrl+D

EJERCICIO 1-1: Navegar y Usar el Shell del Administrador

No se provee repuesta para este ejercicio.

1.- Ingrese como root y ejecute algunas de las aplicaciones y utilidades gráficas orientadas a la administración de GNU/Linux. Si estas en SuSE (YaST) o una distro de las anteriores, ejemplo RedHat 7.1 o Mandrake 8.1, Debian 2.2 podrías experimentar con Linuxconf. Este tipo de aplicaciones se utilizó mucho en GNU/Linux pero con el tiempo fueron abandonadas, ya que están en contra de las filosofías UNiX de no concentrar demasiado poder detrás de una sola aplicación. Ahora SuSE mantiene el YaST y Mandriva el DRAKNONF. En RedHat la funcionalidad existe pero las utilidades son individuales.

El Uso de la Cuenta de ROOT

Lograr que un sistema GNU/Linux opere a perfección es un proceso gradual. Toma tiempo ingresar los usuarios con toda su información, instalar las aplicaciones necesarias, configurar el servicio de redes, etc. Ingresar al sistema como root para ejecutar tareas simples puede ser el más simple de los errores y el más catastrófico. Por esto no es sorpresivo perder todo o parte de la funcionalidad del sistema al utilizar la cuenta root, esto es un dolor de cabeza para cualquier administrador. Veamos este ejemplo: ¿Qué pasaría si ejecutas este comando como el superusuario?

 # rm * .bak
 # rm -rf / home/miguel/tmp

En el caso # rm * .bak, hemos dejado un espacio después del asterico y esto borra todos los archivos mas el .bak y en el caso de # rm -rf / home/miguel/tmp, dejamos un espacio entre la barra y home/miguel, lo cual borra todos los archivos en barra(/) y el home/miguel/tmp, es decir el contenido inadvertido de espacio en blanco arroja resultados no esperados al borrar todos los archivos del disco.

Más aún borrar archivos en GNU/Linux accidentalmente es complicado por su diseño. Como cualquier otro sistema operativo basado en UNiX, GNU/Linux emplea un sistema de archivos indexado con acceso directo a la lista de bloques de memoria libre. Cuando un archivo se borra, su contenido forma los bloques del próximo archivo a crearse, todo esto en forma completamente democrática, el que primero venga se le servirá primero. Pues entonces una vez un archivo ha sido eliminado, un nuevo archivo es creado, el bloque viejo ha sido rehusado.

Aún otra inmensa consecuencia de trabajar como root tiene que ver con la creación de archivos. Estos archivos tendrán acceso restringido, situación creada porque los creo como administrador. Al crear estos archivos con estas restricciones tendrás por obligación que entrar al sistema como root.

Convención para Ejecutar Tareas del Superusuario

El administrador rápidamente puede cambiar de usuario a superusuario con el uso del comando su, puede cambiar la cuenta del usuario efectivamente a user ID cero “0” permitiendo acceso a archivos y servicios. Otra variante del comando su es:

 # su -cuenta-de-usuario

El menos “-” significa que va a cambiar de la cuenta del superusuario a la cuenta del usuario, y que ademas desea cargar sus variables de ambiente, lo cual es similar a haber ingresado como ese usuario. Para retornar a la cuenta del superusuario, simplemente log out del shell del usuario. Para retornar a la cuenta original salga de la cuenta del superusuario.

Se pueden colocar restricciones en el uso del comando su a un grupo cerrado de cuentas, esto se logra asignando permisos al comando su y asociando esos permisos a un grupo, por ejemplo al grupo admin.

Otra manera de incrementar la seguridad de los programas utilizados por root es usar la utilidad sudo. Esto es algo similar a restringir el comando su a ciertos grupos pero va un poco más lejos y se pueden restringir otros comandos. En el archivo /etc/sudoers se determina quien puede usar el comando sudo.

Shells de Administración

Los shells de administración han sido introducidos para que GNU/Linux sea más aceptable al mercado comercial. Los shells simplifican la administración de la mayoría de las funciones relacionadas para que usuarios novatos puedan administrar sistemas GNU/Linux.

Por lo general, el shell se configura para proveer 2 tipos de COMMAND PROMPTs. El símbolo de $

representa a los usuarios sin privilegios (con un valor positivo) UIDs. El símbolo de # representa el estatus del superusuario y un valor de UID = 0 que permite al manejador de archivos ignorar el esquema de seguridad. La cuenta de superusuario es la única cuenta que puede requerir algunos servicios, por ejemplo, cambiar la fecha y hora, agregar nuevos usuarios e incrementar las prioridades de los procesos.

La variable PAGER es utilizada para determinar cual programa de pantalla se usa para filtrar la salida. El por defecto puede ser less o more, aunque less es probablemente el más utilizado por que incluye mayor funcionabilidad.

```
$ PAGER=less
$ export PAGER
```

Otra variable que debe identificar si está exportada correctamente es la variable MANPATH, la cual es usada por man para identificar su directorio, que por lo general es, /usr/man. Si está vigente entonces man buscará en el directorio especificado por la variable.

Identificar un Sistema GNU/Linux

El comando uname despliega al administrador del sistema, información sobre el equipo y el sistema operativo. Las siguientes opciones pueden ser utilizada por el comando uname:

-a, --all	muestra toda la información
-m, --machine	muestra el tipo de máquina (hardware)
-n, --nodename	muestra el nombre de `host' del nodo de red de la máquina
-r, --release	muestra la distribución del sistema operativo
-s, --sysname	muestra el nombre del sistema operativo
-p, --processor	muestra el tipo de procesador
-v	muestra la versión del sistema operativo
--help	muestra esta ayuda y finaliza
--version	informa de la versión y finaliza

```
[root@www /root]# uname -r
2.2.19-7.0.8
[root@www /root]# uname -a
Linux www 2.2.19-7.0.8 #1 Thu Jun 21 06:28:56 EDT 2001 i686 unknown
[root@www /root]# hostname
www
[root@www /root]#
```

El comando hostname es en realidad un alias a uname -n y puede ser que no esté disponible en todas las versiones de GNU/Linux.

Identificar los Usuarios Activos

La familia de los comandos who retorna la identificación original de los usuarios durante el proceso de inicio de sección. La información desplegada del comando who es mantenida en /var/run/utmp. Un historial de cada login es también mantenido en /var/log/wtmp. Si el usuario subsecuentemente cambia su identidad con el comando su, el comando who le retorna el nombre original. El comando who también puede identificar el usuario actual, así como lo hace el comando id. Algunos sistema también tienen un comando w originalmente de BSD el cual es muy parecido al comando who.

```
[root@www /root]# who
root pts/0 Jul 29 01:13
admin. Pts/1jul 29 01:13
```

```
$whoami
carlos                console jul 25 11:31

[root@www /root]# id
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),10(wheel)
```

El comando who muestra los usuarios actuales bajo la cuenta que operan después del cambio:

```
[root@www /root]# who
root pts/0 Jul 5 00:13
root pts/1 Jul 5 00:52
root pts/0 Jul 5 03:17
root pts/0 Jul 14 09:20
root pts/0 Jul 15 22:02
root pts/0 Jul 16 20:24
root pts/0 Jul 16 20:58
root pts/0 Jul 16 21:23
admin pts/0 Jul 18 18:44
root pts/0 Jul 18 18:46

$ su - lp
$ who
root console jul 17 23:35
id
uid=7(lp) gid=9(lp) groups=9(lp)
switche to use lp: who shows the name I logged with: id shows the current
$w
09:47     up      10:19   2 users         load            average 0:00    0:00    0:00
user      tty     login@  idle            JCPU            PCPU            what
root      lft0    11:33   10:00           13              11              xinit
root      pts/0   11:46   0               24              0               w
```

El Sistema de Archivos

Una de las grandes diferencias de GNU/Linux y otros sistemas operativos es que todo es considerado un archivo. El kernel es un conjunto de archivos, las librerías son archivos, el directorio es un archivo, el disco duro es un archivo. Como los dispositivos de almacenamientos son vistos como un archivo no necesita letras para identificarlos, esto permite una gran flexibilidad y modularidad, permitiendo incluir discos y archivos desde otros sistemas. Este método de direccionar todo dentro de la jerarquía de sistema de archivos es similar a las versiones de UNiX comerciales que manejan directorios, memoria y discos. Con algunas excepciones, aprender a navegar dentro del sistema de archivos GNU/Linux prepara al usuario para usar un sistema UNiX.

Antes de que un sistema de archivos sea accesado en GNU/Linux, debe ser montado en la estructura del sistema de archivos. Se usa un directorio como punto de montaje de estos discos y particiones para montar sus sistemas de archivos. Dentro de la jerarquía del sistema de archivos pueden ser mapeados diferentes discos o particiones usando el comando mount.

La jerarquía del sistema de archivos GNU/Linux se asemeja a la estructura de la raíz de un árbol. Empieza con un directorio singular (inteligentemente nombrado root o directorio raíz) y se ramifica. Todos los archivos en un sistema GNU/Linux son accesados a través de la estructura del sistema de archivos. Las siguientes páginas le introducirán al estándar del sistema de archivos GNU/Linux. Discutiremos la creacion de particiones en los siguientes capitulos.

El sistema de archivos UNiX está caracterizado por:

- Una estructura jerárquica.
- Un tratamiento consistente de la información de los archivos.
- Protección de los archivos.

Este estándar del sistema de archivos GNU/Linux sigue el mismo principio básico que la mayoría de los sistemas de archivos UNiX. Sin embargo, este estándar no intenta concordar en cada aspecto posible con alguna implementación particular del sistema UNiX. De cualquier forma, muchos de los aspectos de este estándar están basados en ideas encontradas en UNiX y sistemas similares a UNiX.

Es posible después de cuidadosa consideración definir dos categorizaciones ortogonales de archivos: Compartibles vs. No-Compartibles y Variables vs. Estáticos, entre los factores se incluyen:

- Prácticas comunes en la comunidad Linux.
- La implementación de otras estructuras de sistemas de archivos.
- Los estándares aplicables.

La información compartible es aquella que puede ser compartida entre varias máquinas diferentes; la no compartible es aquella que debe ser local a una máquina particular. Por ejemplo, los directorios hogar de los usuarios son compartibles pero los archivos de bloqueo de dispositivo (lock files) son no compartibles.

La información estática incluye binarios, librerías, documentación y todo aquello que no cambia sin la intervención del administrador del sistema. La información variable es todo lo que cambia sin la intervención del administrador.

Conocer estos principios básicos le ayudará a entender la estructura de cualquier sistema de archivos bien planificado.

La distinción entre información compartible y no compartible es necesaria por varias razones:

- En un ambiente de red (pe. más de un host en un site), existe una buena cantidad de información que se puede compartir entre diferentes máquinas para ahorrar espacio y facilitar la tarea de administración.
- En un ambiente de red, ciertos archivos contienen información específica a una sola máquina, por tanto, estos sistemas de archivos no pueden ser compartidos (sin tomar medidas especiales).
- Las implementaciones de facto del sistema de archivos no permiten que la jerarquía /usr fuera montada sólo-lectura, porque contien archivos y directorios que necesitaban ser escritos muy frecuentemente. Éste es un factor que debe atacarse cuando algunas partes de /usr se comparten en una red, o se montan sólo-lectura debido a otras consideraciones tales como la seguridad.

La distinción “compartible” puede ser usada para soportar, por ejemplo:

- Una partición /usr (o componentes de /usr) montada (sólo-lectura) a través de la red (usando NFS).
- Una partición /usr (o componentes de /usr) montada desde medios de sólo-lectura. Un CD-ROM puede ser considerado como un sistema de archivos sólo-lectura compartido con otros sistemas GNU/Linux utilizando el sistema de correo como una red.

La distinción “estática” contra “variable” afecta el sistema de archivos de dos maneras principales:

- Dado que / contiene ambos tipos de información, variable y estática necesita montarse lectura-escritura.
- Dado que el /usr tradicional contiene ambos tipos de información variable y estática y dado que podrí mos desear montarlo sólo-lectura (vea arriba), es necesario proporcionar un método para hacer que /usr se monte sólo-lectura. Esto se logra con la creación de una jerarquía /var que se monta lectura-escritura (o es parte de una partición lectura-escritura tal como /), que toma mucho de la funcionalidad tradicional de la partición /usr.

El Directorio Raíz

/	Directorio raíz, donde todo empieza
bin	Binarios de comandos esenciales
boot	Archivos estáticos de cargador de arranque(boot-loader)
dev	Archivos de dispositivos
etc	Configuración del sistema local-máquina
home	Directorios home de los usuarios
lib	Librerías compartidas
lost+found	Directorio para almacenar archivos a recuperar
mnt	Punto de montaje de particiones temporales
opt	Para colocar software que no fue incluida en el sistema operativo
root	Directorio hogar del usuario root
sbin	Binarios del sistema esenciales
tmp	Archivos temporales
usr	Segunda jerarquía mayor
var	Información variable

Cada directorio listado será discutido en detalle en una sección más delante. /usr y /var, cada uno tiene su propia sección en este documento.

El kernel de GNU/Linux estaría localizado en (/) o en (/boot). Si está localizado en / recomendamos usar el nombre VMLINUX o VMLINUZ, nombres que han sido usados en paquetes fuentes del kernel de Linux. Más información de la localización del kernel se puede encontrar en la sección acerca de (/), más adelante.

(/) : Directorio Raíz

La barra es el directorio raíz, el principio del sistema de archivos. Sólo existe un directorio root y todos los archivos descienden o están colgando de el. Este directorio contiene todos los archivos y programas necesarios.

/bin: Directorio de los Binarios

Binarios de comandos esenciales de los usuarios (disponibles para todos los usuarios). El /bin contiene comandos que pueden ser utilizados por los usuarios y el administrador del sistema, pero que son requeridos en el modo mono-usuario (single-user mode) puede también contener comandos que son utilizados indirectamente por algunos scripts.

Todos los binarios utilizables sólo por root, tales como daemons, init, getty, update, etc. Están localizados en /sbin o /usr/sbin, dependiendo si son o no esenciales. Una regla muy importante es que no habrá subdirectorios dentro de /bin.

Los binarios de los comandos que no son esenciales para estar en /bin estarán localizados en /usr/bin, los elementos que son utilizados por usuarios solamente (pero no por root) mail, chsh, entre otros que no son esenciales para estar dentro de la partición.

Los siguientes comandos han sido incluidos porque son esenciales. Algunos están presentes debido a que tradicionalmente han estado en /bin:

arch, cat, chgrp, chmod, chown, cp, date, dd, df, dmesg, echo, ed, false, kill, ln, login, mkdir, mknod, more, mount, mv, ps, pwd, rm, rmdir, sed, setserial, sh, stty, su, sync, true, umount, uname.

Si /bin/sh es Bash, entonces /bin/sh sería un enlace simbólico o duro a /bin/bash dado que bash se com-

porta diferente cuando es llamado como sh 0 bash. La pdksh que puede ser la /bin/sh en los discos de instalación y sería igual que /bin/sh como un enlace simbólico a /bin/ksh. El uso de enlaces simbólicos en estos casos permite que los usuarios vean fácilmente que /bin/sh no es una shell estilo bourne.

Dado que la localización estándar por defecto del shell estilo 'c' es /bin/csh, si y sólo si está disponible en el sistema una shell estilo c o equivalente (tal como /bin/tcsh), estaría disponible con el nombre /bin/csh. El /bin/csh puede ser un enlace simbólico a /bin/tcsh 0 /usr/bin/tcsh.

Los comandos eval y test son comandos internos en bash, pdksh, zsh y las shell korn recientes, en cada reemplazo de las shell tipo bourne que hay para Linux. Estos comandos estarían localizados dentro de /usr/bin (Se deben incluir como binarios separados con cualquier sistema Linux que intente cumplir con el estándar POSIX).

El comando bin/arch produciría el mismo resultado que uname -m, específicamente: 386 0 486 para sistemas intel y compatibles.

Comandos para Restaurar

Estos comandos se han incluido para hacer posible restaurar el sistema (siempre que / este intacto). tar, gzip, gunzip, zcat. Si se hacen respaldos de sistemas utilizando otros programas, entonces la partición / contendrá los componentes mínimos necesarios. Por ejemplo, muchos sistemas incluirían cpio como la segunda utilería más usada para respaldos después de tar. Pero si jamás se espera restaurar el sistema desde la partición /, entonces estos binarios se pueden omitir (i.e., montar / en chip ROM, montar /usr desde NFS). Si la restauración del sistema se planea a través de la red, Entonces FTP 0 TFTP (junto con todo lo necesario para obtener una conexión FTP) estarían disponibles en la partición /.

Los comandos de restauración pueden aparecer en, /bin 0 /usr/bin en sistemas diferentes GNU/Linux.

Comandos de Red

Estos son únicamente los binarios de red que los usuarios y root querrán o necesitarán ejecutar que no sean los que están en /usr/bin 0 /usr/local/bin: domainname, hostname, netstat, ping.

/boot: Archivos estáticos del cargador de arranque (boot loader)

Este directorio contiene todo para arrancar excepto los archivos de configuración y el instalador de mapas. En sentido más sencillo /boot es el lugar donde debe colocar todo lo que debe utilizar antes de que el kernel ejecute /sbin/init. Esto incluye sectores maestros de arranque (master boot sectors) guardados, archivos de mapeo de sectores y cualquier otra cosa que no es editada directamente a mano. Los programas necesarios para arreglar que el cargador de arranque sea capaz de arrancar un archivo (tal como el instalador de mapas [lilo]) estarán localizados en /sbin. Los archivos de configuración para cargadores de arranque podrían estar localizados en /etc.

Como se expuso arriba, el kernel de Linux puede estar localizado en / 0 en /boot, si se localiza en /boot, recomendamos que se le dé un nombre más descriptivo.

/dev: Archivos de dispositivos

Este es el directorio de los dispositivos. Contendría un archivo por cada dispositivo que el kernel de Linux pueda soportar. Este directorio dev también contiene un script llamado MAKEDEV el cual puede crear dispositivos cuando se necesiten. Puede contener un MAKEDEV local para dispositivos de uso sólo-local.

MAKEDEV debe hacer previsión para crear cualquier archivo de dispositivo especial listado en la lista

de números mayores/menores, no sólo aquellos de una distribución particular.

Los enlaces simbólicos no se deben distribuir en sistemas Linux, sino sólo como se prevé en la lista de dispositivos de Linux. Esto es porque las instalaciones locales seguro diferirán de aquellas de la máquina del desarrollador. Además si un script de instalación configura enlaces simbólicos en la instalación, estos enlaces seguramente no se actualizarán si se hacen cambios locales en el hardware. Cuando se usan responsablemente, como sea, son de buen uso.

Este documento incorpora como referencia la lista de dispositivos de Linux, mantenida por: Peter.Anvin@linux.org: El encargado de los dispositivos Linux. Todos los archivos especiales de dispositivo seguirán el estándar en ese documento, que está disponible en ftp.yggdrasil.com en /pub/device-list.

/etc : Configuración del Sistema Local a la Máquina

El directorio /etc contiene archivos y directorios que son locales al sistema actual. Ningún binario debe estar directamente dentro de /etc. Los binarios que en el pasado se encontraban en /etc, están en /sbin 0 /usr/sbin. Esto incluye archivos tales como init, getty y update. Los binarios tales como hostname que son utilizados por usuarios ordinarios y por root no irían en /sbin sino en /bin.

/etc/skel: Esqueletos de Configuración de Usuarios

El directorio /etc/skel es la localidad para los llamados archivos esqueletos de los usuarios, que le son dados por defecto cuando un nuevo usuario recibe una cuenta, este directorio puede contener subdirectorios para diferentes grupos de usuarios (i.e./etc/skel/apoyo, /etc/skel/usuarios).

/etc/X11: Archivos de Configuración para el X11

El directorio etc/X11 es el lugar recomendado para todos los archivos de configuración de X11 locales a la máquina. Este directorio es necesario para permitir el control local si /usr se monta sólo-lectura. Los archivos que deben ir en este directorio incluyen Xconfig (y/o XF86Config) y Xmodmap.

Los subdirectorios de /etc/X11 pueden incluir aquellos para xdm y para cualesquier otros programas (como algunos manejadores de ventanas por ejemplo) que lo necesiten. Recomendamos que los manejadores de ventanas con un solo archivo de configuración que es un archivo.*wmrc por defecto, que lo llamen system.*wmrc (a menos que exista una alternativa ampliamente aceptada) y que no utilice un subdirectorio. Cualquier subdirectorio de un manejador de ventanas se llamaría idéntico al binario del manejador de ventanas.

El etc/X11/xdm retiene los archivos de configuración de xdm. Esto es la mayoría de los archivos normalmente hallados en /usr/lib/X11/xdm; Vea la sección 5, /var/lib/xdm, para mayor información.

La siguiente sección intenta parcialmente examinar la descripción del contenido de /etc con algunos ejemplos. Esta no es una lista exhaustiva.

Archivos requeridos en /etc:
Archivos generales:

Estos archivos son necesarios en la mayoría de los sistemas GNU/Linux.
adjtime, csh.login, disktab, fdprm, fstab, gettydefs, group, inittab, issue, ld.so.conf, lilo.conf, magic, motd, mtab, mtools.conf, passwd, profile, psdatabase, securetty, shells, syslog.conf, tercamp, ttytype

Archivos de Red:

Estos archivos estarían instalados en la mayoría de los sistemas GNU/Linux.

exports, ftpusers, gateways, hosts, host.conf, host.equiv, host.lpd, inetd.conf, networks, printcap, protocols, resolv.conf.rpc, services

Hay dos modelos para la instalación de los scripts de comandos "rc" los cuales son invocados por init(8) al momento de arrancar, el modelo /etc/rc.d/* estilo SystemV. Cualquiera puede ser utilizado o una mezcla de los dos.

Los sistemas con la suite de passwords sombreadas (shadow password) tendrán archivos de configuración adicionales, en /etc (/etc/shadow y otros) y /usr/bin (useradd, usermod y otros).

/home: Directorios Hogar de los Usuarios (opcional)

El home es un concepto algo estándar, pero es claramente un sistema de archivos específico de un site. El arreglo diferirá de máquina a máquina. Esta sección describe una localización sugerida para los directorios hogar /de los usuarios, aun así, recomendamos que todas las distribuciones GNU/Linux usen este lugar como la localización por defecto de los directorios hogar.

En sistemas pequeños, cada directorio de usuario es uno de los subdirectorios debajo de /home, Ej. /home/smith, /home/torvalds, /home/operador, etc.

En sistemas grandes (especialmente cuando los directorios /home son compartidos entre varias máquinas usando NFS) es útil subdividir los directorios hogar. La subdivisión puede ser llevada a cabo utilizando subdirectorios tales como /home/apoyo, /home/huéspedes, /home/estudiantes, etc.

Muchas personas prefieren poner las cuentas de los usuarios en una variedad de lugares. Por tanto, ningún programa deberá confiar en esta localización. Si usted desea encontrar el directorio hogar de cualquier usuario, debería usar la función de librería getpwent(3) en vez de contar con /etc/passwd, por que la información puede estar almacenada remotamente usando usando sistemas como NIS.

/lib: Librerías Compartidas y Módulos Esenciales del Kernel

El directorio /lib contiene aquellas imágenes de las librerías compartidas que se necesitan para arrancar el sistema y ejecutar los comandos en el sistema de archivos raíz.

lib-	Librerías compartidas y módulos de kernel esenciales.
modules-	Módulos de kernel cargables.

Esto incluye /lib/libc.so.*, /lib/libm.so.*, el enlazador dinámico compartido /lib/ld.so.*, y otras librerías compartidas requeridas por binarios en /bin y /sbin.

Las librerías que son necesitadas sólo por los binarios en /usr (como cualquier binario de X Window) no pertenecen a /lib. Sólo las librerías compartidas requeridas para ejecutar los binarios dentro de /bin y /sbin deben estar aquí. La librería libm.so.* podría estar localizada en /usr/lib si no es requerida por nada en /bin 0 /sbin.

Por razones de compatibilidad, /lib/cpp necesita existir como una referencia al pre-procesador C instalado en el sistema. La localización usual del binario es /usr/lib/gcc-lib/<target>/<versión>/cpp. Puede existir un enlace/lib/cpp apuntando a este binario o a cualquier otra referencia a este binario que exista en el sistema de archivos. (Por ejemplo, /usr/bin/cpp se usa frecuentemente). La especificación para /lib/modules está aún por aparecer.

/lost+found

Algunas veces cuando el sistema experimente apagados inesperados esto causara que cese la operación antes de que el sistema escriba toda la data desde la memoria al disco. Esto usualmente causara perdida de data y/o la corrupción de algunos archivos en el sistema. Cuando este sistema se reinicia Linux hará el intento de salvar la data corrompido al directorio /lost+found en orden de tratar de provee al usuario una manera de recuperarla.

/mnt y /media: Punto de Montaje para Sistemas de Archivos Montados Temporalmente

Estos directorios se proveen para que el administrador pueda montar temporalmente sistemas de archivos cuando lo necesite. El contenido de este directorio es un asunto local y no debe afectar la manera en la cual se ejecuta ningún programa.

Se recomienda utilizar el directorio /media para los sistemas de archivos en medios removibles como cdroms, dvds, etc y /mnt para sistemas de archivos en medios permanente.

/opt:

Este directorio es de los tiempos de UNiX y era usando primariamente para programas que no venían en la distribución original en Linux era utilizado primariamente para los archivos del K Desktop Environment (KDE).

/proc: Sistema de Archivos Virtual de Información de Procesos y del Kernel

El sistema de archivos proc se está convirtiendo en el estándar de facto para el manejo de información de procesos y de sistema en vez de /dev/kmem y otros métodos similares. Recomendamos fuertemente esto para el almacenamiento y obtención de información de procesos así como otra información del kernel y de memoria.

/root: Directorio hogar de root (opcional)

El directorio / es tradicionalmente el directorio hogar del usuario root en los sistemas UNIX. /root se usa en muchos sistemas GNU/Linux y en algunos sistemas UNIX. El directorio hogar de la cuenta de el usuario root puede ser determinada por el desarrollador o por preferencias locales. Las posibilidades obvias incluyen **/**, **/root** y **/home/root**.

Si el directorio hogar de root no está almacenado en la partición raíz, será necesario asegurarse que tome / por defecto si no puede ser localizado.

> *NOTA: Recomendamos severamente contra el uso de la cuenta root para tareas cotidianas tales como leer el correo y ver las noticias (mail & news) sino que se use solamente para la administración del sistema. Por esta razón recomendamos que no aparezcan subdirectorios como Mail y News en el directorio hogar de la cuenta del usuario root. Recomendamos que el Mail para root y postmaster sean redirigidos a un usuario más adecuado.*

/sbin: Binarios del Sistema (Alguna vez mantenidos en /etc)

Los útiles usados por la administración del sistema (y otros comandos que sólo root utiliza) están almacenados en /sbin, /usr/sbin y /usr/local/sbin. /sbin típicamente contiene binarios esenciales para arrancar el sistema además de los binarios en /bin. Cualquier cosa que se ejecuta después de que se sabe que /usr se ha montado (cuando no hay problemas) debería estar en /usr/sbin. Los binarios de administración de sistema sólo-locales deben estar localizados en /usr/local/sbin.

Decidir que cosa va en los directorios de /sbin es sencillo: Si un usuario necesitará ejecutarlo, debe de ir en otro lado. Si sólo será ejecutado por el administrador del sistema o por root como scripts de administración, entonces debe ir en /sbin (o en /usr/sbin o en /usr/local/sbin, si el archivo no es vital para la operación del sistema).

Archivos como chfn que los usuarios usan sólo ocasionalmente deben aun estar en /usr/bin. El ping aunque es absolutamente necesario para el root (recuperación de la red y diagnóstico) es también frecuentemente usado por los usuarios y por esa razón debe ir en /bin. Los usuarios ordinarios no tendrán que poner ninguno de los directorios sbin en su búsqueda (path).

Recomendamos que los usuarios tengan permisos de lectura y ejecución en todo lo que se encuentra en /sbin excepto Talvez ciertos programas; setuid y setgid. La división entre /sbin y /bin no fue creada por motivos de seguridad o para evitar que los usuarios vieran el sistema operativo, sino para proveer una buena partición entre binarios que todos usan y los que se usan, principalmente las tareas de administración. No hay ganancia inherente en seguridad en hacer que /sbin este fuera del alcance de los usuarios.

Archivos requeridos en /sbin:

Comandos Generales
clock, getty, init, update, mkswap, swapon, swapoff, telinit.

Comandos de Apagado
fastboot, fasthalt, halt, reboot, shutdown.

Comandos de manejo de sistemas de archivos
fdisk, fsck, fsck.*, mkfs, mkfs.*

*donde * = uno de los siguientes: ext, ext2 minix, msdos, xia y Talvez otros.*

Comandos del sistema de archivos ext2, ext3 (opcional)
badblocks, dumpe2fs, e2fsck, mke2fs, mklost+found, tune2fs.

Instalador del mapa del cargador de arranque
Lilo, Grub

Comandos de Red
arp, ifconfig, route.

Archivos opcionales en /sbin:

Binarios estáticos. (Compilados estáticamente)

El ln estático sln y sync estático ssync son útiles cuando las cosas salen mal. El principal uso de sln (reparar enlaces simbólicos incorrectos en /lib después de una actualización mal orquestada) ya no es de preocupación mayor ahora que existe el programa ldconfig (usualmente localizado en /usr/sbin) y puede actuar como una mano guiadora al actualizar las librerías dinámicas. El sync estático es útil en algunas ocasiones de emergencia. Note que estas no necesitan ser versiones compiladas estáticamente de los ln y sync estándares, pero pueden ser.

El binario ldconfig es opcional en /sbin, dado que un servidor puede escoger ejecutar ldconfig al arrancar, en vez de sólo cuando se actualizan las librerías compartidas. (No está claro si es o no ventajoso ejecutar ldconfig en cada arranque). Aun así, a algunos les gusta tener ldconfig a mano para las siguientes (muy comunes) situaciones:

Se acaba de remover /lib/<archivo>

No se puede encontrar el nombre de la librería porque ls está enlazado dinámicamente. Se está usando una shell que no tiene ls inter-construida y no se sabe como usar “echo * “ como reemplazo. Se tiene un sln, pero no se sabe como nombrar al enlace.

ldconfig, sln, ssync.

Misceláneos

Para lidiar con el hecho de que muchos teclados vienen con una tasa de repetición tan alta como para hacerlos inutilizables, se puede instalar kbdrate en /sbin en algunos sistemas. Dado que la acción por defecto del kernel ante la combinación de teclas Ctrl+Alt+Del es un re-inicio instantáneo duro, es recomendable generalmente deshabilitar esta conducta antes de montar el sistema de archivos raíz con modo lecturaescritura. Algunas suites init son capaces de deshabilitar Ctrl-Alt-Del, pero otras pueden requerir el programa ctrlaltdel, el cual puede ser instalado en /sbin en estos sistemas: **Ctrl.+alt+del, kbdrate**

/tmp: Archivos temporales

El directorio tmp se utiliza para archivos temporales, preferentemente en un /dispositivo rápido (un sistema de archivos basado en memoria por ejemplo). La "persistencia" de la información que es almacenada en /tmp es diferente de aquella que sea almacenada en /var/tmp. /tmp puede ser limpiada en cada arranque o a intervalos relativamente frecuentes. Por tanto, no se debe esperar que la información almacenada en /tmp permanezca por algún periodo largo de tiempo.

Los programas deben utilizar /tmp 0 /var/tmp (que era originalmente /usr/tmp) de acuerdo a los requerimientos esperados de la información, pero no deben confiar en alguna persistencia temporal particular en

cualquier directorio de almacenamiento temporal.

Los administradores de sistemas pueden elegir enlazar /tmp a algún otro directorio, tal como /var/tmp; esto es útil, por ejemplo, para conservar espacio en la partición raíz. Si esto se lleva a cabo, entonces la persistencia de archivos en /var/tmp debe ser al menos tan larga como la de /tmp.

El directorio /tmp puede estar en un disco RAM. /var/tmp no debe nunca localizarse en /algún dispositivo RAM.

La Jerarquía /usr

El usr es la segunda mayor seccion del sistema de archivos. /usr es /información compartible, de sólolectura, esto significa que /usr, debe ser /compartible entre varias máquinas que corren Linux y no se debe /escribir. Cualquier información que es local a una máquina o varía con el /tiempo, se almacena en otro lugar.

Ningún paquete grande (como TeX o GNU Emacs) debe utilizar un subdirectorio directo bajo /usr, en vez, debe haber un subdirectorio dentro de /usr/lib (o /usr/local/lib si fué instalado completamente local) para ese propósito, con el sistema X Window se hace una excepción debido a un considerable precedente y a la práctica ampliamente aceptada.

/usr	**Segundo mayor punto de montaje (permanente)**
X11R6	Sistema X Window Versión 11 release 6
X386	Sistema X Windows Versión 11 release 5 en plataformas X 86
bin	La mayoría de los comandos de usuario
dict	Listas de palabras
doc	Documentación miscelánea
etc	Configuración del Sistema (todo el site)
games	Juegos y binarios educacionales
include	Archivos header incluidos por programas C
info	Directorio primario del sistema GNU Info

lib	Librerías
local	Jerarquía local (vacía justo después de la instalación principal)
man	Manuales en línea
sbin	Binarios de Administración del Sistema No-Vitales
share	Información independiente de la arquitectura
src	Código fuente

Los siguientes enlaces simbólicos a directorios pueden estar presentes. Esta posibilidad se basa en la necesidad de preservar la compatibilidad con sistemas anteriores hasta que en todas las implementaciones se pueda asumir el uso de la jerarquía /var:

/usr/adm ——————> /var/adm
/usr/preserve ——————> /var/preserve
/usr/spool ——————> /var/spool
/usr/tmp——————> /var/tmp
/var/spool/locks ———> /var/lock

Una vez que el sistema ya no requiera más alguno de los anteriores enlaces simbólicos, el enlace se puede remover, si se desea. Notablemente, sólo se necesita poco esfuerzo para remover completamente /usr/preserve, dado que sólo ex y vi lo utilizan.

/usr/X11R6: El sistema X Window, Versión 11 release 6

Esta jerarquía está reservada para el sistema X Window, Versión 11 release 6 y archivos relacionados. Los directorios de esta subjerarquía de /usr/X11R6 - y el X Window System (Versión 11, release 6) son:

• bin • doc • include • lib • man

Para simplificar los problemas y hacer XFree86 más compatible con el sistema X Window en otros sistemas, los siguientes enlaces simbólicos deben estar presentes.

/usr/bin/X11 ——————> /usr/X11R6/bin
/usr/lib/X11 ——————> /usr/X11R6/lib/X11
/usr/include/X11 ————> /usr/X11R6/include/X11

En general, el software no se debe instalar o manejar vía los anteriores enlaces simbólicos. Sólo están para la utilización por usuarios. La dificultad está relacionada con la versión y el release del sistema X Window; en períodos transicionales es imposible saber que release de X11 está utilizándose.

Por la misma razón no debe existir un enlace desde /usr/X11 apuntando a la jerarquía del sistema X Window actual.

/usr/bin: La Mayoría de los Comandos del Usuario.

Éste es el directorio principal de comandos ejecutables en el sistema.

X11- Enlace simbólico hacia /usr/X11R6/bin

Debido a que los interpretadores de scripts de los shell (invocados con #! <ruta> en la primera línea del script de shell) no pueden depender de una ruta, es ventajoso el estandarizar la localización de ellos. La shell Bourne y C están fijos en /bin, pero Perl, Python, tlc se encuentran en muchos lugares diferentes /usr/bin/perl, /usr/bin/python y /usr/bin/tcl deben referenciar a los intérpretes de shell perl, python y tcl respectivamente. Éstos pueden ser enlaces simbólicos a la localización física de los intérpretes de shell.

/usr/dict: Listas de Palabras (Archivos recomendados en /usr/dict: words)

Tradicionalmente este directorio contiene sólo el archivo words, de palabras inglesas, el cual es utilizado

por look y varios programas de ortografía, words puede utilizar ortografía americana o británica. Los sites que requieran ambos, pueden enlazar words a /usr/dict/american-english 0 /usr/dict/britishenglish.

Las listas de palabras para otros lenguajes se pueden añadir usando el nombre en inglés para ese lenguaje, por ejemplo, /usr/dict/french, /usr/dict/danish, etc. Éstos deben, si es posible, utilizar un juegos de caracteres ISO 8859 que sea apropiado para el lenguaje en cuestión, si es posible el juego de caracteres ISO 8859-1 (Latin1) debe ser utilizado (esto es a veces imposible).

Cualquier otra lista de palabras, tal como el directorio web2, debe ser incluido aquí, si está presente. Las razones tras tener sólo las listas de palabras aquí es que ellas son los únicos archivos comunes a todos los verificadores de ortografía.

/usr/etc: Configuración del Sistema (todo el site)

Almacenar la configuración en /usr/etc del software que se encuentra en /usr/bin y /usr/sbin es un problema. Hace que el montar /usr sólo-lectura de un CDROM o a través de NFS sea difícil en el mejor de los casos. Una posible solución que se consideró fue eliminar completamente /usr/etc y especificar que todas las configuraciones se almacenen en /etc. Un problema con esta aproximación es que no anticipa propiamente la posibilidad de que muchos sites pueden querer tener algunos archivos de configuración que no sean locales de máquina.

Eventualmente se decidió que /etc deberá ser el único directorio que sea referenciado por los programas (esto es, todos deben buscar configuraciones en /etc y no en /usr/etc). Cualquier archivo de configuración que necesite ser para todo el site y que no es necesario antes de montar /usr (o en una situación de emergencia debe entonces estar localizado en /usr/etc. Entonces archivos específicos (en /etc), en máquinas específicas pueden ser o no ser enlaces simbólicos a los archivos de configuración localizados en /usr/etc. Esto también significa que /usr/etc es técnicamente un directorio opcional en el sentido estricto, pero aún así recomendamos que todos los sistemas Linux lo incorporen.

No se recomienda que /usr/etc contenga enlaces simbólicos que apunten a archivos en /etc. Esto es innecesario e interfiere con el control local en máquinas que comparten un directorio /usr.

/usr/include: Directorio para Archivos Include Estándar

Aquí es donde todos los archivos include de uso general del sistema para programación en lenguajes C y C++ deben ser localizados.

/usr/include- Archivos include, algunos de sus directorios:

X11	Enlace simbólico hacia /usr/X11R6/include/X11
arpa	Definiciones del protocolo definido por ARPNET.
asm	Enlace simbólico hacia /usr/src/linux/include/asm-<arch>.
bsd	Archivos include de compatibilidad con BSD.
g++	Archivos include de GNU C++.
gnu	Archivos include GNU.
linux	Enlace simbólico a /usr/src/linux/include/linux.
net	Definiciones genéricas relacionadas con redes.
netax25	Definiciones específicas a +AX25 (ARRL AX25).
netinet	Definiciones específicas a TCP/IP.
netipx	Definiciones específicas a +IPX (Novel IPX/SPX).
protocols	Definiciones de protocolos (Mayormente basadas en INET)
readline	La librería readline GNU.

rpc	Definiciones RPC de Sun Microsystems.
rpcsvc	Definiciones de servicios RPC de Sun Microsystems.
sys	Archivos include de generación de sistemas.

El subdirectorio arpa contiene definiciones de cabecera de protocolos para los protocolos ARPANET, TCP/IP, definiciones para ftp, prototipos telnet y material similar.

El subdirectorio net contiene definiciones genéricas relacionadas con redes, define la interfase sistema-kernel, detalles de la familia de protocolo, etc.

El subdirectorio netinet contiene definiciones específicas de INET (DARPA Internet, que también es conocida como TCP/IP) ARRLAX.25 es mejor conocido como packet radio. Los protocolos Novell IPX/SPX son parte de los servicios de archivos Novell NetWare.

/usr/lib: Librerías para Programas y Paquetes

El usr/lib incluye librerías objeto, binarios del programa compilador e /información estática de varias clases, ambos, códigos ejecutable (por /ejemplo los binarios internos de gcc están localizados bajo //usr/lib/gcc-lib) y otros tipos de información.

/usr/lib/ - librerías para programación y paquetes:

X11	Enlace simbólico a /usr/X11R6/lib/X11
emacs	Archivos de soporte estáticos para el editor GNUEmacs.
games	Archivos de datos estáticos para /usr/games.
groff	Librerías / Directorios para GNU groff
gcc-lib	Archivos/ Directorios específicos del sistema para gcc.
kbd	Tablas de traducción de teclado e información relacionada.
Mh	Librerías para el sistema de manejo de correo MH:
news	Cnews/INN.
smail	Smail.
terminfo	Directorios para la base de datos terminfo.
texmf	TeX/MF (y LaTeX) librerías de información.
uucp	Comandos de UUCP.
zoneinfo	Configuración e información de la zona horaria.

Históricamente, /usr/lib ha incluido además algunos comandos ejecutables tales como sendmail y makewhatis.

Dado que makewhatis no es referenciado por otros programas, no hay problemas al moverlo a un directorio binario. Dado que los usuarios tienen buena razón para usar makewhatis, /usr/lib es donde pertenece. El binario catman que remplaza al script makewhatis en muchos sistemas Linux, debe también estar en usr/bin

El binario sendmail es referenciado por muchos programas con su nombre histórico /usr/lib/sendmail. Éste debe ser un enlace simbólico a la localización estándar para los agentes de transferencia de correo con una interfaz de línea de comando compatible con sendmail, /usr/bin/sendmail.

En sistemas que utilizan smail deben localizar smail en /usr/sbin/smail y /usr/bin/sendmail debe ser un enlace simbólico a smail.

Este arreglo también se conforma a la nueva ubicación estándar sendmail definida en Sendmail 8.6.x y BSD 4.4. Note que esta localización demanda que /usr/sbin y /usr/sbin/sendmail deben ser ejecutables para

usuarios normales.

Cualquier paquete o programa que contenga o requiera información que no necesite ser modificada debe almacenar tal información en /usr/lib (0 /usr/local/lib, si está instalado localmente). Se recomienda la utilización de un subdirectorio en /usr/lib para este propósito.

La información de juegos almacenada en /usr/lib/games debe ser solamente información estática. Cualquier archivo modificable. Tal como archivos de marcadores, registros de juego y similares, deben ser localizados en var/lib. Si es necesario para compatibilidad de juegos con el viejo estilo BSD, se puede usar un enlace simbólico desde /usr/games/lib hacia /usr/lib/games.

Nota: ninguna información específica de host para el sistema X Window debe almacenarse en /usr/lib/X11 (que es realmente /usr/X11R6/lib/X11). Los archivos de configuración específicos de host tales como Xconfig 0 XF86Config deben almacenarse en /etc/X11. Éste debe incluir información de configuración como system.twmrc aún si es sólo un enlace simbólico a un archivo de configuración más global (Talvez en /usr/etc/X11 0 /usr/X11R6/lib/X11).

/usr/local: Jerarquía Local

La jerarquía /usr/local está para ser utilizada por el administrador del sistema cuando se instale el software localmente. Necesita estar a salvo de ser sobrescrito cuando el software del sistema se actualiza. Puede ser usado por programas y por información que son compartibles entre un grupo de máquinas, pero no se encuentran en /usr.

/usr/local- Jerarquía local:

bin	Binarios sólo-locales
doc	Documentación local
etc	Binarios de configuración sólo-local
games	Juegos instalados localmente
lib	Librería para /usr/local
info	Páginas de info local
man	Jerarquías de páginas de manual para /usr/local
sbin	Administración del sistema sólo-local
src	Código fuente local.

Este directorio debe estar vacío al terminar de instalar Linux por primera vez. No debe haber excepciones a la regla, excepto quizá los subdirectorios vacíos listados.

El software instalado localmente debe estar localizado dentro de /usr/local, en vez de /usr a menos que este siendo instalado para reemplazar o actualizar el software en /usr.

Note que el software localizado en / o en /usr puede ser sobrescrito por actualizaciones del sistema (aunque recomendamos que las distribuciones no sobrescriban información en /etc bajo estas circunstancias). Por esta razón, el software local no se debe localizar fuera de /usr/local sin una buena causa.

/usr/man: Páginas del Manual.

Esta sección detalla la organización de las páginas del manual a través del sistema. Incluyendo /usr/man. Las páginas del manual están almacenadas <mandir>/<locale>/man [1-9]. Más delante se dá una explicación de <mandir> y <locale>.

<mandir>/<locale>- Una jerarquía de páginas de manuales.

man1	Programas para usuarios.

man2	Llamadas al sistema.
man3	Subrutinas y funciones de librería
man4	Dispositivos.
man5	Formato de archivos
man6	Juegos.
man7	Misceláneas.
man8	Administración del Sistema.
man9	Funciones y variables internas del kernel.

El <mandir> primario del sistema es /usr/man contiene información del manual para comandos e información bajo los sistemas de archivos / y /usr. Obviamente no hay páginas de manual en / por que no se necesitan para arrancar ni en emergencias.

Se debe hacer provisión en la estructura de /usr/man para el soporte de páginas del manual que estén escritas en diferentes (o múltiples idiomas). Estas provisiones deben tomar en cuenta el almacenamiento y referencia de estas páginas del manual. Los factores relevantes incluyen el idioma (incluyendo diferencias basadas en la geografía) y el código del conjunto de caracteres.

Esta nomenclatura de los subdirectorios de idiomas de /usr/man está basada en el apéndice E del estándar POSIX 1003.1 que describe la cadena de identificación locale —-El método más aceptado para describir un ambiente cultural. La cadena <locale>es:

<idioma>[<_territorio>][.<conjunto_de_caracteres>][,<versión>]

El campo <idioma> se tomará del ISO639 (un código para la representación de los nombres de los idiomas). Será de dos caracteres de ancho y especificado con minúsculas solamente.

El campo <_territorio> será el código de dos letras de ISO3116 (una especificación de la representación de los nombres de los países), si es posible. (Mucha gente está familiarizada con el código de 2 letras empleado en el código de país en las direcciones de e-mail.

El campo <conjunto_de_caracteres> debe representar el estándar que describe el código de caracteres. Si el campo <conjunto_de_caracteres> es sólo una especificación numérica, el número representa el número del estándar internacional que describe a ese conjunto de caracteres. Se recomienda que este sea una representación numérica siempre que sea posible (especialmente, estándares ISO), que no incluya símbolos de puntuación y que todas las letras sean minúsculas.

Un parámetro que especifique <versión> del perfil puede ser colocada después del campo <conjunto_de_caracteres >, delimitado con una coma. Ésto puede utilizarse para discriminar entre diferentes necesidades culturales, por ejemplo un orden de diccionario en vez de un orden de acomodo más orientado hacia el sistema. Este estándar recomienda no usar el campo <versión> a menos que sea necesario.

En sistemas que usen sólo un idioma y un código de conjunto de caracteres para todas las páginas del manual, pueden omitir la subcadena <locale> y almacenar todas las páginas del manual en <mandir>. Por ejemplo en sistemas que sólo tienen páginas del manual en inglés codificados en ASCII, pueden almacenar las páginas del manual (los directorios man[1-9]) directamente en /usr/man (éstas son las circunstancias y el arreglo tradicional, de hecho).

En países para los cuales existe un código de conjunto de caracteres estándar, puede omitir el campo <conjunto_de_caracteres>, pero recomendamos fuertemente que se incluya, especialmente para países con varios estándares en competencia. Varios ejemplos:

Idioma	Territorio	Conjunto de caracteres	Directorio
Inglés	----------	ASCII	/usr/man/en
Inglés	Reino Unido	ASCII	/usr/man/en_GB
Inglés	Estados Unidos	ASCII	/usr/man/en_US
Francés	Canadá	ISO8859-1	/usr/man/fr_CA
Francés	Francia	ISO8859-1	/usr/man/fr_FR
Alemán	Alemania	ISO646-DE	/usr/man/de_DE646de
Alemán	Alemania	ISO6937	/usr/man/de_DE6937
Alemán	Alemania	ISO8859-1	/usr/man/de_DE.88591
Alemán	Suiza	ISO646-CH	/usr/man/de_CH.646ch
Japonés	Japón	JIS	/usr/man/ja_JP.jis
Japonés	Japón	SJCS	/usr/man/ja_JP.sjis
Japonés	Japón	UJ (o EUC-J)	/usr/man/ja_JP.ujis

Las páginas del manual para los comandos e información que se encuentra bajo /usr/local están almacenadas en /usr/local/man. las páginas del manual para el sistema X Window están almacenadas en /usr/X11R6/man. Luego todas las jerarquías de páginas del manual en el sistema deben tener la misma estructura que /usr/man. Los directorios vacíos pueden ser omitidos de la jerarquía de páginas del manual. Por ejemplo si, /usr/local/man no tiene páginas del manual en la sección 4 (dispositivos) entonces se puede omitir /usr/local/man/man4.

Las secciones de páginas cat (cat[1-9]) que contiene páginas del manual formateadas, también se encuentran dentro de los subdirectorios /<mandir>/<locale>, pero no son requeridas ni deben ser distribuidas en el lugar de las fuentes nroff de las páginas del manual. Las páginas del Manual del sistema de manejo de correo mh deben tener el prefijo mh en todos los nombres de archivos de las páginas.

Las páginas del sistema X Window deben tener el prefijo x en todos los nombres de los archivos de las páginas. La práctica de colocar las páginas del manual de diferentes idiomas, en los subdirectorios apropiados de /usr/man también se aplica a las otras jerarquías de páginas del manual, tales como /usr/local/man y /usr/X11R6/man. (Esta porción de este estándar también se aplica más delante en la estructura opcional de /var/catman). A continuación una descripción de cada sección.

man1: Programas del Usuario.

Las páginas del manual que describen los comandos accesibles públicamente se encuentran en esta sección. La mayoría de la documentación de los programas que un usuario necesitará se encuentra aquí.

man2: Llamadas al Sistema.

Esta sección describe todas las llamadas al sistema (requisiciones hacia el kernel de Linux para realizar ciertas operaciones).

man3: Subrutinas y Funciones de Librería.

La sección 3 describe programas rutinas de librería que no son llamadas directas a servicios del kernel. Esta sección y la 2 son de interés casi solamente para programadores.

man4: Archivos Especiales de Dispositivos.

Esta sección describe los archivos especiales, funciones relacionadas con los manejadores y el soporte para la red que estén disponibles en el sistema. Típicamente, esto incluye los archivos de dispositivo que se encuentran en /dev y la interfaz del kernel para el soporte de protocolos de red.

man5: Formatos de Archivos

Aquí se encuentran los formatos para muchos de los archivos cuyo formato no sea intuitivo. Esto incluye varios archivos include, archivos de salida de programas y archivos de sistema.

man6: Juegos.(Binarios Educativos)

Esta sección documenta los juegos, demos y programas triviales. Muchas personas tienen una opinión muy diferente de que tan esencial es esta sección.

man7: Misceláneos

Las páginas del manual que son difíciles de clasificar se designan como pertenecientes a la sección 7. Las de troff y otros macro paquetes de procesamiento de texto se encuentran aquí.

man8: Administración del Sistema

Aquí se documentan los programas utilizados por los administradores de sistemas para la operación y mantenimiento. Algunos de estos programas son ocasionalmente útiles para usuarios normales.

man9: Funciones y Variables Internas del Kernel

Este es utilizado para documentar el código fuente del kernel en los Sistemas Linux.

/usr/sbin: Binarios de Sistema Estándar No-Esenciales

Este directorio contiene cualesquier binario no-esencial utilizado exclusivamente por el administrador del sistema. Los programas de administración del sistema que sean requeridos para la reparación del sistema, recuperación del sistema, montaje de /usr u otras funciones esenciales deben localizarse en /sbin en vez de aquí.

Típicamente /usr/sbin contiene los daemons de red, cualquier herramienta de administración no-esencial y binarios para programas servidores no-críticos. Esto incluye los daemons de Internet que son llamados por inetd (llamados in.*) tales como in.telnetd y in.fingerd y los daemons basados en rpc manejados por portmap (llamados rcp.*) tales como rcp.infsd y rcp.mountd.

Estos programas servidores son utilizados cuando se entra en un estado que el System V conoce como “run level 2” (estado multi-usuario) y el “run level 3” (estado en-red) o el estado que el BSD conoce como “modo multi-usuario”. En este punto se hacen disponibles los servicios para los usuarios (p. Ej. soporte de impresión) y hacia otras máquinas (p. Ej. exportar NFS).

Los programas administrativos instalados localmente deben estar localizados en : **/usr/local/sbin**.

/usr/share: Información Independiente de Arquitectura

Cualquier especificación para /usr/share se incluirá en un documento suplementario al FSSTND. Note que es la opinión en consenso del FSSTND que /usr/share no es necesario en la mayoría de los sistemas GNU/Linux. En este momento, si nos confinamos a proporcionar una definición extensiva de este directorio, sería una mala idea. Por favor refiérase a la sección 6 para una discusión más de /usr/share.

/usr/src: Código Fuente

Código fuente para el kernel de Linux. Cualquier código fuente no-local debe localizarse en este directorio. El único código fuente que siempre debe localizarse en un lugar específico es el código del kernel (cuando exista o esté enlazado como parte de una estructura en /usr/include). Se pueden usar subdirectorios si se desea.

El código fuente para el kernel debe siempre estar en su lugar o al menos los archivos include del código del kernel. Esos archivos están localizados en estos directorios:

/usr/src/linux/include/asm-<arch>
/usr/src/linux/include/linux

El usr/include debe contener enlaces a estos directorios, llamados asm y /Linux, dados que son necesitados por el compilador de C, al menos /estos archivos include deben siempre ser distribuidos en las instalaciones /que incluye un compilador C. Deben ser distribuidos en el directorio /usr/src/linux de forma que no existan problemas cuando los administradores /del sistema actualicen su versión del kernel por primera vez. /usr/src/linux puede también ser un enlace simbólico a un árbol de código /fuente del kernel.

La Jerarquía /var

/var	Información variable
adm	Info administrativa del sistema (obsoleto). Enlace simbólico hacia /var/log
catman	Páginas del manual formateadas localmente
lib	Información del estado de aplicaciones
local	Información variable del software de /usr/local
lock	Archivos de bloqueo
log	Archivos de bitácora
named	Archivos DNS, sólo red
nis	Archivos base de datos NIS
preserve	Archivos almacenados después de una falla de ex 0 vi
run	Archivos relevantes a procesos ejecutándose
spool	Directorios de trabajos en fila para realizarse después
tmp	Archivos temporales, utilizado para mantener /tmp pequeño

El var contiene archivos con información variable. Esto incluye archivos y /directorios en fila de ejecución, información de bitácora administrativa y /archivos temporales y transitorios.

Algunas porciones de /var son no-compartibles entre diferentes sistemas. Por ejemplo, /var/log, /var/lock y /var/run. Otras porciones son compartibles, notablemente /var/spool/mail y /var/spool/news.

El var se especifica aquí para hacer posible el montar /usr sólo-lectura. /Todo aquello que alguna vez fué en /usr que es escrito durante la operación /normal del sistema (pero no durante la instalación y el mantenimiento del /software) debe ir en /var.

Si /var no puede ser una participación separada, es preferible mover /var fuera de la participación raíz pero dentro de la partición /usr (esto se hace algunas veces para reducir el tamaño de la partición raíz o cuando hay poco espacio en la partición raíz). Como sea, /var no debe ser enlazada a /usr, porque hace que la separación entre /usr y /var sea más difícil y seguramente creará un conflicto de nombres, En vez enlace /var a /usr/var.

/*var/adm: Bitácora del Sistema y Archivos Contables (obsoleto)*

Este directorio ha sido remplazado por /var/log y otros directorios. Debe ser un enlace simbólico a /var/log hasta que todos los programas ya no se refieran más a algún archivo en /var/adm. El utmp se ha movido a /var/run. Todos los archivos bitácoras se han movido a /var/log incluyen en el archivo wtmp. El soporte de empaquetado de distribuciones se debe almacenar en /var/lib/<nombre>.

Nota: El enlace simbólico /var/adm no debe ser necesario en la mayoría de los sistemas Linux-i386ELF dado que el cam-

bio fué introducido antes que ELF fuera liberado al público.

/var/catman: Páginas del Manual Formateadas Localmente (opcional)

Este directorio proporcionara una localización estándar para los sites que proporcionan una partición /usr sólo-lectura, pero que desean permitir el almacenamiento temporal de páginas del manual formateados localmente. Los sites que montan /usr como escribible (Ej. instalaciones mono-usuarios) pueden escoger no usar /var/catman y escribir las páginas del manual formateadas dentro de los directorios cat[1-9] dentro de /usr directamente. Recomendamos que la mayoría de los sites utilicen una de las siguientes opciones en su lugar.

- Prefórmatee todas las páginas del manual dentro de /usr con el programa (catman).
- No se permita el almacenamiento temporal de las páginas formateadas del manual y requiera que se ejecute nroff cada vez que se necesite una página.
- Se permita el almacenamiento temporal local de las páginas del manual en /var/catman.

La estructura de /var/catman necesita reflejar ambos, el hecho de la existencia de múltiples jerarquías de página del manual y la posibilidad del uso de múltiples idiomas.

Dada una página del manual sin formatear que normalmente aparece en /usr/<ruta1>/man/man[1-9], la versión formateada almacenada temporal debe ir en /var/catman/<ruta2>/cat[1-9], donde <ruta2> es <ruta1>. Los componentes <ruta2> y <ruta1> están ausentes en el caso de /usr/man y /var/catman.

Por ejemplo:
/usr/man/man1/ls.1 se formatea en /var/catman/cat1/ls.1 y /usr/X11R6/man/<locale>/man3/XtClass.3x lo hace en /var/catman/X11R6/<locale>/cat3/XtClass.3x

Las páginas del manual escritas en /var/catman/cat[1-9] pueden eventualmente, transferirse a /usr/<ruta>/cat[1-9] 0 expirarse. De igual forma las páginas del manual formateadas dentro de /usr/<ruta>/cat[1-9] pueden expirarse si no son accesadas en un periodo de tiempo.

Si vienen páginas del manual preformateadas con un sistema Linux en un medio sólo-lectura. (Ej. un CDROM), deben estar instaladas en /usr/<ruta>/cat[1-9]. /var/catman está reservado como un lugar de almacenamiento temporal para páginas de manual formateados.

/var/lib: Información del Estado de Aplicaciones

emacs	Directorio del estado de Emacs
games	Información variable de juegos(archivos de marcadores)
news	Archivos variables de Cnews/INN
texmf	Información variable asociada con TeX
xdm	Archivos de autenticación y bitácoras de error del manejador de despliegues X

El var/lib/<nombre> es el lugar apropiado para el soporte de /empaquetamiento de todas las distribuciones. Diferentes distribuciones de /Linux pueden utilizar diferentes nombres por supuesto.

/var/lib/emacs

El directorio del estado GNU Emacs, el lugar donde los archivos de información independiente de la arquitectura, que Emacs modifica cuando se ejecuta, debe ser /var/lib. En el presente, Emacs sólo localiza su directorio de archivos de bloqueo bajo el directorio de estado (en <direstado>/emacs/lock), pero puede hacer uso más extenso del mismo en el futuro. Notablemente, sólo se requiere la adición de una opción sencilla en el programa configure de Emacs para hacer este cambio (antes de compilar).

/var/lib/games

Así como los subdirectorios antes citados, cualquier información variable relacionada con los juegos que se encuentran en /usr/games, deben estar aquí. /var/lib/games debe incluir la información variable que previamente se encontraba en /usr/lib/games; La información estática, tal como textos de ayuda, descripciones del nivel y demás, debe permanecer en /usr/lib/games.

/var/lib/news

El var/lib/news se debe usar para almacenar toda la información variable /asociada con los servidores de news tales como Cnews y INN, inclusive el /archivo histórico, el archivo activo y demás.

/var/lib/texmf

El var/lib/texmf se debe usar para almacenar la información variable /asociada con TeX. Particularmente, en /var/lib/texmf/fonts se /almacenarán todas las fuentes tipográficas que son generadas /automáticamente por MakeTeXPK.

Debe haber un enlace desde /usr/lib/texmf/fonts/tmp hacia /usr/lib/texmf/fonts. Este enlace permite a los usuarios hacer uso de una sola ruta /usr/lib/texmf/fonts/tfm cuando le hacen cambios a su variable de entorno TEXFONTS (ésta es la ruta por defecto en las herramientas TeX de Karl Berry distribuidas desde ftp.cs.umb.edu:pub/tex [La razón de mencionarlos aquí es que son el estándar de facto en las instalaciones UNiX, estas herramientas son ampliamente usadas entre la comunidad Linux]. Si se utiliza otra distribución de TeX, se debe hacer un enlace desde el directorio de fuentes apropiado hacia /usr/lib/texmf/fonts).

El MakeTeXPK que se distribuye con dvipsk colocará los archivos .pk en fonts/pk/<dispositivo>/<nombre_ de_la_fuente>, (Ej. fonts/pk/Canon_CX/cmr10.300pk).

Los archivos .pk se pueden purgar periódicamente del árbol /var/lib/texmf o se puede mover dentro del árbol /usr/lib/texmf. Si se usan generadores automáticos de .mf 0 .tfm, éstos deben poner su información en los subdirectorios mf 0 tfm de /var/lib/texmf/fonts.

/var/lib/xdm

El /var/lib/xdm contiene la información variable de xdm que consiste en los /archivos xdm-errors y cualquier archivo de autoridad xdm. Los binarios de /xdm tales como chooser deben aún estar localizados en la localidad /histórica en /usr/X11R6/lib/X11/xdm. El archivo xdm-pid debe estar en //var/lib/xdm a pesar de existir /var/run. Los archivos restantes deben estar en /etc/X11/xdm.

/var/local: Información Variable del Software que está en /usr/local

Este directorio contiene toda la información variable que esté relacionada con el software que se encuentra en /usr/local. Naturalmente la implementación de este subdirectorio se deja a el administrador del sistema. Como sea la información que se puede categorizar en otro lugar del directorio /var, no se debe colocar en /var/local. Por ejemplo, todos los archivos de bloqueo aún irán en /var/lock.

/var/lock: Archivos de Bloqueo

Los archivos de bloqueo deben almacenarse dentro de una estructura del directorio de /var/lock. Para preservar la habilidad de montar /usr sólo-lectura, no se deberá colocar los archivos de bloqueo en la partición /usr. Los archivos de bloqueo de dispositivo, tales como los archivos de bloqueo de dispositivos serie que antes se encontraban en /usr/spool/lock 0 en /usr/spool/uucp deben ahora almacenarse en /var/lock. La convención para la nomenclatura que debe utilizarse es LCK... seguido del nombre base del dispositivo. Por ejemplo, para bloquear /dev/cua0 se deberá crear el archivo LCK... cua0.

El formato usado para los archivos de bloqueo de dispositivo en Linux deberá ser el formato de archivos de bloqueo HDB UUCP. El formato HDB es almacenar el PID (Identificador del proceso) como un número decimal en ASCII de 10 bytes, con un caracter de línea nueva.

Por ejemplo, si el proceso 1230 retiene un archivo de bloqueo, contendrá los siguientes once (11) caracteres: espacio, espacio, espacio, espacio, espacio, espacio, uno, dos, tres, cero y nueva línea.

Entonces cualquier cosa que desee usar /dev/cua0, puede leer el archivo de bloqueo y actuar de acuerdo (Todos los archivos de bloqueo en /var/lock deben ser leíbles por todos).

/*var/log: Archivos Bitácora y Directorios*

Este directorio contiene archivos bitácora misceláneos. la mayoría de los archivos bitácora se deben escribir en este directorio o subdirectorios apropiados.

lastlog	Registro del último acceso de cada usuario
messages	Mensajes del sistema desde syslogd
wtmp	Registro de todos los accesos y salidas

Se puede requerir de un enlace simbólico desde /var/log/utmp hacia /var/run/utmp hasta que ningún programa se refiera a /var/adm/utmp (/var/adm es en sí mismo un enlace simbólico transicional hacia /var/log).

/var/named: Archivos DNS

Este directorio contiene todos los archivos de trabajo del servidor de nombres Internet, named. Recomendamos que /etc/named.boot sea un enlace simbólico hacia /var/named/named.boot, dado que /etc/named.boot es el archivo de arranque por defecto, si no se dan argumentos a named.

/var/nis: Archivos de Bases de Datos del Servicio de Información de Red (NIS)

El sistema de información de red (NIS) era anteriormente conocido como las Páginas Amarillas Sun (YP). La funcionalidad y localización de directorios de ambos es el mismo pero el nombre (Yellow Pages) es una marca registrada en el Reino Unido, pertenece a Bristish Telecommunications PLC y no puede ser usada sin permiso.

/var/preserve: Archivos Guardados después Caída 0 Terminación Inesperada de ex 0 vi

Este directorio contiene los archivos que son almacenados ante cualquier terminación no-esperada de ex, vi, 0 de alguno de sus clones.

/var/run: Archivos Variables de Tiempo de Ejecución

Este directorio contiene archivos con información del sistema que lo describen desde que arrancó. Generalmente los archivos en este directorio se deben limpiar (remover o truncar, según corresponda) al comenzar el proceso de arranque.

Los archivos identificados de poceso (PID), que estaban originalmente /etc, se deben colocar en /var/run. La convención de nomenclatura de archivos PID es <nombre-programa>.pid, por ejemplo el archivo PID de crond se llama /var/run/crond.pid.

El formato interno de los archivos PID permanecen sin cambio. El archivo debe de consistir del indicador del proceso en decimales codificado como ASCII, seguido por un carácter nueva línea Por ejemplo, si crond fue el proceso número 25, /var/run/cond.pid contendrá 3 caracteres, dos cinco y nueva línea.

Los programas que léan archivos PID deben ser flexibles en lo que aceptan, Ej. Deben ignorar los espacios extras, ceros a la izquierda, ausencia del carácter nueva línea o líneas adicionales en el archivo PID. Los programas que crean archivos PID deben utilizar la sencilla especificación dada en el anterior párrafo.

El archivo utmp, que almacena información acerca de quién está actualmente utilizando el sistema, se localiza en este subdirectorio. Los programas que mantengan sockets transitorios de dominio UNIX, deben colocarlos en este directorio.

/var/spool: Directorios de Fila de Trabajos para Procesamiento Posterior

El var/spool es tradicionalmente utilizado para la información local de /máquina que es enviada para procesarse después, hacia o desde subsistemas /UNIX. Por ejemplo, trabajos de impresión que son almacenados aquí para /entrega posterior al daemon de la impresora, el correo que sale es /almacenado aquí para entrega a sistemas remotos y los archivos UUCP son /almacenados aquí para transmisión a los sistemas UUCP vecinos. El correo /que entra y las noticias son almacenados aquí para entregarse a los /usuarios y los trabajos de at y cron son almacenados aquí para ejecución /retardada por el daemon cron.

/var/spool

at	Trabajos de at
cron	Trabajos de cron
lpd	Directorio de impresora *
mail	Archivos caja-postal(buzón) de los usuarios
mqueue	Fila del correo saliente
news	Directorio de noticias *
rwhod	Archivos rwhod
smail	Directorio de smail *
uucp	Directorio de UUC

*Nota: * Significa fila de trabajos para proceso posterior.*

Los archivos de bloqueo UUCP deben localizarse en /var/lock. Véa la sección acerca de /var/lock.

/var/spool/lpd

/var/spool/lpd-	Directorio de fila de trabajos para proceso posterior o impresión
<impresora>	Directorio que tiene la fila específica de esta impresora

El archivo de bloqueo para lpd, lpd.lock debe estar localizado en /var/spool/lpd. El archivo de bloqueo de cada impresora debe localizarse en el directorio <impresora> de la impresora especifica y se debe llamar lock.

/var/tmp: Archivos Temporales, Utilizando para Mantener /tmp Pequeño

Los archivos que están en /var/tmp están almacenados por una duración no especifica. (Recuerde que los directorios temporales del sistema no garantizan mantener la información por ningún periodo particular).

La información almacenada en /var/tmp típicamente se limpia en una “forma definida localmente” pero usualmente menos frecuentemente que /tmp. Se puede encontrar información sobre directorios temporales en la selección dedicada a /tmp (arriba).

Debe existir un enlace simbólico desde /usr/tmp hacia var/tmp por razones de compatibilidad.

La Red

La red presenta un dilema interesante, algunas personas quieren separar los binarios de configuración de la red de los que no lo son. Como sea, estamos en desacuerdo. Sentimos que la red no es un "paquete", sino una parte integral de la mayoría de la máquinas UNiX (y similares). Debido a lo anterior, la red no debe colocarse en un sólo directorio sino localizarse sistemáticamente en los directorios apropiados.

/bin: Cualquier cosa que algún usuario querrá utilizar y que estambién considerado vital:
hostname, netstat, ping

/sbin: Cualquier cosa que sólo root necesita y se considere vital:
arp, ifconfig, route

/usr/bin: Algunos binarios que algún usuario querrá utilizar y que no son vitales:
finger, rep, rlogin, telnet, etc.

/usr/sbin : Algunos binarios sólo para el administrador que no son vitales:
in.ftpd, inetd, lpd, portmap, etc.

Aunque puede parecer confuso al principio (y toma tiempo digerirlo), tiene sentido. Si por alguna razón usted sólo puede montar la partición raíz y necesita accesar a la red para reparar su sistema, no se quiere que los archivos estén en /usr/etc (como están algunas veces). Los archivos que se necesitan para montar /usr en las situaciones normales (y de emergencia) están colocados dentro del sub-árbol raíz y cualesquier otros se colocan en /usr, para mantener el tamaño del sistema de archivos raíz pequeño. Los archivos de configuración para la red pertenecen a /etc.

Estructuras Independientes de la Arquitectura

El directorio /usr/share típicamente contiene archivos independientes de la arquitectura, tales como páginas del manual, zona horaria, información de terminfo, etc. En el momento presente no hay diferentes arquitecturas para Linux, pero con el tiempo, veremos que Linux incluirá otras arquitecturas y otros sistemas similares a UNiX.

Nota: Ningún programa nunca deberá hacer referencia a alguna cosa en /usr/share. Por ejemplo, un programa de páginas del manual no debe nunca buscar directamente /usr/share/man/man1/1s.1, sino que siempre se debe referir a /usr/man/man1/1s.1. Cualquier cosa en /usr/share, será "apuntada" a través de uso de enlaces símbolos desde otras áreas del sistema de archivos, tales como /usr/man, /usr/lib/<algo>, etc... Aún se trabaja en las especificaciones de /usr/share.

Enlaces Simbólicos

Hay muchos usos para los enlaces simbólicos en cada sistemas de archivos. Aunque un estándar como éste no respalda el uso de enlaces simbólicos en la implementación por defecto (Los encontrados después de instalar Linux), se usan frecuentemente con buenos propósitos en diferentes sistemas. El punto es que los enlaces simbólicos deben estar allí para mantener todos los archivos y directorios donde cada quien los espera encontrar.

Está preparado para aceptar que ciertos directorios, aún aquellos contenidos en el directorio raíz, aún sean enlaces simbólicos. Por ejemplo en algunos sistemas /home no estará en el raíz, sino enlazado simbólicamente a un directorio /var o algún otro lugar. /home podría tener también su propia partición física y desde luego, ser montado como tal.

Similarmente, dado que /usr podría estar en un servidor de archivos central montado vía NFS, /usr/local

se puede enlazar simbólicamente a /var/local. Este cambio se puede justificar recordando la razón principal de tener /var: separar directorios de archivos que varían con el tiempo y entre diferentes sistemas y máquinas de aquellos que se pueden compartir y sean sólo-lectura.

Algunos sistemas además enlazarán /tmp a /var/<algo> si la partición raíz se vuelve muy pequeña (0 es muy pequeña). Hay más ejemplos de "buenos" usos de enlaces simbólicos, pero todo el asunto se reduce a dos cosas: Los paquetes deben ser capaces de encontrar las cosas donde lo esperan (razonablemente) y los enlaces simbólicos se pueden utilizar para resolver los problemas de muchos casos. Como sea, se pueden generar problemas con el uso de demasiados enlaces simbólicos. este problema incluye sobre-confianza en los enlaces simbólicos para resolver problemas, confusión resultante del sobre-uso de enlaces simbólicos y las preferencias estéticas de diferentes personas.

Binarios Compilados Estáticamente

GNU/Linux se ejecuta actualmente en una gama de sistemas, algunos con sólo un usuario y disco pequeño, otros como servidores en ambientes con redes muy grandes, dada esta variedad, este estándar no impone regla sobre qué binarios están compilados estáticamente o dinámicamente, con las siguientes excepciones. Ambos ln y sync, deben existir en /bin; cualquier versión estática se puede colocar en /sbin o remplazar aquellas en /bin.

Los grandes sistemas Linux pueden desear incluir otros binarios estáticos (sh, init, mkfs, fsck, tunefs, mount, umount, swapon, swopff, getty, login y otros). Los desarrolladores y los administradores de sistemas, son libres de enlazar dinámica o estáticamente éstos y otros binarios según le convengan, siempre que la localización de los binarios no cambie.

En sistemas de red, (especialmente aquellos que no tienen unidades de disco flexible), pueden querer compilar estáticamente ifconfig, route, hostname y otras herramientas de red. Esto usualmente no es necesario.

ARCHIVOS "DEFAULT" DEL SISTEMA

Los archivos default de GNU/Linux una vez eran incluido en los programas individuales. Gradualmente, archivos default configurables por el usuario fueron permitidos en una manera no muy organizada Linux almacena los defaults de algunos comandos en archivos almacenados en el directorio /etc/default las siguientes características aplican al contenido de este directorio:

- Nombre del archivo es el mismo nombre del comando
- Entradas en el archivo toman el formato de variables de ambiente
- Detalles del default son definidos en la pagina del manual del comando
- Los archivos son usualmente editados a mano

```
# more /etc/default/useradd
# useradd defaults file
GROUP = 125
HOME = /home
INACTIVE = -1
EXPIRE =
SHELL = /bin/bash
SKEL = /etc/skel
```

Resumen

En este capítulo fue introducido algunos conceptos de administración, incluyendo:

- El rol de la cuenta super usuario
- Utilizando el comando su para temporariamente ejecutar como un usuario diferente
- El rol de un administrados debe ocupar en el trabajo
- Usando una gama de programas utilitarios, incluyendo lo siguiente.
- uname
- who, id
- write, wall, talk
- more, less

Preguntas Post-Exámen

Las respuestas a estas preguntas están en el Apéndice A.

1.- ¿Por qué no debe usted permanecer en la cuenta root todo el tiempo?

2.- ¿Cómo pueden un usuario comunicarse con otro usuario?

3.- ¿Cuándo deben las tareas administrativas ser delegada? ¿En quién deben ser delegadas?

CAPITULO 2

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

KERNEL, MÓDULOS Y CONFIGURACIÓN

OBJETIVOS

Al completar este capítulo, usted podrá:

- Explique la función del Kernel y como interactúa con el resto del sistema.
- Explique la diferencia entre un Kernel Monolítico y uno Modular.
- Distinga entre Kernels Estables y Desarrollo. (Explique el sistema de numeración de versión del kernel)
- Navegue los fuentes del Kernel y la documentación que el contiene.
- Configure, Compile, e Instale un Kernel y sus Módulos.
- Describa el rol de Prueba de Estándares en acceso de CPU y Estructura de Sistema así como diseño de Compiladores.

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1.- ¿Cómo puedes determinar si un Kernel es Estable o de Desarrollo?

2.- ¿Cuáles son los tres métodos disponibles para configurar el Kernel antes de Compilarlo?

3.- ¿Qué es un módulo del Kernel?

4.- ¿Por cuáles razones compilarías un Kernel?

5.- ¿Cuáles son los objetivos del Kernel?

INTRODUCCION

El Kernel es el núcleo del sistema operativo. El actúa como un intérprete entre el usuario y el hardware. El kernel controla el acceso a los recursos de hardware de la computadora y determina como compartir estos recursos de una manera equitativa. El incluye los drivers del hardware, sistema de archivos, redes, manejo de memoria y administración de los procesos.

Virtualmente, el Kernel puede ser configurado y optimizado para cualquier entorno, a través de recopilación del kernel mismo. Podrías querer recompilar el kernel para incluir drivers para hardware específico o para actualizar drivers para la corrección de errores o para incluir nuevas características.

Linus Torvalds y un grupo de programadores están cometidos al desarrollo de mejoramiento del código fuente. Más allá con la cooperación de la comunidad Free Source, el kernel Linux ha logrado superar virtualmente todos los otros sistemas operativos hoy en existencia. En este capítulo cubriremos lo básico del kernel, compilarlo, sus módulos y optimización.

EL KERNEL DE GNU/LINUX

El Kernel es de los primeros software a ejecutarse en un computador. Una vez el kernel ha terminado su iniciación, hace un llamado al proceso init (llamado el proceso padre de todos los procesos). El kernel provee todas las funcionalidades básicas a los programas así como el manejo de los recursos del sistema: hardware, procesos, memoria, I/O y sistema de archivos. La funcionalidad del kernel es mejorada sumándole/removiéndole código compilado llamado módulos o manejadores (drivers) de dispositivos.

El kernel de Linux es un proyecto activo con un desarrollo continuado. En este proceso existen dos ramificaciones que viajan en paralelo. La primera es la denominada versión estable del kernel y su intención es para producción solamente y no investigación. La otra es la versión de desarrollo y es donde los desarrolladores denominados hackers prueban y analizan propuestas de mejoras. Casi siempre es inestable, con problemas y características incompletas. Puedes reconocer el tipo de kernel por sus números de versión. El formato de este número es: X.Y.Z, donde X es la versión mayor y es la menor y Z es el nivel de patch o revisión de mejora. Si la versión menor es impar entonces es desarrollo (developers) y si es par entonces es estable.

Por ejemplo, el kernel 2.2.19 es de versión estable mientras que 2.3.15 es de desarrollo. Para saber la versión del kernel basta con ejecutar el comando uname -r. Note algunas distros le colocan un cuarto numero para denotar el numero de cambios que ellos le han aplicado al kernel, ejemplo es Debian 2.2.19-6 cual es mayor 2, menor 2, parcho 19, versión 6.

Los administradores de sistemas deben usar versiones estables de kernel para así garantizarse estabilidad de servicios para sus usuarios. En casos escasos existen razones para utilizar un kernel de desarrollo, pueden ser:

- Lograr soporte para un dispositivo no soportado aun en los kernels estable.
- Obtener características no disponibles aun en las versiones estables.

Pueden existir razones para correr kernels inestables que pudiesen aparentar necesarias (como ayudar la Comunidad de Software Libre), pero un administrador debe recordar que su objetivo y sus responsabilidades son a sus usuarios, empleador y entonces a la Comunidad de Software Libre. El administrador debe solo colocar un kernel inestable solo cuando las circunstancia la obligan, por razones como las arriba mencionadas.

Al publicar este libro el kernel actual es el 2.4.21 y el de desarrollo es el 2.5.75 yo estoy actualmente usando un Debian con un 2.4.19. El sitio web de la fundación dedicada al mantenimiento y desarrollo del kernel es http://www.kernel.org. Sin embargo, los fabricantes proveen versiones mas especificas para sus distribu-

ciones. Estas contienen variaciones que ya ellos le han implementado para optimizar sus respectivas distros. Estos vendedores de Linux proveen versiones precompiladas que se instalan como si fuesen una aplicación.

En ésta seccion, discutiremos los siguientes tópicos:
- Estructura del Kernel
- Estructura de árbol del Código Fuente

Estructura del Kernel

El Kernel Linux es el componente base de todas las distribuciones de GNU/Linux. El Kernel hace que todas las partes del sistema, hardware y software, interactúen y operen. Muchos aspectos de la funcionalidad de GNU/Linux están construidos dentro del Kernel, como lo son soporte para los dispositivos a través de módulos y soporte para otro tipo más especifico de arquitecturas. El kernel también incluye soporte para protocolos de redes como el stack de TCP/IP y formatos de múltiple sistemas de archivos. Cuando soporte para este tipo de componentes es compilado en el kernel, es a menudo denominado soporte nativo.

Además de soporte de hardware y protocolos de comunicación, el kernel administra los recursos de memoria y tareas y provee una interfase de llamadas de sistema para que los programas tengan acceso a los recursos.

Programar el Tiempo

Una de las tareas principal del Kernel es programar las tareas (o procesos). El kernel utiliza varios algoritmos de para maximizar y optimizar el tiempo de ejecución del sistema. Esto introduce la idea de optimización de proceso. Optimización de proceso es un algoritmo donde cada tarea es asignada una prioridad.

Procesar la tarea entonces es llevada a cabo de acuerdo con esta asignación de prioridad. Todos los usuarios tienen derecho a manipular en decrementar la asignación de prioridad de un proceso, pero solo el root tiene derecho a incrementarlo. Aunque Talvez esto aparente caótico, funciona muy bien.

Abstracción de Arquitectura

Aunque Linux fué originalmente diseñado para arquitecturas x86, el kernel ha sido alterado en los últimos años para correr en una gran variedad de plataformas de hardware. En su mayoría el kernel ha sido escrito en lenguaje C y por eso facilita su portabilidad. Pero como el kernel debe proveer una interfase entre el hardware y los programas, existen porciones del kernel que varían entre plataformas. Estas porciones son denominadas dependientes de arquitectura, lo que significa que ellas deben ser implementadas independientemente en las diferentes arquitecturas de hardware. Las plataformas en la cual Linux ha sido porteado con éxito son la sigue:
- Intel x86 y compatibles Sistemas PC (ix86)
- Compaq (Digital) ALPHA (AIX)
- Sun SPARC y compatibles (SPARC y SPARC64)
- PowerPC, incluyendo Power Macintosh (PPC)
- Motorolla 680x0, incluyendo Macintosh de generación anterior (m68k)
- MIPS R4000 serie, incluyendo el Cubo de Cobalt y algunas SGI (Silicon Graphics)
- StrongARM, incluyendo sistemas de Acorn (arm)

Nota: esta lista solo incluye los reportados públicos pero la Comunidad Open Source reporta porteos de los famosos handhelds o PDAs, XBoxes, Sony playstation, etc, entonces esta lista no es completa, pero desde el punto de vista de un administrador de sistema quien es el lector objetivo de este libro, esas plataformas no correrán en un kernel estable.

Drivers de Dispositivos

El kernel provee acceso a todo los recursos de hardware del sistema a través de los drivers o módulos. En lo general existe un modulo o driver para que los programa tengan acceso a cada dispositivo del hardware. A menudo estos módulos son directamente compilados con el kernel, como método de optimización. En ocasiones es mejor dejar los módulos fuera del kernel. Si el kernel es diseñado para ser portable (ejemplo: desde un floppy montar root/boot) Talvez fuese mejor dejar los módulos fuera para así obtener un kernel más pequeño, ya que mientras más módulos colocas en el kernel más grande se convierte y meno es el chance de caber en un disquete. Los módulos consisten en código objeto que puede ser instalado y removido del kernel. En algunos casos, usar módulos es preferible ya que permite que hardware sea cambiada sin la necesidad de reconstrucción del kernel.

Con el crecimiento de GNU/Linux en los últimos dos años la disponibilidad de drivers para una gran variedad de hardware es ya existente. En el pasado, tomaba de meses hasta un año para nuevo hardware ser soportado. Más recientemente drivers para nuevo hardware aparece junto con la introducción del equipo. Ejemplo de esto es que ya DELL y HP, COMPAQ, IBM venden equipos nuevos con Linux, principalmente RedHat, lo cual en el pasado era imposible ya que estos nuevos equipos traen nuevos dispositivos los cuales no eran soportados y el tiempo de espera era inaceptable para las casas comercializadoras.

Sistemas de Archivos

GNU/Linux utiliza un sistema de archivos para poder organizar el almacenamiento en disco. Un sistema de archivos es una manera de organizar archivos para que ellos puedan ser luego localizable, encontrados, leídos, escritos y manejados. Existen tres tipos de sistemas de archivos:

- Local (Cintas, discos fijos, discos flexibles, etc)
- Network (NFS, Samba) (redes)
- Virtual (manipulados por el kernel, ejemplo: /proc)

Sistema de archivos Local provee acceso a archivos de los que estamos más familiarizados. El sistema de archivos más utilizado en Linux es EXT2 y rápidamente siendo remplazado por más nuevo EXT3. La mayor parte de las nuevas distribuciones de hoy utilizan EXT3 como instalación por defecto. Linux posee la capacidad de lectura de muchísimos sistemas de archivos, siempre y cuando esta capacidad sea incluida en el momento de compilar el kernel. Si no es así será entonces cargada como un módulo dinámico. Los más comunes son iso9660 para CD-ROMs, vfat para DOS y otros sistemas operativos y HFS para sistemas Macintosh.

Sistemas de archivos Network (de redes) hacen que archivos estén disponibles vía una interconexión de red. Sistemas de archivos de network poseen siempre dos partes, la parte cliente y la parte servidor. Los servidores almacenan la data en su sistema de archivos local y la hacen disponibles en la red, para que los sistemas clientes puedan tener acceso de ellos para escribir, leer y ejecutar. El server es responsable por la seguridad y la integridad de la data. Los Servers casi siempre son programas que corren externos al kernel mientras que los clientes son compilados en el kernel y se comportan como drivers o módulos. Los sistemas de archivos más comunes son NFS (Sun), SMBFS (Samba) y NCPFS (NetWare).

Sistemas de archivos Virtuales no almacenan archivos en la forma tradicional. Los archivos son una representación de espacio de memoria usado por el kernel. Ellos muy a menudo proveen de buena fuente de información con los que esta sucediendo con el kernel. El mejor ejemplo de un sistema de archivos virtual es el directorio /proc. A demás de leer información de estos archivos, podemos escribirle y así modificamos los parámetros del kernel. Dos otros ejemplos de sistemas de archivos virtuales son devpts y devfs.

Las Redes

Drivers de redes proveen soporte para dispositivos de hardware y protocolos de redes. Linux provee uno de los sistemas más rápidos y seguros disponible hoy día. Soluciones a problemas de seguridad reportados, son rápidamente presentadas. Además, la velocidad y alta compatibilidad con los otros sistemas operativos lo hace un sistema super versátil.

Aunque GNU/Linux es mejor conocido por su alto rendimiento de IP stack, otros protocolos de disponibles de alto rendimiento. Linux ha implementado Internetwork Packet Exchange/Sequence (IPX/SPX) y AppleTalk y otros protocolos más. El kernel optimizado y las capacidades avanzadas del establecimiento de una red han permitido el desarrollo de cortafuegos de paquete-filtrados y de servidores avanzados proxy. Las opciones optimizadas de routing disponibles al kernel permiten un mejoramiento incrementado en el uso de Internet.

Administración de Memoria

Una computadora tiene un monto limitado de memoria, el cual debe ser distribuido a todos los programas y procesos. El kernel esta a cargo de colocar estos recursos de memoria. El implementa sistemas de memoria virtual, lo cual le permite correr que más programas sean ejecutados en memoria que el sistema posee físicamente en sus bancos de memoria. Linux utiliza memoria compartida para comunicar mensajes de un proceso a otro. Tome el comando ps, por ejemplo. El reporta desde la información contenida en los archivos encontrados del kernel en /proc (archivos creados por el kernel en memoria).El RAM es utilizado como cache acceso a disco para incrementar E/S funcionamiento. El monto de cache es asignado dinámicamente basado en las demandas del sistema en el momento dado.

Llamadas del Sistema

Llamadas al sistema es la forma en la cual programas tienen acceso a las funcionalidades del kernel. En conjunto con las librerías del sistema, el sistema de llamadas provee una interfase con la cual comunicarse con el hardware. Este interfase es conocido como Application Programing Interface (API). Estas APIs proveen un interfase portable entre un programa y el hardware. En su mayor parte estas llamadas conforman al Portable Operating System Interface for Unix (Interfase de Sistema Operativo Portable tipo Unix) (POSIX) estándar, el cual define los sistemas operativos tipo UNiX. Casi toda operación que requiere acceso al hardware o recursos del sistema requiere una llamada al sistema desde el kernel. Esta llamada le transfiere control al kernel desde el programa, utilizando un modo privilegiado de operación. El kernel usa este modo privilegiado para ejecutar la tarea y le pasa el resultado devuelta al programa que requirió.

Estructura del Árbol del Código del Kernel

El código fuente del kernel se amacena en el directorio /usr/src/linux. A veces este es un vinculo a otro directorio, algo como /usr/src/linux-2.4.20, el cual contiene el código fuente especifico a la versión del kernel instalado. Note que algunas distribuciones no instalan el código fuente del kernel en instalación normales, si no que este paquete debe ser instalado adicionalmente.

A continuación las ramificaciones del árbol del código fuente. Las ramas son las divisiones del código del kernel que permite la personalización y optimización.

Core	Los archivos necesarios no importando la configuración elegida.
Doc	Archivos de la documentación de la compilación y opciones necesarias para la configuración
Include	Estos son vínculos a los archivos del sistema include; estos a menudo se encuentran en /usr/include y contienen parámetros específicos a la arquitectura.
Dependent	Dependiente de Arquitectura, código especifico a la arquitectura de cada maquina. (x86, AIX, etc)
Drivers	Esta sección contiene los drivers de los dispositivos que permiten al kernel controlar y comunicarse con

	el hardware instalado.
Networking	Conjunto de archivos que modifica el comportamiento y la funcionalidad del networking del server. Estos archivos controlan todo desde la optimización hasta cual protocolo son usados por el sistema.

Compilar el Kernel

¿Por qué compilarías el kernel? Algunas razones incluyen:

- Optimizarlo para un modelo de CPU particular.
- Obtener soporte para un dispositivo que no se encuentra en la imagen estándar.
- Obtener soporte para un dispositivo que no estaba disponible.
- Actualizar drivers con correcciones de patches.
- Tener acceso a una característica en un kernel más reciente.
- Remover módulos que no se están usando para optimizar uso de la memoria.

En esta sección, cubriremos los siguientes tópicos:

- Preparar el árbol del código
- Configurar el Kernel
- Compilar
- Instalar

Preparar el Arbol del Código Fuente

Antes de poder compilar el kernel, primero debes desempacar el código fuente en el sitio adecuado y preparar el directorio fuente. Esta sección asume que el uso de un kernel estándar desde el sitio WEB http://www.kernel.org. El paquete del kernel Source incluido con su distribución también puede ser utilizado, pero talvez no sea tan fácil de actualizar.

Desempaquetar y Aplicar Enmiendas (Patches)

Para desempaquetar el kernel, ejecute al similar a esto:

```
$ cd /usr/src
$ tar zxvf ~/download/linux-2.4.2.tar.gz
```

Esto desempaquetará el kernel versión 2.4.2 en el directorio /usr/src/linux.

Los paquetes del kernel son grandes y se ponen más grandes con cada nuevo lanzamiento. Para reducir el ancho de banda de las descargas, archivos de patch conteniendo solo la diferencia del kernel nuevo al anterior están disponibles. Si esta actualizando desde una versión reciente a otra el patch en vez del kernel entero debe ser descargado. Aplicarle este patch a kernel es algo así:

```
$ cd /usr/src
$ zcat ~/download/patch-2.4.3.gz | patch -p0 -N -E -s
```

Esto actualizaría el árbol del código e /usr/src/linux desde la versión 2.4.2 al 2.4.3. Note que el comando patch fracasará y le dará error si no encuentra la versión del kernel que el esperaba.

Make mrproper

Lo último que deberá hacer una ve ya desempacado el árbol fuente es decirle a la fuente del kernel que se limpie. Para lograr esto, debes darle este comando:

```
$ make mrproper
```

El árbol completo del kernel ya esta listo para ser configurado y luego compilado. Note que este paso es solo necesario si estas aplicando un patch al árbol del kernel.

Configurar el Kernel

Después que el árbol del código del kernel ha sido preparado, el kernel debe ser configurado. Esto le dice al sistema que debe ser compilado en dentro del kernel y que debe ser compilado como un modulo separado, y que se quedara fuera del kernel. Existen tres maneras de configurar el kernel:

- make config
- make menuconfig
- make Xconfig

make config

El método original de compilar el kernel. Es el único garantizado a trabajar con los kernel de desarrollo. Lo otros métodos han sido desarrollados para proveer facilidad de uso. Otro método relacionado es make oldconfig, solo hace preguntas relacionadas con opciones y utiliza las opciones de kernel anterior como guía para las anteriores.

El proceso make config te muestra las opciones por nombre con una pequeña descripción. Entonces te pregunta que indique si deseas incluir esta característica. Algunas te permiten incluirlas como módulos en vez de compilarlas directamente en el kernel. Responde cada dialogo con una Y, N o M y entonces presionas ENTER. La letra por defecto es impresa con letras mayúscula y negrita; si presionas ENTER aceptas la repuesta por defecto. También puedes digitar un (?) de pregunta para recibir ayuda. Esto imprime una pequeña ayuda. Una de la debilidades de make config es que una vez pasada la pregunta no puedes regresar ha esta pregunta. Y para corregir un error tuviese que volver a empezar el proceso entero de nuevo.

He aquí un pequeño ejemplo de una sesion de make config:

*** Loadable module support**
Enable loadable module support (CONFIG_MODULES) [Y/n/?]
Set version info on symbols for modules (CONFIG_MODVERSIONS) [Y/n/?]
Kernel daemon support (autoload of modules) (CONFIG_KERNELD) [Y/n/?]
*** General Setup**
Kernel math emulación (CONFIG_MATH_EMULATION) [Y/n/?]
Networking support (CONFIG_NET) [Y/n/?]
Limit memory to low 16MG (CONFIG_MAX_16M) [Y/n/?]
PCI bios support (CONFIG_PCI) [Y/n/?]
PCI bridge optimization (experimental) (CONFIG_PCI_OPTIMIZE) [Y/n/?]
System V IFC (CONFIG_SYSVIPC) [Y/n/?]

Recuerde que las opciones varían dependiente de la versión de kernel que este utilizando.

make menuconfig

Una manera mas amistosa de configurar el kernel es proveída por la utilidad make menuconfig. Simplemente seleccione las opciones deseadas desde el menú en la consola, estas opciones son fácil utilizar le permite retroceder a cambiar parámetros ya elegidos. También es una manera útil de guarda diferentes configuraciones para experimentar con diferentes opciones.

make xconfig

Esto es esencialmente la misma utilidad que (make menuconfig), pero ya con la interfase del ratón. Usted se debe encontrar en la interfaz grafica y tener instalado TCL/TK.

FOTOOOOOOOOOOOOO>>>>>>COMPILADOR KERNEL

Compilar

Probablemente usted, tendrá que ingresar como root, para instalar el kernel y sus módulos. Una vez el kernel ha sido desempaquetado y configurado, compilarlo es fácil. Solamente ejecute los siguientes comandos:

```
$make dep
$make clean
$make zImage (o $ make bzImage)
```

La sentencia make dep, construye la lista de dependencias para entender el orden a usar para construir las piezas. El comando make clean, elimina archivos, anteriores, de su ultima compilación. El comando make zImage (o make bzImage) es donde el verdadero trabajo empieza. Dependiendo la velocidad de su equipo puede tomar de unos de minutos hasta varias horas compilar y limpiar el nuevo kernel, este mismo puede ser encontrado en el directorio /usr/src/linux/arch/(tipo de arquitectura)/boot/ . El proceso de compilar también creo un archivo llamado system.map, que debe ser copiado al directorio /boot (este proceso es importante únicamente si estas cambiando la versión del kernel), la imagen zImage es limitada al tamaño de un mega cuando esta descomprimida, este no es el caso de bzImage. También, zImage debe ser usada en sistemas no i386.

touch

Existen momentos cuando usted acaba de ajustar el reloj del computador, que usted encontrara algunas quejas de su sistema referente a fechas de futuro. El compilador se quejara de que archivos del código tiene fecha futurística si ignora este error y permite que el compilador continua es posible que su kernel trabaje o no. El comando touch modifica atributos de fecha en un archivo así que es mejor corregirlo:

- Para corregir un archivo único invoque el comando touch seguido por el nombre del archivo:
  touch algo.h
- Para corregir un directorio completo utilice el comodín:
  touch *
- Para hacerlo recursivo para sub-directorios:
  find ./ -exec touch {} \;

Instalar

De la manera que usted instale el kernel dependerá de como usted arranca su sistema. L a mayor parte de personas utilizan grub, otros usan lilo, pero algunos aun están utilizando Loadlin y ocasionalmente talvez tenga que crear un disquete de inicio. Muchos de esto puede ser automatizado utilizando el comando make install en vez de make bzImage y zImage.

GRUB

GRUB es el Gran Gestor de Arranque Unificado de GNU (GNU GRand Unified Boot loader), un proyecto que intenta solucionar todos los problemas de arranque de una vez por todas.

Una de las características más interesantes es que no tienes que instalar una partición nueva o un núcleo nuevo, puedes cambiar todos los parámetros en el arranque usando la Consola GRUB, ya que conoce los sistemas de ficheros.Un ejemplo trabajando y editando desde la consola de GRUB es:

```
grub> map (hd0) (hd1)
grub> map (hd1) (hd0)
grub> root (hd1,0)
grub> makeactive
grub> chainloader +1
grub> boot
```

LILO

Si boteas desde el disco duro, va a ser necesario instalar y reconfigurar. Para reconfigurar lilo agregue a lilo una entrada similar a la siguiente:

```
image = /boot/vmlinuz-2.4.2-15
label = Linux
root = /dev/hda2
read - only
```

Después reinstale lilo ejecutando el comando lilo. Es conveniente e inteligente dejar la entrada del kernel anterior /etc/lilo.conf. En el caso donde la compilación del kernel este corrompida, esta entrada le puede salvar muchos dolores de cabeza al administrador ya que tu sistema aun estará funcional.

Disquete de Inicio (Arranque)

Para crear un disquete de arranque simplemente ejecute el comando make zdisc.

EJERCICIO 2-1. Reconstruir el Kernel

Este ejercicio lo guiara a través del proceso de la compilación e instalación del kernel. Aquí se configurara el kernel para añadirle el soporte de dispositivos Joystick. Este ejercicio requiere un espacio en disco duro de 100mg. y al rededor de 2h. No damos solución para este ejercicio, asegurase que usted cuente con un disquete de boteo, en caso de que su kernel no arranque. Primero asegurase que su dique inicie su máquina, este ejercicio asume que usted ha descargado la versión 2.4.2-15 y le va a aplicar un parche para actualizar a la versión 2.4.20. Usted puede usar cualquier versión que deseé. Si usted desea practicar con el código fuente de su sistema, usted puede obviar la descarga y el desempaquetamiento del kernel. Utilizar el código fuente del kernel de su distribución es una buena idea; ya que usted podrá estar mas seguro de las características con las que la cuenta:

1.- Descargue el kernel , lo puede obtener de http://www.kernel.org

2.- Identifique su kernel actual, cual esta almacenado /boot. Común mente nombrado vmlinuz y tiene la versión agregada Ej.: vmlinuz-2.4.2-15

3.- Efectué una copia de resguardo de su kernel. Asumimos que su imagen de kernel es de versión 2.4.2-15, usted utilizaria el siguiente comando:

```
# cp vmlinuz vmlinuz-2.4.2-15-bueno
```

4.- Cree una entrada en su /etc/lilo.conf para que usted pueda iniciar con su kernel ya existente en caso de que el nuevo no funcione:

```
image=/boot/vmlinuz-2.4.2-15-bueno
label=Funciona
root=/dev/hda2
read-only
```

5.- Si ya usted tiene el código fuente del kernel en /usr/src/linux , renombre el directorio

```
#mv /usr/src/linux /usr/src/linux-bueno
```

6.- Desempaquete el código fuente del nuevo kernel

```
# cd /usr/src/
# tar zxvf linux.2.4.19.tar.gz
```

Si descargo bz2Image, ejecute:

```
# tar xvfy o tar xfl
```

También puede usar el comando:
 # bzcat linux.2.4.-19.tar.bz2 | tar xvf -

7.- Si descargo algún parcho que desea aplicar:
 # zcat patch 2.4.19.gz | patch -pO -N -E -s
*****Reemplace el zcat con bzcat si descargo el patch.bz2*****

8.- Cambie al directorio del kernel recién creado:
 # cd linux

9.- Prepare el árbol del código:
 # make mrproper

10.- Configure el kernel:
 # make menuconfig 0 make xconfig 0 make config

11.- Navegue a través de las categorías seleccionando la categoría INPUT CORE SUPORT y presione ENTER; presione Y para habilitar INPUT CORE SUPORT

12.- Un listado de dispositivos de entrada aparecerá elija Joystick Support y presione Y para habilitarla selecciones EXIT y EXIT nuevamente.

13.- Navegue hacia abajo y selecciones CHARACTER DEVICES y luego seleccione Joystick.

14.- Habilite Joystick Support al presionar Y o use la barra espaciadora para selecionar YES/NO/MODULE.

15.- Una vez Joystick Support esta habilitado aparecerán más opciones específicas a dispositivos de Joystick. Elija Classic PC analog joysticks and gamepads.

16.- Salga de la utilidad menuconfig y guarde su configuración.

17.- Construya el kernel:
#make dep
#make clean
 # make bzImage

18.- Instale el kernel en lugar del anterior
 # cp arch/i386/boot/bzImage /boot/vmlinuz

19.- Ejecute lilo para decir al cargador de arranque que el kernel esta en el disco
 # /sbin/lilo

20.- Construya los módulos
 # make modules

21.- Instale los módulos
 # make modules_install

22.- Reinicie el sistema
 # /sbin/shutdown -r now

23.- Si todo ha funcionado bien usted podrá reiniciar el sistema y verificar la versión de su nuevo kernel:
 # uname -a
Linux localhost 2.4.19 #1 SMP Mar Jun 15 13:15:55 EST 2003 i386

24. Al menos que usted no este totalmente confortable con el kernel experimental de este ejercicio, restablez-

ca las copias de seguridad.

EJERCICIO 2-2. Restaurar al Kernel Anterior en Caso de Fracaso

Nota: Este ejercicio no será posible si usted no hizo una copia de seguridad en el ejercicio anterior

1.- Reinicie el sistema e intercéptelo al momento de inicio Cuando el prompt le presente Lilo: Presione la tecla ctrl. (Control) y luego presione la tecla tab para ver las posibilidades de boteo que usted tiene disponibles. Dependiendo las características puestas en el /etc/lilo.conf usted obtendrá el mensaje de Linux y linux-bueno; entonces usted digitará linux-bueno para accesar a la versión anterior del kernel viejo.
lilo: <TAB>
linux linux-bueno
lilo: linux-bueno

2.- Inicie seccion como root y restablezca su kernel anterior y reinicie. Asumiendo que la imagen de su kernel es llamada vmlinuz-2.4.2-15-bueno, usted ejecutara lo siguiente:
#cp -a /boot/vmlinuz-2.4.2-15-bueno vmlinuz

3.- Ejecuto el comando lilo:
 # /sbin/lilo

4.- Reinicie el equipo.

MODULOS

En su inicio el kernel linux era monolítico. Todo era empaquetado en una sola imagen y las características o existían compiladas dentro de la imagen o no. Eventualmente, el tamaño del kernel dificulto cargarlo y era inflexible trabajar con el, ellos corrían como si fueran parte del kernel pero eran dinámicamente agregada al kernel mientras se ejecutaba.

Lo opuesto a un kernel monolítico es uno del tipo micro kernel. Un micro kernel no contiene ningún driver compilado dentro de el. El núcleo del kernel es muy pequeño y provee un reducido número de operaciones primitivas. Drivers de dispositivos, Drivers de sistemas de archivos, Drivers de redes y a veces el manejo de memoria son manejados por programas externos al micro kernel. Los módulos de linux, proveen un buen balance entro los 2 extremos de kernel monolítico y micro kernels.

Aunque usted nunca tenga que recompilar su kernel, probablemente si tendrá que manejar módulos, para instalar y configurar dispositivos, afortunadamente, la mayoría de los módulos de sistema son automatizados.

Los siguientes tópicos se cubrirán en esta seccion:
- Compilando e Instalando
- Utilidad de los Módulos
- Configurando

Compilando e Instalando

Como el kernel, los módulos son fáciles de compilar e instalar una vez ya completado el proceso de configuración. Normalmente usted compilará los módulos inmediatamente después de compilar el kernel. Para compilar todos los módulos del kernel de los directorios /usr/src/linux, simplemente ejecute:
$ make modules

Para instalar los módulos desde la cuenta de root, ejecute:

```
# make modules_install
```

Esto instalara los módulos dentro del subdirectorio /lib/modules/versión_del_kernel. Una vez los módulos han sido instalados, configúrelo y cárguelo al kernel por demanda de necesidad.

Utilitarios de Modulos

Existen varias utilidades diseñadas para el manejo de los módulos en Linux. Ellas son utilizadas para cargar o descargar y monitorear módulos del kernel. Existe también un mecanismo para automáticamente cargar y descargar los módulos del kernel.

Comando depmod

Existen módulos que dependen de otros módulos y que no pueden ser cargadas hasta que estas otras no estén ya cargadas. Esto es llamado en Linux Dependencia de Módulos. La utilidad depmod es utilizada para el árbol de dependencia de todos los módulos en el directorio /lib/modules. Y esta información es almacenada en un archivo de nombre Modules.dep. Este es un archivo de texto, que asocia un modulo con los módulos de cuales ella depende. El siguiente es un ejemplo:

```
/lib/modules/2.4.19-5/fs/umsdos.o /lib/modules/2.4.19-5/fs/fat.o
```

La utilidad modprobe es utilzada para revisar dependencia de un driver de dispositivo. Veremos después que el instala estas primero y después las necesarias para el funcionamiento.

Es importante entender el significado del comando depmod, pero en verdad, es muy raro cuando un administrador necesite recurrir a el desde la línea de comandos, en las distribuciones modernas de hoy día. Este comando acompañado de la opción -a es muy a menudo agregado al script de inicio para que sea creado siempre y cuando sea necesario.

Comando insmod

Para instalar un modulo al kernel, utilice el programa insmod. Opciones disponibles son:

-f	Forzar aunque la versión del kernel es otra de la requerida.
-s	Salida al syslog en vez del stdout.

Siga estas opciones con parámetros del modulo. Ejemplo de estos parámetros puede ser un dirección de memoria o un número de IRQ.

Comando modprobe

Básicamente lo mismo que el insmod pero revisa interdependencias de módulos y los cargas si son requerimientos y existen.

La opción -l (o lista) combinada con la opción -t (tipo) lista todos los módulos disponibles de cierto tipo. Por ejemplo, para ver un listado de manejadores disponibles para el comando mount ejecute el siguiente comando:

```
# modprobe -l -t fs
```

Comando lsmod

El comando lsmod lista los módulos actualmente cargados por el kernel, ejemplo:

```
# /sbin/lsmod
```

Module	Size	Used by
ds	8336	1

i82365	24284	1
pcmcia_core	48192	0 [ds i82365]
slip	10876	2 (autoclean)
slhc	4432	0 (autoclean)
AppleTalk	22980	0
r128	62312	0 (unused)
dmasound	30912	0
soundcore	3428	3 [dmasound]

Las entradas marcadas autclean han sido automáticamente cargadas por el kerneld (discutido al final de esta presente sección) y serán automáticamente descargadas cuando no sean necesarias.

Comando rmmod

Este comando remueve los módulos desde el kernel en base a el nombre del módulo, no el archivo. El driver no puede estar en uso para que rmmod lo pueda remover. Una vez retirado el modulo, cualquier recurso que este utilizara del sistema (por ejemplo memoria) son retornada al manejo del kernel. La opción -a remueve todos los módulos que no están en uso actualmente por el sistema.

El Demonio kerneld

El daemon o servidor del kernel es probablemente la característica más avanzada del kernel de Linux y su sistema de módulos. Este proceso determina cuando los módulos son necesarios y los carga y descarga en un proceso automatizado. Este proceso es completamente dependiente de la asociación de dispositivos y sus drives del archivo /etc/conf.modules. En las versiones modernas el kerneld ha sido reemplazado por kmod.

Configurar Modulos para su Carga Automática

El archivo conf.modules es utilizado para crear aliases de módulos con nombres referente a dispositivos. Por ejemplo, la tarjeta de red el eth0 puede ser asociada con el modulo de rtl8139 agregando lo siguiente a los archivos modules.conf:

alias eth0 rtl8139

Parámetros pueden ser pasados al dispositivo, incluyendo líneas de configuración post-instalación o preinstalación para agregar funcionabilidad al driver. Más opciones pueden ser agregadas al driver de manera similar a través de este mismo archivo. Por ejemplo:

options eth0 full_duplex

OPTIMIZACION DEL KERNEL

La manera más directa de optimizar el kernel es a través de su configuración y recompilando. El kernel de hoy exhibe características que le permite al usuario cambiar parámetros mientras el kernel esta en ejecución. Esta característica es llamada sysctl. La manera consistente de lograr esto es a través del uso de la interfase del sistema de archivos virtual /proc/sys. Esto requiere que el kernel haya sido compilado con soporte para sistemas de archivos /proc. Esto es recomendado en todas las recopilaciones, ya que no existen razones para no hacerlo.

Actualmente, la interfase /proc/sys es la única opción si necesitas alterar el kernel. Mientras que las mayorías de los archivos en /proc son de solo lectura, algunos son escribible. Si le escribes a uno de estos archivos, el kernel tratara de interpretar lo que escribes y modificar los parámetros internos, apropiadamente. Por ejemplo, imprimiendo el contenido del siguiente archivo nos dice si los cookies TCP SYN se encuentran

habilitadas o no:

```
# cat /proc/sys/net/ipv4/tcp_syncookies
0
```

En este caso, el '0' indica que esta característica esta deshabilitada. Para habilitarla, solo debes remplazar el '0' con un '1' en este archivo:

```
# echo 1 > /proc/sys/net/ipv4/tcp_syncookies
```

Esto activara la característica. Si ahora imprimes el contenido del archivo reflejara el cambio. Pero el '1' que escribió al archivo nunca se digito allá. Debemos entender que paso aquí, lo que sucedió fue que el kernel interpretó la acción y escribió el parámetro. Y al volver a leer el archivo estamos en realidad viendo el parámetro del kernel.

Existen muchos archivos en la jerarquía del directorio /proc/sys. La mayor parte de los parámetros son medio complejos de interpretar tan simplemente como este. Puedes encontrar mas información sobre estos parámetros en la documentación del kernel en /usr/src/linux/Documentation, especialmente en el directorio sysctl. Generalmente, no debes jugar muchos con estos parámetros del kernel y solo debes cambiar esos parámetros que están en la documentación y que entiendes las consecuencias de sus cambios.

Benchmarking/Pruebas Estandár

Benchmarking es la herramienta para comparar el tiempo que un computador, equipo o sistema completo necesita para completar una tarea en comparación con otro equipo. El acto de Benchmarking produce medidas estadísticas cuantitativas la cual podemos analizar para deducir y producir opiniones sobre CPU, arquitectura, compiladores, software, etc. Dada la complejidad de los sistemas de cómputos de hoy día es difícil dar opiniones de comparación de sistema a sistema y benchmarking o sea poniendo a prueba estándar es talvez la única manera de lograr establecer una opinión educada. Cuando se inicia un proceso de benchmarking para comparación o reportar opiniones sobre un sistema, seguro que nuestras emociones van a querer intervenir. Para impedir corrumpir el análisis existen pautas que se pueden seguir:

1.- No tratre de analizar el sistema completo
2.- Analicen procesos pequeños
3.- Establezca metas claras y concisas
4.- Comprenda que entiendo absoluto es solo una meta y nunca se adquiere.
5.- Ejecute las pruebas más de una vez para asegurar su veracidad
6.- Demuestre objetividad incluyendo información externa a sus pruebas

Finalmente entienda que Benchmarking es un proceso y que después de concluido recibirá comentarios indicándole donde fallo, simplemente tendrá que empezar de nuevo y corregir los errores. Este proceso es que mejora sus conclusiones.

Ejemplos de benchmarks reconocidas de Linux son:

UnixBench	Byte Magazine CPU y file E/S
Xengine	XWindow tool
x11perf	XWindow tool Rendimiento de X
Xbench	XWindow tool para Rendimientos de Xservers
Xmark93	Similar al Xbench
Whetston	Rendimientos de WebServers
Stream	Velocidad de bus vs. Punto Flotante
CacheBench	Velocidad de Acceso a memoria vs. tamaño de data
bonnie	Rendimiento de E/S

Iozone	Rendimiento de archivos secuénciales de E/S
netperf	Analizador de Rendimiento Networking
ttcp	Medidor Ancho de banda una red punto a punto
Ping	Medidor de retardo de host a host
Perlbench	Un benchmark hecho en perl
hdparm	Una herramienta multiuso de corrección y prevención de fallas
Dga	XWindow tool programa de Xfree
Mdbench	Escrito en Fortan 77. Medidor de Memoria

Uso de Programas de Benchmark

Benchmarking puede ayudar a un administrador del sistemas a determinar como diferente versiones de un kernel de Linux, rindiera en un computador, pero, recuerde que hay otros factores que inciden como servicio, confiabilidad, costo de operación y rendimiento; todas toman parte en la decisión final.

Programas de Benchmarking básicamente se ejecutan de las 2 siguientes maneras. Una manera es medir el tiempo que un sistema recorre cierto numero de iteraciones de un segmento de código. Otra manera es medir el número de iteraciones que toma un sistema para correr segmento de código sobre un tiempo. Las calificaciones del Benchmark son medidas en Seg/iteración o iteración/seg. Debidamente, un estándar debe ser implementado para que todas las pruebas utilicen parámetros comparables. A menudo una maquina en particular es tomada como referencia con la cual su rendimiento es comparado con los otros sistemas.

Ejemplo de Programa de Benchmark

Muchas de las distribuciones de linux vienen pre-configurada con el programa hdparm ubicado en /sbin/hdparm. Para poner a prueba el primer IDE el super usuario ejecutaría el siguiente comando:

```
# /sbin/hdparm -tT /dev/hda
```

Retomando, este comando debe ser ejecutado como el operador root, el argumento -tT le dice a hdparm medir la data que rinde ese proceso al ser llevado del dispositivo (IDE) al cache. Subsiguiente indicaciones reportan resultados más y más efectivos por que la data ya esta en cache. El siguiente es una salida posible del comando:

```
# hdparm -tT
macg3:/home/miguel # /sbin/hdparm -tT /dev/hda
/dev/hda:
Timing buffer-cache read: 128 MB in 1.84 seconds = 69.57 MB/sec
Timing buffered disk reads: 64 MB in 7.09 seconds = 9.03 MB/sec
macg3:/home/miguel #
```

RESUMEN

En este capítulo, introdujimos cuestiones relacionadas con el kernel de Linux. La parta fundamental de la estructura del kernel así como los pasos básicos a compilar. Uso de módulos fue discutido. Algunos puntos claves a recodar son:

- El Kernel controla el acceso a recursos de sistema
- El Kernel viene en dos versiones Estable y Desarrollo
    - Si el segundo numero es par la versión es Estable
    - Utilice versiones estables para sistemas de producción
- El kernel maneja la programación de los procesos, abstracción de arquitectura, manejadores de dispositivos, sistemas de archivos, redes, manejo de memoria y llamadas al sistema.
- Existen varias razones por la cual usted compilaría un kernel:
    - ° Para optimizar un hardware especifico
    - ° Para corregir errores o agujeros de seguridad
    - ° Para poder soportar nuevas actualizaciones de manejadores
    - ° Para accesar nuevas características
- Compilar el kernel requieres los siguientes pasos:
    - ° Descargar el código fuente y los parches
    - ° Desempaquetar el Source y aplicar los parches
    - ° Limpiar el directorio a construir
    - ° Configurar el kernel usando una de las siguientes utilidades de configuración:
        - ° make config
        - ° make menuconfig
        - ° make xconfig
        - ° make oldconfig
- Compile el kernel con el comando make bzImage.
- Instale el kernel
- Ejecute lilo para decirle al cargador donde esta el kernel
- Los Módulos permiten que parte del kernel sea cargado solo cuando son necesitados
- La carga de módulos puede ser configurado en el archivo /etc/modules.conf
- El kernel puede ser optimizado recompilando y aplicando diferentes opciones de configuración o cambiando parámetros en el directorio /proc/sys.

Preguntas Post-Exámen

Las respuestas a estas preguntas estan en el Apendice A.

1.- Liste y describa por lo menos 2 diferentes tipos de código fuente encontrado en el árbol del kernel

2.- Describa un método de optimizar el kernel

3.- ¿Cuando se pueda dar el caso de que un administrador utilice módulos en vez de compilar en el código del kernel?

4.- ¿Cuál es la diferencia entre un zImage y un bzImage?

5.- ¿Cuando es apropiado recompilar el kernel? ¿Cuales medidas de seguridad deben ser tomadas cuando se reinicia una maquina utilizando un nuevo kernel?

CAPITULO 3

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

ADMINISTRACIÓN DE PAQUETES

OBJETIVOS

Al completar este capítulo, usted podrá:

- Compilar e Instalar programas desde el fuente
- Manejar librerías compartidas
- Poder usar el Manejador de paquetes Debian (DPKG y Apt)
- Poder usar el Manejador de paquetes RedHat (RPM)
- Poseer un conocimiento completo del comando RPM y sus opciones, en particular, las relacionadas con la instalación y consultas a los paquetes
- Estar familiarizado con los elementos básicos del fuente (*.src.rpm) de los paquetes RPM
- Entender el uso del manejador de paquetes al utilizar paquetes de softwares

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1.- ¿Qué es un paquete?

2.- ¿Nombre algunos de los paquetes más comúnes?

3.- ¿Qué es una librería compartida?

4.- ¿Si usted descargo un paquete Source como lo instalará?

INTRODUCCION

El software de linux es comúnmente distribuido en un formato llamado paquete. Un paquete es una colección de archivos combinado para formar un software. Este archivo además contiene información especial sobre el contenido, así como dependencias, las cuales son otro cualquier paquete necesario para su funcionamiento apropiado. Manejadores de paquetes como RPM de RedHat y el DEB de Debian, son utilizados para manipular sus respectivos paquetes. Una de las funciones del administrador del sistemas de linux es manejar estos paquetes ya sea instalando, actualizándolo, removiéndolo o verificándolo.

En este capítulo cubriremos los manejadores de paquetes así como los paquetes fuentes denominados tarballs (tar.gz). También cubriremos la compilación de programas desde el código fuente y manejando librerías compartidas.

Administrar los Paquetes

Esta seccion detallará como manipular paquetes en ambos formatos RPM y DEB, paquetes en formato RPM llevan la siguiente nomenclatura:

Paquete-versión-revision.arq.rpm

y los paquetes DEB utilizan la nomenclatura:

Paquete-versión-revision.deb

Para estas descripciones, paquete es el nombre del paquete, versión es el numero del contenido, revisión es la versión del distribuidor y arq el tipo de maquina que el paquete se hizo para ejecutarse. En esta seccion discutiremos los siguientes tópicos:

- Instalando paquetes
- Actualizado paquetes
- Removiendo paquetes
- Cuestionando paquetes
- Verificando paquetes RPM
- Forzando paquetes RPM
- Utilidades graficas de Manejo de paquete

Instalando Paquetes

Usted puede instalar paquetes con los comandos:

RPM: rpm -i package-x.y.z-r.i386.rpm
DEB: dpkg -i package-.x.y.z-r.deb

Ambos sistemas de paquetes mantiene una base de datos de los paquetes instalados y los archivos que le pertenecen. La base de datos puede ser cuestionada para mantener un record de todos los paquetes que existen en el sistema. Cuando un paquete es instalado, actualizado o removido. Una entrada apropiada es hecha en la base de datos.

Un paquete puede depender de otros paquetes lo cual significa que ellos requieren que otros paquetes estén instalados en orden de ejecutarse apropiadamente. Si tratas de instalar un paquete para el cual existe una dependencia no resuelta recibirás un mensaje detallando que paquetes se requieren. Por ejemplo:

 # rpm -i package-x.y.z-r.i386.rpm
Failed dependencies:
Other package is needed by package-x.y.z-r.i386.rpm

Si recibe un error de dependencia usted deberá instalar los paquetes requeridos

EJERCICIO 3-1: El Uso Básico de los RPM

En este ejercicio usted se familiarizara con el rpm y su utilidad poderosa para instalar paquetes rpm en un sistema linux. Instalar paquetes rpm es una tarea simple si el sistema soporta el uso de la utilidad rpm, instalaremos un paquete rpm llamado mpg123.4-24.i386.rpm este paquete sera descargado de ftp.redhat.com. Las repuestas a este ejercicio se encuentran en el Apéndice A.

1.- Obtener paquete

2.- Utilizando un web browser

3.- Utilizando un ftp gráfico

4.- Utilizando wget

5.- Navegue al directorio donde lo descargo

6.- Simplemente invoque el comando rpm e instale el paquete añadiendo las siguientes opciones: explicito (verbose) y que despliegue las barras de progreso.

Actualizar los Paquetes

Usted puede actualizar los paquetes con los siguientes comandos:
RPM: rpm -U package-x.y.z-r.i386.rpm
DEB: dpkg -i package-.x.y.z-r.deb

Esto reemplazara versiones anteriores con nuevas. Si no existían viejas simplemente las instalaría. RPM guardará los archivos de configuración anteriores, presentando un mensaje así:
saving /etc/package.conf as /etc/package.conf.rpmsave

Al utilizar dpkg se le preguntará una lista de opciones incluyendo la sobre-escritura de los archivos de configuración o dejarlo en el estado actual.

Los cambios a los archivos de configuración pueden que no sean compatibles con los nuevos paquetes instalados. Usted debe investigar la consecuencia antes de tomar la decisión.

Eliminar Paquetes

Desinstalar un paquete es tan simple como instalarlo y se hace asi:
RPM: rpm -e paquete
DEB: dpkg -r paquete

Observe que usamos el nombre paquete y no paquete -x.y.z-r.i386.rpm o paquete -.x.y.z-r.deb. Al remover un paquete solo digite su nombre y no las versiones ya que siempre solo habrá un solo instalado y ese debe ser el que quieres remover. Un problema seguro a enfrentar al remover paquetes es la ruptura de dependencia, al querer remover un paquete otros dependen de el para su funcionamiento. Por ejemplo:
Removing these packages would break dependencies:
paquete1 es requerido por el paquete2

Con los debs, usted puede remover los archivos de configuración de un paquete si este seguro que así lo desea. El comando dpkg --purge paquete logra el mismo objetivo que el comando dpkg -r paquete, menos

que también el primero remueve los archivos de configuración.

Cuestionar los Paquetes

Para ver si un paquete esta ya instalado, ejecute el siguiente comando:
RPM: rpm -q paquete
DEB: dpkg -q paquete

Para que paquete instalo el archivo mi_archivo, ejecute el siguiente comando:
RPM: rpm -qf /usr/bin/mi_archivo
DEB: dpkg -S /usr/bin/mi_archivo

Para ver la información de un paquete ya instalado, ejecute el siguiente comando:
RPM: rpm -qi paquete (incluya un rpm -qip paquete.rpm para paquetes no instalados)
DEB: dpkg -s paquete

Para ver una lista de los archivos que un paquete instalara, ejecute el siguiente comando:
RPM: rpm -qpl paquete-x.y.z-r.i386.rpm
DEB: dpkg --contents paquete-x.y.z-r.deb

Verificar los Paquetes RPM

Al verificar un paquete comparas la información acerca de los archivos instalados por un paquete y esos mismos archivos ya instalados. Además de esto, comparas el tamaño, MD5 SUM, permisos, tipo, dueños y grupos de cada archivo.

El comando rpm -V verifica un paquete. Tienes disponible las opciones listadas para cuestionar para especificar el paquete que deseas verificar. Un ejemplo simple es rpm -V paquete1, el cual verifica que todos los archivos el paquete paquete1 permanecen idénticos que cuando fueron originalmente instalados.

Por ejemplo, para verificar un paquete conteniendo un archivo particular:
 # rpm -Vf /bin/vi

Para verificar todos los paquetes instalados en sistema:
 # rpm -Va

Para verificar un paquete instalado con un RPM sin instalar:
 # rpm -Vp foo-1.0-1.i386.rpm

Esto te ayudaría si sospechas que tu base de datos de RPM esta corrompida:
 # rpm -qdf /usr/bin/info

Para averiguar qué documentación viene con el paquete que “posee” ese programa.

Si todo verifica correcto no habrá ninguna salida a pantalla. Si existen discrepancias serán desplegadas. El formato de la salida es una cadena de ocho caracteres, una posible “c” denotando que es un archivo de configuración y entonces el nombre del archivo. Cada uno de los 8 caracteres denota el resultado de una comparación de uno de los atributos del archivo al valor almacenado en la base de datos RPM. Un simple “.” puede significar que la prueba fue superada. Los siguientes caracteres denotan el fallido de cierta prueba:

5	MD5 checksum
S	Tamaño de Archivos
L	Vínculo Simbólico
T	Fecha de Modificación del Archivo

D	Dispositivo
U	Usuario
G	Grupo
M	Modo (incluye permisos y tipo de archivo)

EJERCICIO 3-2: Verificar la Instalación de un Paquete

Solución a este ejercicio aparecen en el Apéndice A.

1.- Utilizando rpm, identifique la versión de un paquete instalado.

2.- Utilizando rpm, identifique todos los paquetes instalados.

3.- Utilizando rpm, despliegue la información de un paquete instalado.

4.- Verifique todos los paquetes instalados utilizando la opción verify.

EJERCICIO 3-3: Verificar Ubicación de la Base de Datos

Solución a este ejercicio aparecen en el Apéndice A.

1.- Verifique la ubicación de la base de datos RPM del paquete perl utilizando el comando rpm y la combinación verify.

Forzando Paquetes

Ocasionalmente se pueden presentar dificultades al manejar archivos de paquetes. Por ejemplo, usted puede encontrar que al instalar un paquete el reclama que no tiene la aplicaciones instaladas para llenar sus dependencias pero, usted lo ha instalado desde un fuente en vez de un paquete. En situaciones donde la recomendación del manejador de paquetes no es apropiada es necesaria ignorar sus mensajes de advertencia y forzar la operación. Existen opciones cuando llega el momento de tener que forzar la instalación:

RPM: - -force, --nodeps
DEB: - -force, --ignore-depens

De nuevo opciones pueden tornarse un poco peligrosas y deben ser utilizadas solamente cuando absolutamente necesarias. Favor consulte su página man para su respectivo manejador de paquetes.

Asistentes Gráficos de Manejo de Paquetes

Hasta ahora hemos explicado el uso de las dos utilidades de manejadores de paquetes más común. Aunque no con todas las características existen muchas utilidades de manejos de paquetes con interfaces mas amistosas graficas aunque no con todas las características. Probablemente usted ha visto una de estas herramientas durante el proceso de instalación (por ejemplo dselect es parte del instalador de linux).

Existen utilidades como alien y kpakage que pueden manejar diferentes tipos de paquetes mediante una consola y el X respectivamente y también existe algunos con formatos específicos como Gnorpm para RPMs y dselect/apt para deb. SuSE utiliza su propia utilidad de manejo del sistema completo incluyendo los paquetes llamado YaST. Aunque no vamos a detallar el uso de estas utilidades existen muchas de ellas y usted debe estar al tanto de su existencia y como emplearlas Gnorpm hace manejo de paquetes simples (como instalar paquetes), y apt puede actualizar todos los paquetes instalados con un solo comando (apt-get dist upgrade).

Dselect es uno de los manejadores de paquetes que utiliza Debian. Esta es la misma herramienta que se utiliza para seleccionar paquetes durante la instalación, es un programa manejado por menú permitiéndole a usted elegir el método de acceso paquete a instalar, paquete a remover y configuración. Generalmente usted querrá navegar por cada ítem del menú en orden. La parte mas importante de la aplicación es su selección de los paquetes. Esta seccion listara todo los paquetes, permitiendo le a usted instalar o remover teclas de desplazamiento para moverse y las teclas de + y - para seleccionar y deseleccionar paquetes para hacer instalados o desinstalados. Indicadores en la mano izquierda si el paquete esta instalado o se ha si do elegido para ser instalado 0 desinstalado.

EJERCICIO 3-4: dpkg/dselect

Solución para este Ejercicio se proveen en el Apéndice A.

dpkg

1.- Instale el paquete de Debian synaptic-0.7.deb

2.- Remueva el paquete synaptic-0.7.deb Asegurase de remover todos los archivos que el paquete instalo incluyendo los archivos de configuración. *** Existe un comando para esta acción.

3.- Cuestione la base de datos listando los paquetes instalados por nombre

dselect

4.- habrá la utilidad

5.- Seleccione la opción que le permite elegir el método de acceso y elija la que le permita utilizar los CDROM

6.- Actualice la lista de posibles paquetes

7.- Instale el paquete disponible

COMPILAR PAQUETES DESDE EL FUENTE

Hay ocasiones en la cual es necesario compilar e instalar paquetes desde su código fuente existen varias razones para hacer eso; por ejemplo: talvez no exista un paquete para ese programa o talvez usted quiera o desee configurar las opciones de instalación.

Aprender a compilar software desde su fuente es una parte importante para usted convertirse en un administrador del sistema GNU/Linux; ya que talvez solo encuentre software disponible en este formato. Desarrolladores del Free Software y el OpenSource distribuyen sus programas como código fuentes y otras personas (Si la licencia lo permite) crean paquetes binarios de ellos para distribuir usted debe tener un entendimiento básico de como compilar programas desde su código fuente para si se encuentra en la necesidad de hacerlo.

Usted también se le puede requerir a usted construir su propio paquete RedHat o Debian para distribución internas para su compañía. Tocaremos este tópico brevemente y además, cubriremos los siguientes tópicos:

- Conseguir el código fuente
- Desempaquetar el tarball

- Compilar
- Instalar
- Construir su propio paquete

Obtener el Código Fuente

Código fuente puede venir de las 2 formas en un tarball y un paquete fuente.

Tarball

Típicamente, los programadores entregan sus programas en el formato comprimido tar.gz. Estos archivos tar contiene el fuente y fue creado con el comando tar, el cual es una herramienta diseñada para archivar y extraer archivos. El tarball típicamente creado contiene una tabla de contenido listando los archivos y esta tabla puede ser vista con la opción tar -t.

Paquete Fuente

En algunos casos, aplicaciones tiene ciertas restricciones que permiten su distribución en formato binario. Un ejemplo muy popular de este caso es qmail un agente de transferencia de correo. Esta aplicación contiene una licencia la cual restringe su distribución binaria y su fuente son distribuidos en formatos de paquetes rpm y deb. Un paquete fuente contiene la información y los archivos necesarios para compilar el software, crear el paquete binario desde el compilado e instalar el paquete binario.

Desempaquetar el Tarball

Usted puede encontrar archivos tar con las siguientes extensiones, .tar.gz, .tgz, tar. El punto .tar.gz y el .tgz les indica que el archivo tar ha sido comprimido con el comando gzip. La mayoría de las distribuciones incluyen una versión del comando tar que ha heredado funcionalidades del gzip, el cual se llama con la opción z, como la mayor parte de los fuentes son distribuidos en archivos gzip tar en orden para economizar espacio y tiempo de descarga todos los ejemplos que daremos en lo adelante incluirán la opción -z, si estas trabajando con un archivo que es .tar, entonces omite la z de tu comando tar.

Para desempaquetar un tar primero cambia al directorio al que tu desees que se desempaquete, entonces ejecutan el siguiente comando:

```
$ tar xfz paquete-y.x.y.g.tar.gz
```

Esto la mayoría de las veces le creara un sub-directorio de paquetes en el cual los paquetes fueron extraídos.

Compilar el Fuente

Para compilar un programa desde su fuente, cambia el directorio al cual usted desempaqueto el paquete. La mayoría de los paquetes que vas a compilar contendrán una documentación adecuada explicando como configurar y compilar el software y el primer paso es leer esta información. El primer lugar que tu querrás dar un vistazo es el README (léeme). Esto casi siempre te dará un vistazo a que es el programa y te dirá como instalarlo o buscar en otro archivo a veces llamado INSTALL (instalar) el cual te dirá como compilarlo e instalar.

La mayoría de los softwares GNU y otros softwares lanzados bajo licencias de free softwares utilizan el sistema autoconf. Lo que esto significa es que tu vas a poder correr el script de configuración y determinar que características están disponibles en tu sistema. Primero, ejecute el siguiente comando para ver si existen algunas características que usted querrá accesar:

$ ./configure --help

Esto listará las secciones del script de configuración las primeras son opciones estándares que le dirán al script donde todos los archivos serán instalados. La más importante de estas es --prefix. Es casi siempre por defecto el directorio /usr/local/ si deseas configurar el paquete para que instale en tu /home entonces ejecutarías configure con el siguiente comando:

$ ./configure --prefix = /home/nombre_de_usuario

Esta opción cerca del fin de la ayuda muchas veces diferencia de paquete en paquete, probable mente querrás leer la documentación incluida para determinar que características querrás habilitar. Pero en la mayoría de los casos el defecto es suficiente.

Una vez haya ejecutado el script de configuración y hayas determinado las características para tu sistema podrás compilar el software esto se logra de la siguiente menara:

$ make

Este proceso la mayoría de las veces toma un tiempo dependiendo del tamaño del paquete y claro la velocidad de su computador. Cuando el proceso completa usted volverá a la línea de comando sin ninguno errores reportados usted tendrá que determinar las causas no podemos cubrir aquí las causas por que son muy variadas y casi siempre son de dependencia.

En muchos paquetes existen comandos para probar que el programa se compilo correctamente este comando toma 1 de 2 formas:

$ make check
$ make test

Si algunas están disponibles correrá una seria de exámenes y reportes sobre la posibilidad de una instalación con éxito. Talvez podría hasta correr el programa antes de instalarlo asegurase de especificar que desea ejecutar el programa desde el directorio actual:

$ ./programa

Note que muchos programas no pueden ser ejecutados sin haber sido propiamente instalado.

Instalarlo

Una vez el código fuente haya sido compilado y producido un ejecutable. Para instalar el programa necesitara derecho a escritura en el directorio en el cual será instalado en la mayoría de los casos, esto significa ingresando como root. Note que solo la fase de instalación requiere acceso de root. La fase de compilación comúnmente no requiere el acceso de root y debe ser ejecutado con usuarios no privilegiados para prevenir danos. Si usted no tiene la cuenta de root, talvez sea posible instalar el paquete dentro del directorio donde usted tiene derecho a escritura, esto se puede lograr utilizando opción --prefix en la fase de configuración. Cuando es listo para instalar simplemente ejecute el siguiente comando:

 # make install

Esto copiará todo los programas y archivos asociados a su ubicación apropiada. Una vez el proceso de instalación haya completa debe probar el programa.

Construir sus Propio Paquetes

En cualquier tipo de ambiente, puede ser que se llegue a la necesidad de construir un paquete, por ejemplo: Talvez no pueda ubicar un paquete, para una pieza clave de software que necesitas instalar en muchas maquinas. En caso como este compilando el código fuente y transfiriendo los binarios manualmente a cada

maquina puede probar ser muy trabajoso. Instalación de paquetes también registraría en la base de datos del manejador de paquete el cual incrementaría la manejabilidad de software personalizado. Un conocimiento básico de como un paquete es construirlo y donde encontrar ayuda es imperativo. Sin llegar muy a fondo a los detalles complejos de la construcción de paquetes explicaremos los principios básicos que uno necesita par entender y donde encontrar documentación apropiada. He aquí una lista de pasos simples para construir un paquete binario desde la fuente:

1.- Descargue el código fuente que quiera incluir en el paquete
2.- Compile el código fuente a un binario
3.- Cree un parcho de los cambios que usted ha hecho a cualquiera de los fuentes
4.- Cree un archivo de control (dependiendo del formato del paquete que usted este usando)
5.- Construir el paquete utilizando su herramienta de manejador de paquete

RPM utiliza RedHat package manager, para construir los paquetes y Debian utiliza dpkg-deb. Ambos paquetes son muy diferentes en su estructura, pero ellos funcionan igual y requieren una preparación básica. Refiérase a las páginas man de RPM y DPKG-DEB para más detalles.

Si desea practicar puede descargar este How-To (Como) en español de como hacer sus propios rpms desde fuentes en esta dirección web: http://es.tldp.org/COMO-INSFLUG/es/pdf/RPM-Como.pdf

Librerías Compartidas

Una librería compartida es un código compartido que muchos programas pueden utilizar a la vez, el código que es común a muchos ejecutables linux, es almacenado por separado en un archivo librería. Estas librerías son típicamente vinculadas a los ejecutables en el tiempo de ejecución. Esto drásticamente recude el tamaño y ahorra disco y espacio de ram.

Cada binario utiliza librerías compartidas y utiliza al rededor de 3kb para encontrar y cargar sus librerías compartidas. Ahora que el código a sido colocado en un directorio especial como librería compartida, /lib/ld.so, toda librerías pueden utilizarlas. Esto consume menos espacio de disco y hace que la actualización sea más fácil.

En esta seccion, discutiremos los siguientes tópicos:

- Esquema de número de versiones
- Ventajas de librerías compartidas
- Desventajas de librerías compartidas
- Manejando librerías compartidas

Esquema de los Números de Versiones

Las librerías tienden a introducir nuevas características y cambian el efecto de las ya existentes, o remueven librerías anticuadas. Esto puede ser un problema para el programa utilizándolas; Que pasa si el dependiera de una características anticuada. He aquí donde entra el esquema de versiones. Uno puede categorizar los cambios que pueden ser hechos a una librería como menos 0 mayor y regirnos que un cambio menor no es permitido romper programas que están utilizando estas librerías. Usted puede ver la versión de una librería mirando su nombre: lib.try.so.1.2 tiene mayor versión 1 y menor versión 2.

Ventajas de las Librerías Compartidas

Librerías compartidas tiene las siguientes ventajas:

- Ahorra memoria y espacio en disco

• Modularizan el sistema para que los errores puedan ser arreglados en un solo lugar del programa

Desventajas de las Librerías Compartidas

Una desventaja de librerías compartidas es que requieren más paquetes. Cada librería se separa en paquetes. La interfase API de la librería no puede ser cambiada sin cambiar la mayor versión y se convierte incompatible con las anteriores versiones. Pero, nuevas llamadas de sub rutinas pueden ser agregadas sin convertirse en incompatibles. Pero note que cualquier programa utilizando las nuevas llamadas va a requerir una versión de librería que implemente esa llamada. Generalmente son mayores las ventajas que las desventajas.

Administración de las Librerías Compartidas

Después de instalar una librería compartida, tendrás que ejecutar el comando ldconfig para asegurar que el sistema va a encontrar estas nuevas librerías cuando cargues el programa. Para determinar que librerías compartidas son requeridas y usadas por un programa ejecutable utiliza el comando ldd:

```
$ ldd /sbin/ifconfig
libc.so.6 => /lib/libc.so.6 (0x40017000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

Este primer campo muestra cuan versión mayor de cada librería es requerido. El segundo campo muesta la localidad del archivo de la librería actual que esta siendo utilizando. Note que esto la mayoría de las veces será un vínculo a una versión específica del archivo de librería.

El archivo /etc/ld.so.conf contiene una lista de los directorios que serán buscado para encontrar librerías compartidas cuando los programas cargan.

RESUMEN

En este capítulo usted fue introducido a los quehaceres relacionados con los paquetes compilando programas desde el fuente y librerías. Esto incluye:

- Paquetes
    - ° Instalar
    - ° Actualizar
    - ° Cuestionar
    - ° Remover
    - ° Validar
- Utilizar tar
- Paquetes Fuente
- Compilando Programas
- Utilizando Librerías Compartidas

PREGUNTAS POST- EXAMEN

Las repuestas a estas preguntas se encuentras en el Apéndice A

1.- ¿Qué hace que un paquete sea diferente de un tarball?

2.- ¿Qué es una dependencia y que pasa si usted continua al instalar un paquete con una que no esta resuelta?

3.- ¿Usted ha encontrado un binario no familiar y deseas investigar a que paquete pertenece. ¿Cómo puede investigar esto con dpkg y rpm?

4.- ¿Qué es un paquete fuente y por que crearías un paquete fuente en vez de un paquete binario?

5.- Acabas de compilar e instalar una librería en la fuente y encuentras que las aplicaciones que la requieren aún no funcionan. ¿Cómo verificarías que las opciones la pudieron localizar y como vieras todas las librerías que ellas requieren?

CAPITULO 4

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

MANEJO DE PROCESOS

OBJETIVOS

Al completar este capítulo, usted podrá:

- Crear, monitorear y (KILL) enviar señales a los Procesos
- Modificar las prioridades de ejecución de procesos
- Identificar los niveles de ejecución y en especial los niveles reservados 0, 1, 6.
- Cambiar los niveles de ejecución y apagar o reiniciar.
- Discutir y estar en capacidad de configurar los servicios desde los niveles de ejecución
- Identificar los comandos asociados con el manejo de los procesos y sus opciones
- Discutir posibles medidas a tomar para controlar y prevenir (core dumps) aborto de aplicaciones.

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1.- ¿Cómo se puede prevenir la creación del archivo CORE cuando un programa aborta inadvertidamente?

2.- ¿Cómo puedes saber cuanto espacio físico y de swap esta en uso en la memoria del sistema?

3.- ¿Define el término Daemon?

4.- ¿Qué es buffer overflow?

INTRODUCCION

Cada programa en ejecución genera un proceso. El kernel le suministra recursos a cada proceso y una vez terminado el proceso se lo retira. En este capítulo, examinaremos procesos y los dos recursos primarios que ellos controlan: memoria y ciclos del CPU.

Discutiremos signaling (llamadas), el cual es el mecanismo utilizado para que los procesos se intercomuniquen. Explicaremos que es un Daemon, los cuales son procesos especiales del sistema que se ejecutan dentro del espacio de usuario. También cubriremos la contabilidad de los procesos, la cual es una opción del kernel que guarda la información de cada proceso ejecutado.

Este capítulo cubre procesos, señales, daemons, memoria y contabilidad de los procesos.

PROCESOS

Un proceso es nada más que un programa en ejecución. Para ser mas preciso digamos que es, el estado actual de ejecución en un dado momento. Este consiste del código del programa, los recursos usados por el programa y la información utilizada para mantener el estado actual del programa. Todo programa en ejecución tiene su propio proceso. (Algunos programas tienen más de un proceso) El kernel utiliza la información del estado del proceso para asignarle recursos, como lo es por ejemplo tiempo del CPU, memoria, archivos y E/S además de otros recursos. También utiliza esta información para determinar como cambiar a otro proceso.

A veces los procesos son referidos como tarea (task). GNU/Linux es un sistema operativo multitarea, lo cual da la impresión o ilusión de que muchos procesos se ejecutan simultáneos. El kernel se encarga de que los procesos se dividan el tiempo del CPU en pedacitos, llamados ciclos y automáticamente cambia de un proceso a otro. Este cambio es denominado “Cambio de Contexto”. Linux implementa “multitarea con derecho preferente”, lo que simplemente significa que el kernel no necesita ninguna colaboración del programa para llevarlo a cabo, si no que el simplemente forza un cambio de contexto cuando el lo considere necesario.

La información del estado del proceso es almacenada en Bloques de Control de Procesos (Process Control Blocks), a los cuales el kernel es quien los supervisa. Todos estos bloques (PCBs) son almacenados en las tablas de procesos del kernel. Cada proceso es asignado una identificación Process ID (PID) el cual es utilizado para localizarlo dentro de la tabla de los procesos del kernel. Todos los procesos, excepto el init, tienen un proceso padre (quien lo genero). El init al ser generado por el kernel al arranque no tiene proceso padre.

Y por ende entonces todos los demás procesos son hijos de algún otro proceso. La correlación familiar de los procesos puede ser representada como un árbol familiar de los procesos, con el proceso init siendo el proceso ancestral de todos los procesos iniciados desde que el equipo fué iniciado.

En esta sección, cubriremos:
- Creando Procesos
- Monitoreando Procesos
- Administrando Procesos

Crear Procesos

Un nuevo proceso es generado regularmente cada vez que se arranca un programa. A excepción de los comandos del shell (Ej.: cd, ls,...) cada programa ejecutado generará un proceso nuevo. Si conectas varios comandos utilizando las tuberías del shell, cada segmento de la tubería (cada comando) es ejecutado en procesos separados. (Más adelante veras el uso de las tuberías del shell)

Puedes utilizar el comando exec para así requerir del kernel que no genere un nuevo proceso para este programa que vas a ejecutar. Típicamente el comando exec reemplaza la imagen del proceso existente con una nueva imagen. Consideremos este ejemplo:

$ exec vi

Algo extraño pasa cuando salimos del vi , se nos cierra el shell también. Este es porque el comando exec le dijo al kernel que reemplace la imagen del proceso del shell con la nueva del vi y al nosotros salir el shell no pudo permanecer viva porque no tenia ya un PID.

El método por el cual un programa crea un nuevo proceso es llamado forking (ramificar). Forking crea un duplicado exacto del proceso, incluyendo código ejecutable, data, ambiente, variables y el timón (handle) para los archivos abiertos. El proceso hijo entonces puede reemplazar a si mismo con otro programa, si esto fuese necesario. En sistemas operativos modernos, como es el caso de Linux, los dos procesos, padre e hijo, ambos comparten la misma memoria hasta que uno o el otro se modifica; cada bloque de memoria es copiado solo después de haber cambiado. Esto es denominado copiar-cuando-escribe (Copy-on-write) y es una manera efectiva de manejar el uso de la memoria. Desde el punto de vista de los procesos, cada uno tiene su propia copia de cada cosa.

Es común que un proceso hijo que ramifico (forked) se reemplace a si mismo de inmediato con un nuevo programa cargado desde el disco. La llamada del sistema para esto es exec, la cual trabaja en esencia idéntica al comando del shell exec. El proceso carga un nuevo programa a memoria en lugar de uno ya caducado y empieza a ejecutarlo. Para iniciar un nuevo programa, el shell desde la línea de comandos ejecuta un fork y de inmediato un exec; este proceso es llamado por el nombre de fork-exec. Mas aun, el shell espera que el proceso hijo que ella creo con la secuencia fork-exec, termine para devolver el prompt de nuevo al usuario. Una alternativa a esto es enviar el proceso al background (FONDO), donde el shell no espera por el proceso que termine.

Monitorear Procesos

Hay mucha información almacenada en el PCB de cada proceso. La mayoría de esta información es solo relevante e importante para el kernel, mientras que existe otras informaciones que si son importantes para todo el sistema. Algunos de los ítems en el PCB más importantes:

- ID del Proceso (PID)
- Parent Process ID (PPID)
- Verdadero User y Group IDs (UIDs y GIDs)
- Efectivo User y Group IDs (EUIDs y EGIDs)
- Process State (Estado del Proceso)
- Signal State (Estado de la Señal)

El PID es un entero único que se usa para no perder de vista de los procesos. Los PIDs son asignados en orden numérica. Es por eso que el init, el primer proceso que se ejecuta en el sistema, es siempre el proceso numero 1. El PID más grande en Linux es el 32,767 (cual es el entero más grande que puede ser representado por 2 bytes de memoria). El proceso creado después del 32,767 obtendría el entero número 1, pero en razón de no poder repetir asignaciones para que sean únicas, un nuevo proceso no puede tener este número puesto que ya esta en uso por el init. Por esto el kernel elija el próximo numero que no sea el PID de un proceso que se esta ejecutando. El PPID es el PID del proceso padre.

La PCB contiene varios campos describiendo el dueño del proceso, por ejemplo, el verdadero User ID y el verdadero Group IDs del usuario que inicio el programa y el efectivo User ID y el efectivo Group ID del usuario cuando se inicio el programa. Estos son utilizados para determinar al cuales archivos y recursos el

proceso puede accesar. Normalmente, el verdadero IDs y el efectivo IDs son una y la misma. Serán diferentes solamente cuando un programa tenga el setuid bit habilitado o cuando un usuario haya utilizado el su para cambiarse a otro usuario. En ambos casos, el Efectivo User ID es usado para determinar privilegios de acceso al sistema.

Discutiremos el estado de los procesos y señales más adelante en este capítulo.

El Sistema de Archivos /proc

El sistema de archivos /proc es una ventana para que podamos observar el funcionamiento del kernel en memoria. Utilizándolo, uno puede extraer valiosísima información sobre los procesos ejecutándose en el sistema. En el directorio /proc existe un numero de subdirectorios correspondiente a cada PID. Dentro de estos directorios enumerados se encuentra archivos y directorios que nos dan información acerca del proceso.

Existe un directorio denominado como self, el cual apunta al directorio del proceso actual. Típicamente, el contiene la información acerca del entorno del proceso, como son el descriptor del archivo abierto, estadísticas del uso de la memoria, etc.

Estado de los Procesos

Los comandos ps y top despliegan la siguiente información del estado de los procesos:

R	Running	Ejecutándose
S	Sleeping	Durmiendo
T	Terminating	Terminando
D	Device I/O	Dispositivo E/S
Z	Zombie	Zombis

Los procesos en el estado de ejecutándose o se encuentran en ejecución en el CPU o están a punto de iniciar a correr en el CPU desde que exista la disponibilidad de acceso. Durmiendo es el estado en cual un proceso cae cuando esta en espera de que un evento suceda para el despertar y entonces continuar procesando.

Casi siempre espera por un tipo de operación de E/S. Un proceso detenido es un proceso que ha sido suspendido por el usuario. Un proceso en el estado D (Dispositivo E/S) no puede ser interrumpido; el kernel esta ocupado gestionando algún tipo de proceso de Entrada/Salida. Este estado D es muy particular Talvez la única vez que se le enfrente es en un problema de network accesando un volumen de NFS.

Procesos Zombis

Cuando un proceso hijo termina antes que el proceso padre, el kernel mantiene información acerca del proceso hijo para mantener al proceso padre informado de cosas como el estatus de salida. El proceso padre es notificado vía la señal SIGCHLD que el proceso ha terminado. El proceso padre debe indicarle al kernel que ha terminado la necesidad del acceso a la información que el kernel mantiene sobre el proceso hijo para que el kernel pueda librar la memoria asociada con este proceso hijo. El proceso padre provee esta información o indicación utilizando el sistema de señales de espera (wait system calls). Hasta que el proceso padre no indica estas llamada, un proceso hijo terminado no puede ser removido del sistema y entra en el estado ZOMBIS; el proceso esta muerto, pero no puede ser destruido.

Zombis no ocupan tiempo del CPU, pero ocupan un espacio en la tabla de los procesos y es reducido del límite de procesos colocados sobre el usuario. Un Zombis es usualmente el resultado del malfuncionamiento de un proceso padre. Aunque es desagradable tener procesos Zombis en su sistema, son difíciles de destruir. Usted debe tratar de recuperar el proceso padre, quizás enviándole una señal SIGHUP o otro tipo de

señal. Si el proceso padre de verdad se torno no responsivo (congelado), Talvez sea necesario darle fin con la señal de KILL. Si el proceso padre termina antes de los procesos hijo, el proceso init (el proceso numero 1) se apropia de el y se convierte en su padre. El init manejara apropiadamente el proceso para que el pueda ser removido desde la tabla de los procesos. En algunos casos un proceso Zombis puede estar huérfano. Esto sucede cuando un proceso no es heredado por ningún otro proceso.

Administrar Procesos

Ya que hemos visto como crear procesos y como observar su comportamiento, echaremos un vistazo a que podemos hacer a un proceso ya en ejecución. Puedes interactuar con un proceso en ejecución, cambiándole su prioridad, pausarlo o resumirlo y enviarle señales. Manejo de señales aun nos falta por cubrir en este capítulo el resto empezamos ahora.

Prioridades

El kernel utiliza un sistema de asignación de un valor numérico de enteros a los procesos para poder administrar el tiempo de CPU de cada proceso. Cada proceso tiene dos valores de prioridad: estático y dinámico. La prioridad estática, mejor conocida como niceness no cambia al menos que el usuario no indique que debe cambiar. La prioridad dinámica esta basada en la estática, pero el kernel la cambia para distribuir mas adecuadamente el tiempo del CPU. Si un proceso esta utilizando todo su tiempo asignado de CPU, el kernel modificara la prioridad dinámica para que otros procesos puedan adquirir más tiempo de CPU. Cuando hablamos de prioridad, estamos casi siempre hablando de prioridad estática ya que no tenemos control sobre la dinámica.

Prioridades se definen por un valor numérico, que tienen un rango desde negativo 20 hasta positivo 20 (-20 hasta +20). La mayor parte de los programas corren con una prioridad de cero (0). Procesos con prioridades de menor valor tienen prioridades más altas, ellos tienen el primer chance a correr que los otros. Elevándole el niceness, incrementándole el valor de su prioridad y bajándole la prioridad todo esto significa lo mismo. Esta manera del manejo de prioridades sabemos que torna la discusión medio confusa, pero después todo será más claro. Pero antes de aclarar para confundir más las cosas el kernel tiene una manera de expresar el manejo de sus prioridades dinámicas comparado con algunas utilidades y herramientas. Lo básico a recordar de prioridades es que los números más bajos reciben mayor tiempo de CPU.

Los procesos heredan su prioridad de su proceso padre. Para iniciar un comando con un niceness diferente al de su proceso padre, en este caso el shell si ejecutamos desde un terminal, debemos utilizar el comando nice:

```
$ nice xload &
$ nice -5 top
```

Por defecto, nice eleva el niceness por un valor de 10, pero puedes especificar cualquier valor que desees cambiar el niceness. Solo el superusuario puede bajar, reducir el niceness de un proceso ejecutando el siguiente comando:

```
# nice --5 /sbin/fsck.ext3
```

Para cambiar el niceness de un proceso ya en ejecución, utilice el comando renice y especifique el PID del proceso que deseas cambiarle su prioridad:

```
root@gnome2:/home/gnome2# ps ax
PID        TTY      STAT   TIME   COMMAND
1          ?        S      0:02   init
2          ?        SW     0:00   [keventd]
3          ?        SWN    0:00   [ksoftirqd_CPU0]
```

```
572         pts/0    S      0:00    -bash
585         pts/0    S      0:00    bash
455         tty2     S      0:00    /sbin/getty 38400 tty2
456         tty3     S      0:00    /sbin/getty 38400 tty3

$ renice 2 572
# renice -5 585
```

Hay que notar dos cosas en la diferencia entre nice y renice. Primero, nice requiere que especifiques el cambio de prioridad, donde renice requiere que le especifiques la prioridad con la cual deseas ejecutar el programa. Segundo, nice requiere un menos (-) antes del valor y dos menos para valores negativos; renice no utiliza los menos, excepto claro para los valores negativos.

Control de Trabajos (JOBS)

El Control de trabajo es un mecanismo que le permite ejecutar más de un trabajo desde un terminal. En cualquier momento dado, existirá solo un trabajo corriendo delante (foreground), pero pueden existir varios procesos ejecutándose detrás (background). El proceso ejecutándose en el primer plano (foreground) es aquel cual tiene acceso al teclado y la salida del terminal. Los que se ejecutan en segundo plano (background) no tienen este acceso, inclusive si uno de ellos necesita alguna entrada desde por ejemplo el teclado su ejecución se detendrá hasta que reciban esta entrada o repuesta. Puedes hacer que programas en el background no escriban a la pantalla con el comando stty tostop.

Normalmente un programa corre en el primer plano (foreground). Para que se ejecute en el segundo plano (background), agréguele una & al final, por ejemplo:

```
$ grep “Linux” /var/log/dmesg > refiere-Linux.txt &
```

El shell responderá de la siguiente manera:

```
[1] 1234
```

El número 1 en las llaves es su job number (numero de trabajo). El número 1234 afuera es su PID. Tú puedes ver los trabajos corriendo con el comando jobs:

```
$ jobs
[1]+ Running         grep “Linux” /var/log/dmesg > refiere-Linux.txt &
```

Una vez este trabajo complete, el shell responderá con la siguiente salida a pantalla:

```
[1]+ Done            grep “Linux” /var/log/dmesg > refiere-Linux.txt &
```

Puedes colocar un trabajo desde el segundo plano (background) al primer plano (foreground) con el comando fg. Hay varias manera de como especificar cual de los trabajos en segundo plano deseas mover al primer plano. La mayoría de las veces solo tendrás un trabajo corriendo en el background y solo ejecutaras el comando fg sin argumentos. Pero también usted puede especificar el número de trabajo (job number).

Algunos shells requieren un % que presida el numero de trabajo, esto es opcional en el shell bash. Puedes especificar el nombre del comando del trabajo. Si tienes varios trabajos en el segundo plano que se ejecutaron con el mismo comando, puedes usar el símbolo %, seguido por uno de (?), seguido por parte del texto que no sea ambiguo. También otra versión requiere el símbolo de porciento seguido por el job number. Todos estos son equivalente, dado la salida de los trabajos anteriormente:

```
$ fg
$ fg 1
$ fg %1
$ fg % grep
```

```
$ fg %?dmesg
$ %1
```

También puedes mover un trabajo desde el primer plano hacia el segundo. Primero debes suspenderlo con CONTRL+Z. Entonces usa el comando bg para colocar el trabajo suspendido en el background. Esto empezará a correr el trabajo pero ya en el segundo plano. (Ya podrías como dijimos anteriormente traerlo al primer plano con el comando fg). Si al ejecutar el comando se lo olvido ponerle un &, esta es la manera mas fácil de ejecutar un programa en el background, para poder seguir ejecutando en el shell. Para enviar un trabajo al background puedes usar la misma notación del fg con el bg. Una manera rápida de mandar el trabajo número 1 al background es:

```
$ %1 &
```

Normalmente al salir del shell, todos los trabajos el background se les manda la señal SIGHUP para detenerlos. A veces deseas que un programa continué ejecutándose después que hayas terminado tu sesión (logout). El comando para lograr esto es nohup. Este comando mantiene los procesos en ejecución después que sus padres dejan de existir.

```
$ nohup dpkg -l >> instalados.txt &
$ exit
```

Un proceso ejecutado bajo el nohup guarda su salida a un archivo llamado nohup.out en el directorio actual.

EJERCICIO 4-1: Procesos

Ejecute el comando ps varias veces y note como el PID se va aumentado. Después corra este comando varias veces:

```
$ ps aux | grep ps
```

Note que la mayoria de las veces verá ambos procesos el de ps y el de grep, pero puede pasar dependiendo la velocidad de su maquina que solo vea el de ps porque el de grep no tubo tiempo para entrar en la tabla de los procesos.

No se da solución a este ejercicio.

SEÑALES

Señales es un mecanismo que provee el kernel para que los procesos se intercomuniquen. Existe un número predeterminado de señales (normalmente 32) disponibles, así es que la intercomunicación es bien simple. Podemos también ver las señales como un tipo de información de sentinelas; ellas interrumpen procesos para corregir situaciones anormales. Ellas tienen significados predefinidos, algunas pueden ser programadas. A estas se les refiere como manejadoras de señales (signal handlers). Muchas de las señales son usadas para proveer información a los procesos desde el kernel.

Programas pueden construir un handler para ser ejecutado cuando una señal en particular es recibida. El handler también es referido como una trampa de señales. El actual manejo de atrapar la señal es llamado catching the signal (capturar la señal). Un programa puede decirle al kernel que desea ignorar una señal en particular. Si esta señal no es capturada o ignorada, la acción por defecto tomará lugar cuando el proceso la recibe. Dos señales jamás podrán ser ignoradas o atrapas que son: **STOP** y **KILL**.

La siguiente tabla lista las señales más comunes, con su número, nombre, significado. Tú puedes usar el comando kill -l para obtener un listado completo de todos los nombres de señales y números:

Nombre	Señal	Descripción
HUP	[1]	Señal de hungup enviada a procesos hijos cuando un termina o un MODEM cierra, también enviado a algunos daemons para que vuelvan a leer su archivo de configuración
INT	[2]	Interrumpe cuando tu presionas CONTROL+C
QUIT	[3]	Sale y arroja un archivo de core; enviado al presionar CONTROL+\
BUS	[7]	Error de BUS; casi siempre un error causado por el programa o por un hardware que falla.
FPE	[8]	Excepción de punto flotante; enviado por el kernel para indicar división por cero y errores similares.
KILL	[9]	kill; detiene el programa inmediatamente y no puede ser ignorada o atrapada.
SEGV	[11]	Violación de segmento; el programa trata de tener acceso a una dirección de memoria de la cual el no es propietario; usualmente esto sucede con un error del programa o un chip de ram dañado.
TERM	[15]	Termina; finaliza el programa adecuadamente (si es posible); señal enviada por defecto por el comando kill.
STOP	[17]	Detiene inmediatamente; pero puede ser continuado por la señal CONT.
TSTP	[18]	Detiene definitivamente; el programa solicita una entrada pero no tiene control del tty; también puede pasar igual al presionar CONTROL+Z; llamada suspender (SUSP)
CONT	[19]	Continuar; usado para resumir o detener un proceso;
CHILD	[20]	Proceso hijo sale; enviado al proceso padre cuando el proceso hijo a terminado.

Puedes enviar algunas señales a programas corriendo con el uso de teclas especiales aun terminal. Para enviar la señal TSTP, presiona CONTROL+Z. esto normalmente suspenderá la ejecución y te retornara al promp de shell. Para enviar la señal INT, presione CONTROL+C. Normalmente esto terminar un programa. Para enviar la señal QUIT, presione CONTROL+\ . Esto normalmente terminar el programa y generara un coredump. Usted puede usar el programa stty para cambiar cual tecla generan estas señales. Usted puede ver cuales teclas generan estas señales en su computador con el comando:

$ stty -a

Fallas relacionadas con el hardware también causan señales como las siguientes:

- Bus (errores de memoría)
- División por cero
- SEGV

Una de las maneras que los procesos de Linux se pueden comunicar es a través de señales. Existen un número de señales estándar, las cuales mas adelante describiremos. Cada proceso puede definir su propio sistema de manejo de señales para determinar que hacer. Típicamente la acción defecto es por terminar el proceso o ignorar la señal. Una señal es un evento puesto en marcha una vez para el proceso que recibe. Solamente cuando el proceso ha dado servicio al evento se reinicia. Múltiples ocurrencias de la misma señal solamente pueden causar una instancia del código de manejo de señales del proceso si son generadas muy seguidas. Las señales despertaran un proceso durmiente o en espera

El uso de señales es un método de alto nivel para interrumpir la ejecución de un proceso. Administradores de sistemas a menudo utilizan señales para eliminar procesos.

DAEMONS

Daemons son procesos que dan soporte a servicios del sistema. Normalmente ellos se inician al tiempo de arranque y la entrada de modo multi usuario y se detienen cuando el sistema se apaga. Un atributo clave de un proceso Daemons es que no es parte de un terminal esto se puede comprobar con el comando ps el cual te listar una marca de pregunta en la columna TTY. Esto implica que el Daemons no recibe señales asociadas con el terminal. Típicamente los procesos que reciben una señal SIGTERM cuando la terminal se apagan. En el caso de los Daemons tal cosa no puede pasar.

Típicamente los Daemons esperan a un evento así como a una señal, un archivo siendo creado, un time out o entrada de datos desde una conexión de red. Cuando el evento ocurre el Daemons se despierta le da servicio al evento y vuelve a dormir a menudo el demonio da a nacer un proceso hijo para que maneje el evento para que el daemons pueda escuchar por otro evento. Todos los procesos que están corriendo en el sistema pueden ser observados en el comando ps usando las opciones -e o -A, o como venga al caso cualquier proceso sin una terminal controlándolo pueden ser vista con el siguiente comando:

$ ps -t ?

La opción -t del comando ps despliega todos los procesos ejecutándose sin terminal. En el caso de lo Daemons, al no tener un terminal controlado se notan con el símbolo de pregunta por eso el comando previo desplegará todos los Daemons ejecutándolo.

MEMORIA

Los Sistemas de memoria virtual son implementados en todos los sistemas operativos multitarea modernos, como una función definida en el kernel y aunque similar a otros UNiX y parecidos-a- UNiX, esto es único en Linux.

En esta sección nosotros vamos a cubrir los siguientes tópicos:

- Memoria Virtual
- Usos de la Memoria

Memoría Virtual

Linux implementa un sistema de memoria virtual. Cada proceso ocupa un espacio de memoria virtual que le aparenta que fuese el único programa en ejecución en el computador. Cada proceso tiene su propio panorama de la memoria. El Sistema Operativo lo hace que lo vea bien. El proceso puede pretender que el es el único ejecutándose en la maquina. Un programa no puede ocupar el espacio de memoria de otro programa. Este proceso es denominado protección de memoria.

Sistemas GNU/Linux utilizan paged virtual memory como su método del manejo y administración de la memoria del computador y para compartirla entre los procesos. Cada proceso tiene su propia vista de la memoria que puede acceder y actualizar. Esto es llamado memoria virtual. Direcciones de estas memorias se llaman virtual addresses. Cada proceso individual tiene el mismo conjunto de direcciones virtuales disponibles. Por ejemplo, cada proceso empieza en la dirección virtual ‘0’ y puede utilizar direcciones hasta un valor máximo.

El mapeo virtual al físico es obtenido a través de la división de la memoria en un numero de pedacitos relativamente pequeños llamados pages (paginas). Paginas físicas correspondiente a paginas virtuales no tienen que ser necesariamente continua y ni siquiera en cierto orden. Es hasta posible que una página virtual no este en memoria física. Si no ha sido accesada por cierto tiempo, una pagina de memoria puede ser copiado a un área especial de disco llamada swap (memoria virtual), permitiendo que la memoria física sea utilizada por otro proceso. Si un proceso intenta accesar una página de memoria que no esta en memoria ocurre lo llamado page fault (falla de pagina) y la pagina puede ser copiada a la memoria para permitir al proceso continuar.

Al tratar la memoria como una secuencia de pagina, el kernel puede hacer eficiente uso de la memoria disponible en la maquina. Pero, existen momento cuando hay tan poco espacio de memoria y los procesos se dilatan un tiempo largo e inaceptable para que el espacio en memoria este disponible. En este caso, el sistema hará uso del swap y bajara procesos desde la memoria al disco, así liberando múltiple página. Linux puede

utilizar espacio de disco para actuar como ram. El mueve áreas de memoria que no han sido recientemente utilizadas al disco duro. Cuando esa área de memoria es accesadas de nuevo, entonces se retorna del disco a la memoria ram, para que este disponible a su uso. Mientras el sistema esta en el proceso de mover la data de nuevo a la memoria, puede ejecutar otros procesos, temporalmente suspendiendo el proceso mientras la memoria retorna a ram.

En GNU/Linux el área de swap consiste de dos categorías: una partición en un disco o un archivo en un sistema de archivos existente. Linux soporta hasta 8 áreas de swap, cada una de ella puede tener hasta 2 giga de espacio. Por esto, Linux soporta un total de 16 GB de área de swap, la cual puede ser optimizada moviendo el área a un disco más rápido.

Uso de la Memoria

El estado actual del uso de la memoria puede arrojar información vital para determinar que esta pasando en el sistema, que esta presentando un comportamiento lento. Existen dos comando GNU, free y vmstat, permitiendo al administrador examinar el uso de la memoria.

Comando free

Despliega la cantidad libre y la utilizada, de la memoria swap y de la física en el sistema, así como la memoria compartida y los buffers utilizados por el kernel, lo siguiente es una salida del comando freeLaplicación en GNU/Linux.

Comando vmstat

El comando vmstat se utiliza para mostrar estadística pertinente al sistema virtual de memoria. El vmstat reporta información acerca de los procesos, memoria, paginación, bloques de E/S, traps (compuerta), y actividad de CPU. El primer reporte produce average desde el último arranque del sistema. Reportes adicionales dan información de un periódo de de tiempo específico. Los reportes de procesos y memorias son instantáneos en cada caso. Vmstat, ejemplo GNU/Linux.

```
[root@proxy-ap root]# vmstat
procs                  memory      swap    io     system         cpu
r b w      swpd   free   buff  cache   si so bi  bo  in   cs  us  sy  id
1 0 0     11916    656   9404  16064    0 2  9  31  153  140  1   1  98
```

Comando memstat

El comando memstat, es otro que se puede utilizar para determinar el uso de la memoria virtual al reportar el uso de la memoria virtual, memstat te retorna los procesos en ejecución, librería compartido y el total de la memoria virtual en uso.

Archivos Core

Son archivos de vaciados de memoria de un proceso cuando el programa fracasa o falla. Este archivo lo escribe, el kernel al disco al ocurrir una falla. Una de la razones para que el kernel genere este archivo es violación de memoria causada por un programa. Violación de memoria ocurre cuando un programa trata de escribir o leer desde la memoria a la cual el no tiene acceso, esto causa una falla de segmentación.

También pueden ocurrir debido a errores de bus y excepciones de punto flotante, este archivo puede ser utilizado con gdb para examinar el estado del programa y determinar su causa de falla.

Comando ulimit

El comando ulimit se utiliza para limitar el monto de recursos que un usuario o proceso puede tener acceso. El uso común de ulimit es para los usuarios en su script de inicio para prevenir que se creen archivos core.

$ ulimit -c 0

La opción -c coloca el tamaño máximo de archivos core y lo coloca en cero o sea que nunca se han creado. Usted puede limitar otros recursos así como memoria virtual (-v), Tiempo por proceso del CPU (-t), y el numero de proceso que un usuario puede ejecutar a un mismo tiempo (-u). Si un programa trata de exceder cualquiera de este límite saldrá con un mensaje de error.

REGISTRO DE PROCESO

Registro de proceso es una característica opcional del kernel que registra la información de cada proceso que esta siendo utilizado en el sistema el mantiene información detallada sobre cada ingreso del usuario, uso del CPU, uso de la memoria, transferencia de E/S y nombre de los comandos. Esta información se escribe a un archivo de registro cada vez que un proceso sale, en esta seccion discutiremos lo siguiente:

- Habilitar registro de procesos
- Revisar información de los registros

Habilitar el Registro de los Procesos

El kernel es la fuente de toda la información que se acumula acerca del registro de los procesos; y por eso, el primer paso es habilitar registro de proceso para asegurarse que el kernel tiene esta característica habilitada. Algunas distribuciones ya vienen en su kernel estándar con esta característica habilitado. Esta característica se llama BSD Process Accounting.

Una vez el kernel esta en capacidad de registrar proceso necesitara instalar software del manejo de registro de proceso, el software esta disponible en el GNU en un paquete llamado ACCT .

Registro de proceso no almacenan un archivo si este no existe. Y por esto necesitará asegurarse que el archivo de registro este presente, sino tendrá que crearlo. Si no ha sido creado cree un archivo vacío y determina quien tú quiere que pueda leerlo.

```
# touch /var/log/acct
# chown root /var/log/acct
# chmod 0600
```

Talvez quiera permitir a algunos usuarios acceso a lectura del archivo, para que ellos puedan utilizar la herramienta de registro. Debe hacer esto especificando el grupo dueño y una mascara de 0640 en vez de 0600. Note que para una seguridad mas estricta, usted no querrá que los usuarios tengan acceso a toda la información disponible. También tendrá que revisar los permisos en los archivos /var/log/savacct y a /var/log/usracct, lo cual es utilizado como información cache para el comando sa. Tengan cuidado que algunas distribuciones pueda ser que coloquen estos archivos de registro en /var/adm.

Después de haber instalado el paquete de registro de proceso talvez tenga que iniciarlo. Esto es causado por el alto costo de recursos que este proceso de contabilidad produce. Para empezar este proceso ejecuta:

```
# /usr/sbin/accton /var/log/acct
```

Talvez quisiera modificar los scripts de inicio para activar este proceso cada vez que encienda el computador. Algunos comando que viene con el paquete (notablemente el comando (ac y last), actualmente usa el archivo /var/log/wtmp para guardar su información. Este archivo no es manejado por el kernel, pero es escrito por cualquier programa que puede iniciar el script de login.

Revisar la Información de Registro

Una vez haya encendido el registro de procesos, podrá analizar la información que este genera. Existen varios comandos disponibles para lograr esto, Ellos presentan la información en manera diferente,

Comando ac

El comando ac se utiliza para mostrar cuanto tiempo ha durado la seccion de un usuario en el sistema (ac despliega esta información en horas), se puede dar la información por días y por usuarios. Sin ninguna opción el comando muestra el total de hora utilizada por todos los usuarios logueado en el sistema, como se muestra abajo:

```
$ ac
total        179.93

$ ac ivelis cristhian
total        96.35
```

Para mostrar el total de hora logueado por cada usuario utilice la sintaxis siguiente:

```
$ ac -p
ivelis       79.93
cristhian    32.75
miguel       28.99
total        141.67
```

Parta mostrar el uso diario para un usuario de nombre ivelis, use la siguientes sintaxis:

```
$ ac -d ivelis
Jan          total    9.93
Jan          total    2.75
Jan          total    8.99
Today        total    11.67
```

Un problema con el comando ac es que si un usuario ha ingresado en más de un terminal a la vez el tiempo se sumara a un total de ambas terminales y ac mostrara el tiempo total, esto también es un problema con usuario que utilizan xterm localmente y usuario que ingresan en varios secciones de telnet al mismo tiempo.

Comando last

El comando last muestra el tiempo especifico que los usuarios entraron y salieron del sistema. Esta lista puede ser extensa si existen muchos usuarios en el sistema. También podemos a traves de argumentos pasarle el nombre del usuario que nos interesa en particular. Aquí les doy dos ejemplos:

```
[root@www /root]# last
root         pts/0    pc-00069          Sat                Jul 19 09:51 still logged in
reboot       system   boot              2.2.19-7.0.8 Sat   Jul 19 09:25 (00:45)
reboot       system   boot              2.2.19-7.0.8 Fri   Jul 18 23:34 (10:35)
root         pts/0    pc-00079          Fri                Jul 18 19:24 - 20:51 (01:27)
root         pts/0    pc-00079          Fri                Jul 18 18:46 - 18:47 (00:00)
admin        pts/0    pc-00079          Fri                Jul 18 18:44 - 18:45 (00:00)
reboot       system   boot 2.2.19-7.0.8 Fri                Jul 18 17:58 (16:12)
reboot       system   boot 2.2.19-7.0.8 Fri                Jul 18 08:21 (1+01:49)
wtmp         begins   Mon Nov 1 14:09

[root@www /root]# last admin
admin        pts/0    pc-00079          Fri      Jul      18       18:44 - 18:45      (00:00)
wtmp begins Mon Nov 1 14:09
```

Comando sa

El comando sa nos proporciona un resumen de la información en base a los programas ejecutandose. Por defecto, muestra cuantas veces el programa fué ejecutado, cual fué el tiempo que se ejecutó realmente, cuanto tiempo del CPU consumió, cuanto I/O consumió este, el average del uso de la memoria y el nombre del programa.

```
[root@www /root]# sa | head -4
2622        967.43re        0.12cp   ivelis    450k
9           264.43re        0.04cp   ivelise   547k    sh
21          262.09re        0.01cp   ivelis    305k    vi
33          40.35re         0.00cp   ivelis    365k    cat
```

En este comando se desplegán las primeras 4 lineas de salida del comando sa. Observe que el orden de salida es por el tiempo del CPU. Existen varias otras opciones, como son --sort-real-time y --sort-num-calls. La primera fila es simplemente el total de todos los campos. Otras opciones también disponibles son:

--print-users	Imprime un listado de todos los comandos ejecutados y por quien.
--user-summary	Combina todos los programas usados por cada usuario.
-c, --percentages	Imprime el porcentaje de los valores del tiempo total del comando de su usuario, sistema y tiempo real.
-n (--sort-num-calls)	Ordena por el número de llamadas
-k:	Ordena por el average del uso de memoria
-t, --print-ratio	Basado en cada entrada, imprime el average de tiempo real a la suma de sistema y tiempos del usuario; si la suma del sistema y el tiempo del usuario es demasiado pequeño para reportar (su suma es cero), se imprimira "*ignore*" en este campo.

Comando lastcomm

El comando lastcomm nos brinda información de los últimos comandos que los usuarios han ejecutado. Si ejecutamos lastcomm sin ningún argumente este simplemente nos imprimirá a pantalla todos los comandos en el archivo de record en /var/log/pacct. Por ejemplo para verificar cual usuario ejecuto el comando find y ver cual usuario estuvo ingresado el tty0, escriba:

```
$ lastcomm find tty0
```

Esta sentencia imprimirá a pantalla cualquier entrada que iguale find o tty0 en cualquier de los campos (comando, nombre 0 terminal). Si desea sólo encontrar los item que igualen *all* de los argumentos en la línea de comandos, deberá usar la opción -strict-match. Por ejemplo para listar todas las ejecuciones del comando ls usadas por el usuario root en el terminal tty1, escriba:

```
$ lastcomm --strict-match ls root tty1
```

El orden como se escribe la sentencia es importante.

EJERCICIO 4-2: Modificar Valores en /proc

Para este ejercicio no se dan soluciones.

El sistema de archivos /proc es una herramienta excelente para monitorear el sistema directamente. En este directorio /proc existen directorios de nombre numéricos. Estos nombres se refieren a los PIDs de los procesos en ejecución y los archivos dentro de estos directorios contienen información correspondiente al proceso. Los archivos con nombres no numéricos corresponden a procesos del sistema y por ende con información pertinente al sistema mismo.

Los archivos representando procesos no tienen permiso de Escritura (Write). Comúnmente solo los archivos que contienen información de configuración pueden ser alterados. Recuerde que esos archivos son utilizados y creados directamente por el kernel y modificarlo puede causar comportamiento impredecible.

El directorio /proa/sys en particular contiene información de la configuración del sistema. Por esto infor-

mación del sistema puede ser directamente alterada a través de estos archivos. Vamos a dar un pequeño ejemplo de una situación que se puede dar en un servidor de páginas web, que en la compilación de su kernel se dio un número de handles que directamente afecta el número de archivos que puede abrir. Bueno este limite se esta viendo amenazado por el volumen alto de visitantes a las paginas web que esta sirviendo.

Ahora la pregunta es ¿Qué podemos hacer? La respuesta es que podemos modificar el archivo /proc/sys/fs/file-max el cual nos dice por ejemplo:

```
[root@www /]# cat /proc/sys/fs/file-max
4096

[root@www /]# echo Nuevo _ valor /proc/sys/fs/file-max
```

1.- Determine en su equipo cual es el valor actual y triplíquelo. Luego desvuélvalo a su estado original.

NIVELES DE EJECUCION (RUNLEVELS)

El kernel ejecuta init al arrancar. Este programa, ahora como proceso, cargará los subprocesos necesarios para la puesta en marcha del sistema. Cuando init ha terminado de cargarse vacía el subdirectorio /tmp y lanza getty que se encarga de permitir hacer login en el sistema a los usuarios.

Los niveles de ejecución (runlevel), definidos en el archivo /etc/inittab determinan los servicios que tendremos disponibles en cada uno de ellos. Es una forma de tener diferentes modos de trabajo, cada uno de ellos con características bien definidas, en función del tipo de tarea a que estén orientados.

Existen ocho niveles de ejecución, que están numerados del cero al seis, más otro denominado con la letra "S" (tiene un alias con la letra "s", que realmente es igual a el nº 1). Los niveles de ejecución de ejecución que manejaremos y una descripción de para qué están definidos se puede ver en la siguiente tabla:

Nivel de ejecución	Modo
0	Detener el sistema
1	Mono usuario, sin soporte de red, Mantenimiento
2	Multiusuario, sin soporte de red, Poco sistemas lo usan
3	Modo multiusuario completo
4	Sin uso. Recomendado para pruebas
5	Modo X11; Multiusuario completo en entorno gráfico
6	Reiniciar el sistema
s/S	Nivel Usuario Unico
a/b/c	Pseudo Estados (Casi nunca usado)

Init necesita un archivo de configuración para saber exactamente lo que tiene que hacer. Este archivo es /etc/inittab y contiene información sobre el nivel a ejecutar por defecto, previsión sobre lo que hacer ante determinadas situaciones, describe qué procesos se inician en la carga y durante la operación normal.

Utilice /sbin/runlevel para ubicarse en el actual runlevel que se encuentra:

```
# /sbin/runlevel
N 5
# shutdown -r -t 60
clean reboot in 60 seconds
```

En el modo de single-user, solo /, /var y /boot son montados y solo procesos esenciales son ejecutados. Como cambiar de runlevels será discutido en la próxima seccion.

Cambiar de Runlevel

Se utilizan los comandos telinit o INIT para cambiar de runlevels. Ejemplo Talvez quisiese cambiar de single-user a multiuser o vice-versa. También puedes utilizar el comando shutdown para detener el sistema. Este le avisa a los demás usuarios que el sistema esta a punto de apagarse. Aquí unos ejemplos del comando init:

 # init 2
Cambiar al runlevel 2

 # init 5
Ahora cambiamos al runlevel 5

 # shutdown
Apagar el sistema

Es de extrema importancia que nunca apagues tu sistema de linux incorrectamente. Las razones son muchas pero aquí unas cuantas:

- Si es un server pueden haber usuarios accesando la red
- Pueden haber trabajos en el background o segundo plano
- Actividad de disco de E/S y los buffered no sean transferidos lo cual corrompe el sistema de archivos

El proceso de init gerencia todo el sistema Linux. El lee información desde un archivo de control llamado /etc/inittab del cual usa la información para manejar los diferentes niveles de ejecución. Init también se utiliza para cambiar entre los diferentes runlevels. El /etc/inittab se utiliza para determinar cuales comandos ejecutar en cuales niveles. Los cambios de niveles son controlados por los archivos (rc) que ejecutan los diferentes comandos de ese nivel. El archivo rc a ejecutar en cada nivel esta definido en el /etc/inittab.

El comando init acepta parámetros que es numérico indicando el nivel de ejecución a iniciar o uno de los siguientes:

init	**s**	Modo de Single-user (También S es reconocida)
init	**q**	Leer nuevamente el archivo /etc/inittab (También Q es reconocida)

Cambiar a un runlevel mas alto es razonable pero a uno menor que el actual es peligroso, recuerde que existen aplicaciones, utilidades y servicios ejecutándose que probablemente terminaran sin avisar. Por ejemplo el nivel 0 es una manera imprudente de apagar ya que no avisa a los usuarios en sesión en el sistema que se esta a punto de apagar todo.

Archivo de Control del Inicio: /etc/inittab

El archivo /etc/inittab contiene la descripción general del proceso de arranque del sistema. Al terminar de cargarse el kernel, se llama al programa /sbin/init, que es el proceso padre de todos los demás procesos existentes en el sistema. init sigue las instrucciones incluidas en inittab para llevar al sistema a un estado usable, creando los procesos descritos en este archivo.

Un runlevel (o nivel de ejecución) es una configuración por software del sistema que permite existir sólo a un grupo seleccionado de procesos. El sistema (por medio de init) está en cada momento en un runlevel concreto. El superusuario puede cambiar el runlevel en cualquier momento ejecutando telinit. Si efectua cambios al archivo inittab no es necesario para que los cambios surjan efectos, sólo deberá ejecutar el siguinete comando:

 # init q o # telinit q

Estos comandos le ordenan a a init que lea su archivo de configuración sin cambiar de runlevel. El comando telinit es un vínculo a init en las mayoría de distros de GNU/Linux de hoy día. Hay ocho runlevels, llamados 0, 1, 2, 3, 4, 5, 6 y S. El runlevel 0 se usa para parar el sistema, el 6 para reiniciarlo y el S o el 1 para ponerlo en modo monousuario. Los demás niveles se utilizan para proporcionar determinado grado de servicios. Por ejemplo, es normal usar un nivel para el uso normal, otro para el arranque automático de XWindow, otro para uso sin red, etc.

Discutimos la estructura del archivo en la siguiente seccion.

Estructura del Archivo /etc/inittab

Cada línea del archivo esta dividida en cuatros campos separados por dos puntos:
Id: nivel: acción: proceso

Campos	**Descripción**
ID	Identificador unico de la línea hasta 4 caracteres alfanuméricos.
Nivel	Runlevel que activa este proceso indicado mas adelante
Acción	Palabra clave de cómo correr el proceso
Proceso	Nombre completo y parámetros del comando para ejecutar

En la siguiente tabla se incluyen las palabras claves principales de acción del archivo inittab/

Campos	**Descripción**
Off	No ejecute este comando
Wait	Ejecute este comando y espere que complete
Once	Ejecute este comando y no espere
Respawn	Ejecute este comando; y si falla, ejecutelo de nuevo
Sysinit	Ejecute el comando durante el primer init
Boot	Ejecutar el comando al tiempo de arranque y no espere
Bootwait	Como el Boot pero espere que complete
Ctrlaltdel	Ejecute el comando especificado al presionar estas teclas
initdefault	Define nivel de arranque por defecto

Al iniciarse, init busca la línea initdefault en /etc/inittab para pasar al nivel por defecto.

A continuación lee las demás líneas del archivo. Cada línea tiene la forma:
Id: niveles: accion: procesos

Donde id es una identificación para la línea, runlevels especifica los niveles en los que esta línea debe aplicarse, action es la acción que se va a realizar y Process es el proceso a ejecutar.

El init va ejecutando los procesos especificados en las líneas que incluyan el nivel de ejecución actual. Cuando se cambia el nivel, los procesos del nivel antiguo son eliminados.

Los procesos que se ejecutan en un estado normal son los scripts de arranque primero y las invocaciones a los programas que permiten a los usuarios usar el sistema después. Una de estas líneas podría ser como la siguiente:

1:2345:respawn:/sbin/getty 38400 tty1

Esta línea indica que en los niveles de ejecución 2,3,4 ó 5 debe ejecutarse el programa /sbin/getty (que es el programa que se encarga de pedir el login y el password a los usuarios) con los parámetros 38400 (que

indica que espere comunicaciones a 38400 baudios), y tty1 (que indica que escuche en la primera terminal virtual). La palabra clave respawn indica que este proceso debe reiniciarse cuando termine, de forma que cuando un usuario salga del sistema, se vuelva a mostrar el prompt de login para aceptar otro usuario.

Hay que manejar con cuidado el fichero inittab, ya que una modificación indebida del mismo puede dejar al sistema en un estado incapaz de arrancar.

LOS SCRIPTS DE LOS NIVELES DE EJECUCION

Los sccripts de ejecución se encuentran en los directorios /etc/rc.d/rc*.d y son iniciados bajo la dirección del archivo /etc/inittab. Cada runlevel diferente contiene sus propios archivos scripts que se ejecutan al inicio de la sesión del runlevel.

rc0	Busca en	/etc/rc.d/rc0.d	Para el nivel 0	Halt/apagado
Rc1	Busca en	/etc/rc.d/rc1.d	Para el nivel 1	Single user/unico usuario
Rc2	Busca en	/etc/rc.d/rc2.d	Para el nivel 2	Multiusuario, sin network
Rc3	Busca en	/etc/rc.d/rc3.d	Para el nivel 3	Total multiusuario con soporte de redes
Rc4	Busca en	/etc/rc.d/rc4.d	Para el nivel 4	Reservado, no se usa
Rc5	Busca en	/etc/rc.d/rc5.d	Para el nivel 5	X11 (inicio gráfico, multiusuarios)
Rc6	Busca en	/etc/rc.d/rc6.d	Para el nivel 6	Reboot/reinicio

La mayoría de distribuciones dejan el nivel dos y el nivel cuatro para que el administrador introduzca sus propios scripts/guiones de inicio en este nivel /etc/rc.d/rc4.d / o el /etc/rc.d.

El scripts busca en el directorio debajo del /etc/rc.d/init.d para scripts a ejecutar para los niveles de ejecución requeridos.

Cada uno de estos directorios contiene vínculos a archivos almacenados en el /etc/init.d. Los nombres de los vínculos son referentes al orden predeterminado a ejecución de dos digitos. Nombres de scripts que empiezan con “S” son iniciado en su apropiado runlevel y los scripts que empiezan con “K” los detiene (killed). Cuando el scripts se invoca desde el programa rc, el primer parámetro esta colocado en la palabra “start” o “stop”. El script en /etc/rc.d/init.d debe buscar el valor de este parámetro para determinar que acción tomar (Start o el kill).

Detalles del rc Script

El siguiente código ilustra un script rc:

```
# less /etc/init.d/inet
case “$1” in
   start) …
          …;;
   stop) …
          …;;esac
# ln init.d/inet rc0.d/K69inet
# ln init.d/inet rc2.d/S69inet
# init 2
# init 0
```

En este ejemplo, el rc script es mantenido en /etc/rc.d/init.d/inet. Este script es vinculado a /etc/rc.d/rc2.d/S69inet con esto el servicio de Internet es iniciado al cambiar al modo multiusuario. El script ejemplo es vinculado a /etc/rc.d/rc0.d/K69inet con esto el servicio de Internet es detenido al entrar al modo usuario único.

El script contiene el código necesario para iniciar o detener el servicio de Internet. La función iniciada o detenida es pasada en una palabra clave al script llamado, bueno una forma típica para todo rc script es el usar un shell… entre la funcionalidad del inicio y detenido.

El comando:	**El comando:**
# init 2	# init 0
ejecuta el comando:	ejecuta el comando:
sh S69inet start	sh K69inet stop

**** Nota esto no es necesario para la secuencia de número de los nombres de archivos que están vinculado al ejemplo en diferentes rc directorios.*

Ejercicio 4-3: Encendido y Apagado del Sistema

En este ejercicio nosotros podríamos reconfigurar el mecanismo de encendido y apagado de linux. Soluciones se proveen en el Apéndice A.

1.- ¿Cuál es el runlevel default de sistemas y como lo determinas?

2.- Cree un simple script de inicio entrando los siguientes comandos:

```
# cat >/etc/rc.d/init.d/tele
case "$1" in
    start) echo Empezando Servicios Telepaticos;;
    stop) echo Deteniendo Servicios telepaticos;;
esac
^D
# ln –s /etc/rc.d/init.d/tele /et/rc.d/rc2.d/S01tele
# ln –s /etc/rc.d/init.d/tele /et/rc.d/rc2.d/K01tele
# ln –s /etc/rc.d/init.d/tele /et/rc.d/rc0.d/K01tele
```

Ahora pruebe sus cambios reiniciando su computador y observando los mensajes en tu pantalla.

Ejercicio 4-4: Cambiando de runlevel (nivel de ejecución)

La solución esta provista en el Apéndice A.

1.- Cambie su default runlevel al nivel 4. Habilita los niveles 2 y 3 dentro del 4 y agrega los requerimientos para configurarla /etc/rc.d/rc4.d directorio.

RESUMEN

En este capítulo usted fué introducido los siguientes conceptos:

- Procesos y administrador de procesos
- Señales
- Daemons
- Administración de memorias
- Registros de procesos
- GNU/Linux runlevels
- El archivo /etc/inittab
- El script de la función rc en proceso de inicio.

Preguntas Post-Exámen

Las respuestas de estas preguntas se encuentran en el Apéndice A.

Si usted tiene tres programas debe cerrar. Al usar el comando ps ux muestra lo siguiente:

```
[root@www /root]# ps ux
USER       PID  %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND
root         1  0.0  0.2  1424 548 ?   S    09:21 0:04  init [7]
root         2  0.0  0.0     0   0 ?   SW   09:21 0:00  [kflushd]
root         3  0.0  0.0     0   0 ?   SW   09:21 0:00  [kupdate]
root         4  0.0  0.0     0   0 ?   SW   09:21 0:00  [kswapd]
root         5  0.0  0.0     0   0 ?   SW   09:21 0:00  [keventd]
root         6  0.0  0.0     0   0 ?   SW   09:22 0:00  [mdrecoveryd]
```

¿Cuál es el PID de cada uno de estos procesos? ¿Cuál es el estado de cada uno de estos procesos?

2.- Usted ha hecho un cambio en el archivo de configuración de un daemon llamado aaad. Sin embargo, el daemon parece no reconocer los cambios. ¿Cómo podemos hacer que el daemon relea el archivo config?

3.- Usted esta corriendo un comando make que da inicio y ejecuta un comando hijo make y este desestabilizando el CPU (debido a que la compilación tiende hacer uso intensivo del CPU). ¿Cómo puede detenerse este proceso?

4.- ¿Cómo puede usted mostrar cuales programas han sido ejecutados más veces desde la última vez que se reinicio el sistema?

5.- Ivelis empezó un proceso con el comando nice y le dió una prioridad inicial menor que la del sistema. Después ella se da cuenta que el proceso tiene más urgencia que lo que pensó. ¿Cómo puede Ivelis alterar la prioridad del proceso?

CAPITULO 5

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

ADMINISTRACIÓN DE DISCO Y CUOTAS

OBJETIVOS

Al completar este capítulo, usted podrá:

- Administrar los permisos de archivos.
- Establecer y visualizar Cuotas en Disco.
- Identificar estándares de sistemas de archivos jerárquicos.
- Destacar la estructura del sistema de archivos.
- Identificar herramientas DOS y su uso en Linux.
- Distribución del espacio en Disco Duro.
- Creación de particiones y sistemas de archivos.
- Mantenimiento de integridad de sistemas de archivos.
- Controlar el montaje y desmontaje de archivos.
- Describa Cuotas, sus conceptos y su implementación en usuarios y grupos.
- Familiarizarse con el sistema de archivos de Linux.
- Listar los diferentes niveles de almacenaje de RAID.
- Defina el rol del archivo fstab.
- Describa la función del fsck.

PREGUNTAS PRE-EXAMEN

Respuestas a estas preguntas se encuentran en el Apéndice A.

1.- ¿Liste algunos de los sistemas de archivos disponibles en GNU/Linux?

2.- ¿Qué comando se usa para particionar un disco duro?

3.- ¿Cuál es los propósitos de la Cuotas?

4.- ¿Qué archivo se puede utilizar para especificar montar volúmenes automáticamente?

INTRODUCCION

Este capítulo examina el sistema de archivos Linux. Este sistema de archivos posee una estructura de almacenamiento jerárquico. El sistema de archivos esta compuesto de archivos y directorios con un unico directorio root (/) proveyendo el acceso a todo el disco, archivos y directorios en el sistema. La jerarquía de estructura de árbol es ilimitada en profundidad y ancho y puede ser extendida utilizando facilidades de redes incluyendo disco y otros sistemas de archivos.

El sistema GNU/Linux ha llevado el concepto de archivo un poco más allá que cualquier otro sistema operativo; en linux, todo es un archivo. Los dispositivos físicos en el sistema son accesados utilizando un nombre de archivo especial en el sistema de archivos. Hasta la memoria y los procesos (programas) ejecutándose son accesibles a través de un archivo en el sistema.

EL SISTEMA DE ARCHIVOS JERARQUICO

El sistema de archivos linux tiene una estructura de almacenaje jerárquica. Este sistema de archivos es como un árbol cubriendo toda la data el sistema. Los archivos son organizados, utilizando directorios, en orden jerárquico o estructura de árbol.

Los siguientes tópicos se cubren en esta seccion:

- Visualización del sistema de archivos.
- Las utilidad MTOOLS
- Control de acceso
- Vínculos

Visualizar un Sistema de Archivos

Aunque en manera diferente en sistemas operativos almacenan su data, totas ellas son gobernadas por principios y comparten un propósito básico:

- Data es almacena en dispositivos físicos, como son disquetes o discos duro.
- Dispositivos físicos se dividen en segmentos virtuales llamados particiones.
- Sistemas de archivos son colocados en particiones en orden para estructurar como el sistema operativo maneja la data.
- No existen letras en el sistema de archivos.
- Dispositivos como disco duro o cdrom se pueden colocar en puntos de montaje, cuales son directorios normales y donde el contenido puede ser accesado.
- Archivos técnicamente no almacenan data, son utilizados solamente como nombre de referencia que corresponde a un inode, el cual contiene toda la información necesaria del archivo, como sus permisos y fechas de modificación y apunta al bloque físico de almacenaje en el dispositivo que contiene la data.
- El término archivo es utilizado para especificar el nombre, inode y el bloque de data físico como un todo.
- Los vínculos son un archivos virtual que apunta a otro archivo, esto es inherente al sistema de archivos, usted puede, colocar un vinculo a un directorio ubicado en otro sitio del sistema de archivos. Acceder un vinculo es idéntico es idéntico a acceder al archivo al que el apunta.
- Información de díspositivos y el KERNEL son virtualmente traducidas a archivos y colocadas en el sistema debajo de directorios.

Raíz Central del Sistema de Directorio

Los usuarios como una estructura grande, parecida a un árbol, la raíz del árbol es un directorio especial llamado "/"o "root". Detalles de cual disco contiene los archivos o directorios es oculto del usuario; solamente existe una jerarquía a respetar. Esto es un contraste al DOS y al OS/DOS donde cada disco tiene su propia jerarquía con su propia raíz. Este concepto es expandido a incorporar disquete, disco duros externos,

CDROM y hasta discos en otras computadoras.

/	Raíz del sistema de archivos.
/dev	Contiene archivos del sistema representando los dispositivos que estén físicamente instalados en el ordenador.
/etc	Este directorio esta reservado para los archivos de configuración del sistema. En este directorio no debe aparecer ningún archivo binario (programas). Bajo este deben aparecer otros dos subdirectorios:
/etc/X11	Archivos de configuración de X Window
/etc/skel	Archivos de configuración básica que son copiados al directorio del usuario cuando se crea uno nuevo.
/lib	Contiene las librerías necesarias para que se ejecuten los programas que residen en /bin (no las librerías de los programas de los usuarios)
/proc	Contiene archivos especiales que o bien reciben o envían información al kernel del sistema (Se recomienda no modificar el contenido de este directorio y sus archivos).
/sbin	Contiene programas que son únicamente accesibles al superusuario o root.
/usr	Este es uno de los directorios más importantes del sistema puesto que contiene los programas de uso común para todos los usuarios. Su estructura suele ser similar a la siguiente:
/usr/X11R6	Contiene los programas para ejecutar X Window.
/usr/bin	Programas de uso general, lo que incluye el compilador de C/C++
/usr/doc	Documentación general del sistema.
/usr/etc	Archivos de configuración generales
/usr/include	Archivos de cabecera de C/C++ (.h).
/usr/info	Archivos de información de GNU
/usr/lib	Librerías generales de los programas.
/usr/man	Manuales accesibles con el comando man (ver más adelante).
/usr/sbin	Programas de administración del sistema
/usr/src	Código fuente de programas.

******** Existen además de los anteriores directorios que se suelen estar en los sistemas de archivos

/usr	Como por ejemplo las carpetas de los programas que se instalen en el sistema.
/var	Este directorio contiene información temporal de los programas (lo cual no implica que se pueda borrar su contenido, de hecho, ¡no se debe hacer!)

Uso del Suite de Herramientas mtools

El hecho de tener que "montar" y "desmontar" puede ser un poco engorroso a la hora de utilizar determinados dispositivos (comúnmente, la disquetera). Por ello, se dispone de las herramientas mtools. Dichas herramientas, utilizan los dispositivos sin tener que "montar" y "desmontar"; y su sintaxis es parecida a la de los programas de DOS. Estas herramientas sólo trabajan sobre dispositivos con sistemas de archivos DOS. Aquí también le mostramos algunas características adicionales.

Nombre de comandos:
mzip, mtype, mtooltest, mshowfat, mattrib, mbadblocks, mcd, mcopy, mdir, mdu, minfo, mmd, mmount, mmove, moartition, mrd, mread y mren.

Comando	**Descripción**
mdir	Muestra el contenido del dispositivo dir.
mcopy	Copia archivos copy.
mdel	Borra archivos del.
mformat	Formatea la unidad format.
mcd	Cambia de directorio cd.
mmd	Crea un directorio md.
mrd	Borra un directorio rd.

Control de Acceso

Cada archivo tiene tres tipos de permisos para los tres tipos de usuarios. Para accesar un archivo los permisos de accesos del usuarios deben ser permitidos. Utilizando los atributos de propiedad y de grupo, en combinación con los UID de los usuarios y los grupos, tres tipos de accesos pueden ser determinados. El superusuario no tiene restricción. Si el usuario es el dueño de los archivos se usa bandera de dueño. Si el usuario no es el dueño del archivos pero esta en el grupo del dueño, se levanta la bandera de grupo, si el usuarios no esta grupo y no es el propietario se levanta la bandera de otros.

Permisos de Archivos y Directorios

Para entender mejor el concepto de permisos se tendrá que tener en cuenta que cada usuario puede pertenecer a uno o más grupos. Cada usuario pertenece por lo menos a un grupo, que es establecido en el momento en que el usuario se crea.

El administrador del sistema puede agregar al usuario a otros grupos. Estos grupos son necesarios para poder establecer una política de acceso mas organizada dado que en cualquier momento se podría dar a un archivo determinado el acceso a personas de un grupo determinado lo único que se tendría que hacer es agregar a los usuarios que se quieran dar permisos a ese grupo.

Para cada objeto (archivo) que se encuentre en el sistema, GNU/Linux guarda información administrativa en la tabla de inodos, tema que abarcaremos en mayor detalle mas adelante. Entre los datos que se guardan en esta tabla se encuentran la fecha de creación del archivo, modificación del archivo y la fecha en que se cambio el inodo.

Pero además contiene los datos en los que se centra toda la seguridad en UNiX. Estos son:

- El dueño de archivo
- El grupo del archivo
- Los bits de modo o también llamados permisos de archivo

En este tramo nos centraremos en primera medida en entender los permisos y en establecer la forma en que pueden trabajar con ellos.

Conceptos

Al ser GNU/Linux un clon de UNiX y un sistemas operativos multiusuario, para que se puedan proteger los archivos se estableció un mecanismo por el cual se le pueden otorgar permisos a un determinado usuario o grupo. Esto permite, por ejemplo, que si existe un archivo creado por un usuario en particular, este será propiedad del usuario y también tendrá el grupo del usuario. Se permite que los archivos sean compartidos entre usuarios y grupos de usuarios. Por ejemplo si Ivelis quisiera podría prohibir los accesos a un archivo determinado que le pertenezca a todos los usuarios que no pertenezcan a su grupo de usuarios.

Los permisos están divididos en tres tipos: Lectura, escritura y ejecución (rwx). Estos permisos pueden estar fijados para tres clases de usuario: el propietario del archivo, el grupo al que pertenece el archivo y para todo el resto de los usuarios.

El permiso de lectura permite a un usuario leer el contenido del archivo o en el caso de que el archivo sea un directorio, la posibilidad de ver el contenido del mismo. El permiso (w) escritura permite al usuario modificar y escribir el archivo y en el caso de un directorio permite crear nuevos archivos en él o borrar archivos existentes. El permiso de ejecución (x) permite al usuario ejecutar el archivo, si tiene algo por ejecutarse. Para los directorios permite al usuario cambiarse a él con el comando “cd”.

Permiso	Aplica a Archivo
R	Lectura de archivo
W	Escritura de archivo (no implica lectura)
X	Ejecución de un archivo (programa o script de shell); para el programa lectura no es requerida; para shell script si tiene que tener lectura

Permiso	Aplica a Directorio
r	Puede listar directorios (no implica accesos a su archivos)
w	Puede escribir a un directorio (puedes crear, renombrar y borrar directorio)
x	Puede buscar en el directorio (pasar y acceder archivos)

Para crear un archivo necesitas lo siguiente

--x	Permisos en todo los directorios en la ruta
-wx	Permiso en el directorio donde vas a crear el archivo

Para leer un archivo necesitas lo siguiente

--x	Permiso en todo los directorio en la ruta
r--	Permiso en el archivo

Para escribir en un archivo necesitas lo siguiente

--x	Permiso en todo los directorio en la ruta
-w-	Permiso en el archivo

Para ejecutar un comando el bit de ejecución (x) debe estar encendido. Nótese que para el script shell requiere permiso de lectura (r) así también el de ejecución (x) antes del script ser invocado.

El nombre del archivo no es parte del archivo, es una función del directorio que lo contiene. Y por esto para borrar o renombrar el archivo necesita permiso de escritura. Mover un archivo de un directorio a otro requiere permisos de escritura en ambos directorio.

El bit de ejecución de un directorio no significa ejecutar, significa buscar. Si este bit no esta encendido significa que el usuario no puede buscar en el directorio aunque el sea el dueño del directorio.

Como se Interpretan los Permisos

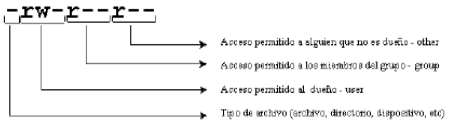
Para poder interpretar los permisos de archivos nada mejor que utilizar el comando “ls -la” Para ver un listado largo de un directorio e ir viendo cada uno de ellos.

```
[desiree@abiertos]$ ls -la
total 13
drwxr-sr-x  2 mike  user   1024  May  2   09:04 .
drwxrwsr-x  4 root  staff  1024  Apr  17  21:08 . . .
-rw- --- ---  1 mike  user   2541  May  2   22:04 .bash_history
-rw-r--r--   1 mike  user   164   Apr  23  14:57 .bash_profile
-rw-r--r--   1 jazzy user   55    Apr  23  14:44 .bashrc
-rwxrwxr-x  1 ivelis user   0     Apr  14  19:29 a.out
-rwxrwxr-x  1 ivelis user   40    Apr  30  12:14 hello.pl
-r-- --- ---  1 ivelis user   64    Apr  29  14:04 hola
-rwxrw-r--   1 ivelis user   337   Apr  29  13:57 lista
-rw-rw-r--   1 desi   user   40    Apr  30  12:31 listador
-rw-rw-r--   1 des    user   0     May  2   09:04 null
-rwxrwxr-x  1 silvia user   175   Apr  30  12:30 prue.pl
```

Como se puede apreciar en este listado, también están el directorio actual, representado por un punto “.” y el directorio padre representado por dos puntos “..”. Ellos también poseen permisos y atributos que son

mostrados.

Para ir entendiendo un poco más vamos a explicar que significan los primeros 10 dígitos. Veamos unas tablas que nos esclarecerán un poco mas que significa cada uno de estos caracteres. Primero que todo veremos aquellos caracteres que podrían aparece en el primer bit que en el ejemplo anterior podemos ver que es un solo guión, esto nos indica que es un archivo común. La tabla siguiente explica el significado del primer símbolo de acuerdo al tipo de archivo:



Contenido	Significado
-	Archivo común
d	Directorio
c	Dispositivo de caracteres (tty o impresora)
b	Dispositivo de Bloque (usualmente disco rígido o CD-ROM)
l	Enlace simbólico
s	Socket
p	Pipe

Los siguientes 9 símbolos se toman en grupos de tres y cada grupo pertenece a una clase de permisos y se muestran a continuación:

-rwxrw-r-- 1 ivelis user 337 Apr 29 13:57 lista

Columnas	Se aplica a	Significado
2,3,4	owner	Establece permisos para el dueño del archivo
5,6,7	group	Establece permisos para el grupo del archivo
8,9,10	other	Establece permisos de los usuarios que no estan en las categorías anteriores

De esta forma podremos interpretar el listado generado a partir de “ls -l” de mejor manera. El primer símbolo nos esta indicando que el archivo es un archivo común. El primer grupo de tres símbolos representa los permisos para el dueño del archivo (owner) que en este caso posee permisos de lectura, escritura y ejecución.

El segundo grupo de tres símbolos representa los permisos para el grupo al cual pertenece el archivo (group), que en este caso tienen permisos de lectura y escritura. El tercer grupo de tres símbolos representa los permisos para todo el resto de los usuarios (other) en este caso es solo de lectura. El numero que sigue (1) representa el numero de nombres que el archivo posee. Esto representa la cantidad de enlaces que existen a este archivo y lo veremos mas adelante cuando tratemos el tema de enlaces simbólicos y duros.

A continuación esta el nombre del dueño del archivo y del grupo al cual pertenece el archivo. El “337” representa el tamaño del archivo expresado en bytes. Lo siguiente es la fecha y hora de modificación del archivo e inmediatamente después esta el nombre del mismo.

Dependencias

Los permisos de los archivos también dependen del directorio donde estén guardados. En un ejemplo común podríamos dar el caso de un archivo que posea todos los permisos, tanto para el usuario, grupo y otros pero no se podrá acceder a el si no se cuenta con permisos de lectura y ejecución en el directorio que los contiene.

Esto funciona en el caso que se quiera restringir el acceso a un directorio determinado y a todos los archivos que este contiene. En lugar de cambiar los permisos uno por uno solo tenemos que sacarle los permisos necesarios para que se prohíba el acceso mismo al directorio y con esto y aunque se disponga de todos los

permisos en los archivos contenidos en este directorio, no podrán ingresar para usarlos.

Esto también esta dado para toda la ruta del archivo. Es decir que no-solo el último directorio, el cual lo contiene, tiene que tener los permisos necesarios, sino que todos los directorios que lo preceden también.

Cambiar Permisos

El comando chmod se emplea utilizando símbolos como "a, u, g, o" que representan a todos (a "all") al usuario (u) al grupo (g) y a todos los demás (o). Existen símbolos para agregar (+) quitar (-) o dejar invariantes loa permisos (=). Además se tendrán que usar los símbolos característicos para cada tipo de permiso. Para el permiso de lectura (r), para el permiso de escritura (w) y para el permiso de ejecución (x). Solo el dueño del archivo puede cambiarlo con él; excepción del root que también lo puede hacer. Para ejemplificar un cambio de permisos usaremos el archivo "lista".

```
[ivelis@ abiertos]$ ls -l lista
total 1
-rwxrw-r-- 1 ivelis user 337 Apr 29 13:57 lista

[ivelis@abiertos]$ chmod a-r lista

[ivelis@abiertos]$ ls -l lista
total 1
--wx-w---- 1 ivelis user 337 Apr 29 13:57 lista
```

De esta forma se le han eliminado a todos los grupos los permisos de lectura. Aquí algunos ejemplos más:

```
[ivelis@abiertos]$ chmod u+r lista
[ivelis@abiertos]$ ls -l lista
total 1
-rwx-w---- 1 ivelis user 337 Apr 29 13:57 lista

[ivelis@abiertos]$ chmod o+w lista

[ivelis@abiertos]$ ls -l lista
total 1
-rwx-w--w- 1 ivelis user 337 Apr 29 13:57 lista

[ivelis@abiertos]$ chmod og-w lista

[ivelis@abiertos]$ ls -l lista
total 1
-rwx------ 1 ivelis user 337 Apr 29 13:57 lista
```

# chown root /usr/bin/passwd	Cambia a root la propiedad del comando /usr/bin/passwd
# chgrp sys /usr/bin/passwd	Cambia al grupo sys la propiedad del comando /usr/bin/passwd
# chown -R ivelis /home/ivelis	Cambia recursivamente la propiedad del directorio /home/ivelise a ivelis
# chmod ugo=rwx /tmp	Establece los permisos del directorio /tmp para todos a rwxrwxrwx 0 777
# chmod +t /tmp	Agrega el stiky bit al directorio /tmp
# chmod ug+s /usr/bin/passwd	Agrega el stiky bit al archivo /usr/bin/passwd para el dueño y el grupo
# chmod 1777 /tmp	Establece permisos de 1777 al directorio /tmp
# chmod -R 664 /projecto/bin	Establece permisos recursivamente en el directorio /projecto/bin a 664

Ahora bien esta es la forma simbólica, pero existe una forma un poco más sistemática que es la forma de representación octal. El comando chmod permite establecer los permisos de un archivo por medio de un numero octal de dígitos. Comúnmente nosotros usamos para contar una representación decimal (0,1,2,3,4,5,6,7,8,9) pero en una representación octal solo se usan 8 números (0,1,2,3,4,5,6,7). Para estable-

cer el permiso habrá que sumar los dígitos octales de acuerdo a una tabla que se dará a continuación. Dado que no se realiza acarreo, la suma será común.

Para dar un ejemplo de la suma que se tendrá que realizar, tomemos un archivo con los permisos expresados en forma simbólica y realicemos la conversión. Ej. Para representar "-rwxr-w—-":

De esta forma si lo que quisiéramos es cambiar los permisos de un archivo, solo se tendría que efectuar la suma necesaria y establecerlo con el comando chmod. Como ejemplo si se quisiera cambiar los permisos para que el dueño tenga permisos de lectura y escritura y que el grupo y otro solo tengan permisos de lectura, la sintaxis seria:

```
[ivelis@abiertos]$ chmod 0644 lista

[ivelis@abiertos]$ ls -l lista
total 1
-rw- r-- r-- 1 ivelis user 337 Apr 29 13:57 lista
```

Con la práctica se sabrán cuales son las sumas mas utilizadas y podrán ver que es mucho más sencillo el establecer de esta forma los permisos de archivos. Aquí le entregamos una tabla con los valores numéricos equivalente a los permisos:

Número octal	Permiso
4000	Establece el número de identificación de usuario al ejecutarse SUID [a]
2000	Establece el número de identificación de grupo al ejecutarse SGID[a]
1000	Establece el bit adhesivo[a]
0400	Lectura por parte del dueño
0200	Escritura por parte del dueño
0100	Ejecución por parte del dueño
0040	Lectura por parte del grupo
0020	Escritura por parte del grupo
0010	Ejecución por parte del grupo
0004	Lectura por parte de los otros
0002	Escritura por parte de los otros
0001	Ejecución por parte de los otros

Lo que nos quedaría ver es el caso en que se quisiera cambiar el usuario o el grupo del archivo. Para esto se usa el comando "chown" y su sintaxis es similar a la de chmod pero con la variante que se dan los nombres de los usuarios y del grupo. Por ejemplo, supongamos que quisiéramos cambiar el nombre de usuario del archivo lista. Al igual que con los cambios de permisos tendremos que ser el dueño del archivo a cambiar, o el root.

```
[ivelis@abiertos]$ ls -l lista
total 1
-rw-r--r-- 1 ivelis user 337 Apr 29 13:57 lista

[ivelis@abiertos]$ chown miguel lista

[ivelis@abiertos]$ ls -l lista
total 1
-rw-r--r-- 1 miguel user 337 Apr 29 13:57 lista
```

Si se quisiera cambiar también el nombre del grupo, se tendría que poner un punto entre el nombre de usuario y el grupo:

```
[ivelis@abiertos]$ ls -l lista
```

```
total 1
-rw-r--r-- 1 ivelis user 337 Apr 29 13:57 lista

[ivelis@abiertos]$ chown miguel.ventas lista

[ivelis@abiertos]$ ls -l lista
total 1
-rw-r--r- - 1 miguel ventas 337 Apr 29 13:57 lista
```

Por supuesto que tanto el usuario como el grupo al que se hacen referencia tendrán que existir en el sistema, sino saldrá un error.

En el caso que solo se quiera cambiar el grupo y no el usuario, se tendrá que poner un punto delante del nombre del grupo, omitiendo poner el nombre del algún usuario. O si se quiere, se podrá poner el nombre de usuario que estaba anteriormente.

```
[ivelis@abiertos]$ ls -l lista
total 1
-rw-r--r-- 1 ivelis user 337 Apr 29 13:57 lista

[ivelis@abiertos]$ chown .ventas lista

[ivelis@abiertos]$ ls -l lista
total 1
-rw-r--r-- 1 ivelis ventas 337 Apr 29 13:57 lista
```

Puntos adicionales

Explicaremos algunos puntos acerca de los permisos que son de gran utilidad para la seguridad de nuestro sistema:

umask.- Esta es la abreviatura de "user file-creation mode mask" o "mascara del modo de creación de archivos de usuario" y es un número octal de cuatro dígitos que se utilizan para fijar los permisos de los archivos recién creado. Esto puede ocasionar confusión pero en realidad es una utilidad que permite el uso del sistema por múltiples usuarios sin que peligre la privacidad. En la mayoría de los UNIX los archivos que son creados por el usuario poseen permisos 0666 que dan permiso de lectura y escritura a cualquier usuario. En relación con los programas, estos se levantan con 0777 donde cualquier usuario puede leer, escribir y ejecutar el programa. Normalmente el administrador del sistema aplica una "mascara" al usuario en el archivo .bash_profile y esta es usada para la creación de archivos haciendo una operación simple "AND" bit por bit con el complemento del valor umask bit por bit. La función umask esta integrada al intérprete de comandos.

Para ejemplificar el proceso tomemos un archivo creado por el usuario ivelis con una mascara de 0022 que produce archivos con permisos de 0644.

```
[ivelis@abiertos]$ ls -l archivo
total 1
-rw-r--r- - 1 ivelis user 337 Apr 29 13:57 archivo
```

El modo resultante es que el dueño tiene permisos de lectura y escritura y los demás y el grupo sólo de lectura. Una forma de darse cuenta de la forma en que funciona el umask es tener en cuenta que el valor (2) inhabilita el permiso de escritura mientras que el valor (7) inhabilita los permisos de lectura escritura y ejecución.

A continuación daremos una tabla con los valore comúnmente usados para el umask.

Umask	Accesos del usuario	Accesos del grupo	Accesos de los otros
0000	Todos	Todos	Todos

0002	Todos	Todos	Lectura y ejecución
0007	Todos	Todos	Ninguno
0022	Todos	Lectura y ejecución	Lectura y ejecución
0027	Todos	Lectura y ejecución	Ninguno
0077	Todos	Ninguno	Ninguno

Establecer ID de Usuario y de Grupo (SUID Y SGID)

Existen ocasiones que los usuarios necesitan ejecutar algún programa que requiere de privilegios. Un ejemplo de esto es el uso del programa passwd para cambiar la contraseña. Seria un error darle a los usuarios los privilegios necesarios para que puedan ejecutar esta clase de programas ya que el usuario podría cambiarse de grupo o crear una cuenta con privilegios de root. Para que esto no suceda se implemento en UNIX un sistema por el cual un programa que cuente con SUID o SGID puede ser ejecutado con los privilegios del dueño y/o grupo del programa.

Para que quede mas claro se tiene que saber que cada usuario esta identificado por el sistema con un numero de identificación tanto para el, como para el grupo. Este numero se denomina UID (user ID) para el caso de los usuarios y GID para el caso de los grupos. Por ejemplo, un usuario podría tener un UID 100 y un GID 500. EN el caso del root, este tiene UID 0 y GID 0. Mas adelante se verá esto en mayor detalle. Lo que se efectúa con el sistema SUID es una adquisición temporal de un UID o GID distinto al propio cuando se esta ejecutando el programa. Cuando un programa cambia de UID se denomina SUID (set-UID: se establece UID) y cuando cambia de GID se denomina SGID (set-GID: se establece GID) Un programa puede ser SUID y SGUID al mismo tiempo.

Para darse cuenta si un programa es SUID o SUI basta con hacer un listado largo con el comando “ls -l” y se verá que donde tendría que estar una “x”, que asigna permisos de ejecución, va a estar una letra “s”. Aquí un ejemplo:

```
[ivelis@abiertos]$ ls -l /usr/bin/passwd
-rwsr-xr-x 1 root root 28896 Jul 17 1998 /usr/bin/passwd
```

El BIT Sticky/Adhesivo

En los antiguos sistemas UNIX la memoria era algo esencial y escasa dado su costo. Para poder aprovechar más esta se empleo una tecnología que mantenía parte de programas esenciales en el área swap de memoria para que pudieran ser usados más rápidamente dado que si se tendrían que ir a buscar al disco se tardaría más. Estos programas se los marcaba con un BIT especial, un BIT adhesivo o “sticky BIT” y estos archivos así marcados eran los que valía la pena mantener ya que esas partes del programa que se guardaban en memoria también podían ser usadas por otros.

En los sistemas operativos modernos de hoy el sticky-bit encendido en un directorio significa que solo el dueño del archivo tiene derecho a borrarlo.

Directorios públicos con permisos de escritura (w) deben tener el sticky-bit encendido ya que de otra manera todos pueden borrar archivos aunque no sean de ellos. Al encender el bit solamente los dueños podrán borrar sus archivos.

```
root@gnome2:/# ls -la /tmp
total 80
drwxrwxrwt        6       root    root    20480   jul 20 22:29    .
drwxr-xr-x        6       root    root    4096    jul 20 22:28    ..
-rw-------        1       gnome2 ged2     264     jul 20 22:20    .gdmfujrEI
```

```
srw-rw-rw-      1       root    root    0       jul 20 22:20    .gdm_socket
drwxrwxrwt      2       root    root    4096    jul 20 22:21    .ICE-unix

$ rm -rf /tmp/.ICE-unix/
rm: no se puede borrar el directorio "/tmp/.ICE-unix": Operación no permitida
```

Aunque el directorio/tmp esta "drwxrwxrwt" con todos los permisos el sticky-bit "t" no solo permite que el dueño "root" borre los archivos.

Ejercicio 5-1: Permisos de Archivos

Solución se proveen en el Apéndice A. Lo que sigue es una sesión:

```
[alex@www ABIERTOS]$ id -a
uid=5027(alex) gid=5027(alex) groups=5027(alex)

[alex@abiertos]$ ls -ld . arch* /etc/passwd /etc
drwxrwxrwx      2       root    root    4096    Jul 19 19:29    .
-rw-rw-r--      1       alex    alex    0       Jul 19 19:29    arch1
-rw-r-----      1       alex    alex    0       Jul 19 19:29    arch2
-rw-rw-rw-      1       alex    alex    0       Jul 19 19:29    arch3
drwxr-xr-x      37      root    root    4096    Jul 19 09:27    /etc
-rw-r--r--      1       root    root    4435    Apr 4 20:24     /etc/passwd
```

1.- Basado en la información arriba presentada, ¿cuáles de las siguientes operaciones son permitidas?

```
$ vi arch1
$ more arch2
$ ls -l > arch1
$ less /etc/passwd
$ rm arch2
$ rm arch3
$ cp arch1 arch4
$ rm /etc/passwd
```

Vínculos/Links

Un archivo puede tener muchos nombres asociados a el. Un ejemplo obvio es que cada directorio siempre tiene por lo menos dos nombres: el nombre con el cual el fue creado y el nombre ¨.¨ en el directorio mismo. El comando ln crea un nuevo vinculo a un archivo o directorio ya existente:

```
ln -[snf] archivo destino
```

Donde las tres opciones son las siguientes:

-s Simbólico o Soft-Link
-n No sobre escriba si archivo ya existen
-f Forzar sobre escritura si archivo ya existe (default)

Los vínculos permiten que un archivo sea llamado por un nombre diferente. Un simple ejemplo es el vi y el vim, si en tu linux ejecutas el comando vi lo mas seguro que estas en realidad ejecutando el comando vim y que lo estas invocando a través de un vinculo.

Existen dos tipos de vínculos: simbólicos (ln -s) y hard (duros; ln). Hard links son en la actualidad nuevas entradas de archivos en los directorios, mientras que vínculos simbólicos son punteros a directorios existentes.

Cada tipo de vínculo tiene su uso; y ninguno satisface todas las necesidades. Vínculos Hard solo pueden vincular archivos dentro del mismo sistema de archivos; mientras que los simbólicos pueden apuntar a archivos dondequiera que el usuario tenga acceso. A menudos los vínculos simbólicos son utilizados para apuntar en sistemas de archivos diferentes. El comando rm se utiliza para remover o sea para borrar archivos. Esta es una manera simplista de ver el comando. Este comando en lo actual no borra el archivo, solo remueve un hard link del archivo. Solo cuando el ultimo vinculo es removido es que el kernel borrara el archivo. En realidad el usuario nunca borra un archivo. El nombre original de un archivo no es especial y no es almacenado en el inode. Es posible remover el nombre original del archivo sin remover el archivo. De hecho Linux no sabe cual es el nombre original de un archivo; todos los nombres son iguales.

Similarmente, el comando mv no renombra los archivos; solo muevo los vínculos a través del sistema de archivos.

Todo esto debe ser considerado al trabajar con los soft links. Como el soft link es solo un puntero a un directorio, removiendo la entrada del directorio causa que el soft link no apunte ya al archivo original, produciendo una ruptura del vínculo. El archivo aun existe con un nombre diferente Talvez. Si fuese un vínculo del tipo hard siempre apuntase a la misma data.

Uso Común

Los vínculos son utilizados para ahorrar espacios o permitir que archivos sean nombrados por múltiples nombres. Utilizando vínculos, podemos organizar los mismos archivos en dos maneras diferentes: por Personas y por Sueldos. Imaginémoslo que cada persona es relacionada en una por su sueldo y entra por su nombre y mas aun que ambos son los mismos archivos. Solo que el archivo real es el nombre y el sueldo es solo un vinculo al archivo nombre.

Los comandos para crear esta estructura son:

$ mkdir departamento departamento/archivo
$ mkdir departamento/limpieza departamento/nomina
$ mkdir departamento/limpieza/2510 departamento/limpieza/2520 departamento/limpieza/2530

Creación de Archivos Vacíos

Los siguientes comandos son utilizados para crear archivos vacíos:

$ cd departamento/archivo
$ touch carta2510 carta2520 carta2530 repuestas2510 repuesta2520 repuesta2530

Crear Vínculos

$ ln archivo/carta2510 limpieza/2510
$ ln archivo/carta2520 limpieza/2520
$ ln archivo/carta2530 limpieza/2530

$ ln archivo/repuesta2510 nomina/2510
$ ln archivo/repuesta2520 nomina/2520
$ ln archivo/repuesta2530 nomina/2530

$ ln archivo/carta2510 2510/limpieza
$ ln archivo/carta2520 2520/limpieza
$ ln archivo/carta2530 2530/limpieza

$ ln archivo/repuesta2510 2510/nomina
$ ln archivo/repuesta2520 2520/nomina
$ ln archivo/repuesta2530 2530/nomina

Confirmando los Vínculos

Los siguientes comandos se utilizan para confirmar los Vínculos:

```
$ find limpieza nomina "25*" -print
./limpieza
./limpieza/2510
./limpieza/2520
./limpieza/2530
./nomina
./nomina/2530
./nomina/2510
./nomina/2520
./2510
./2510/limpieza
./2510/nomina
./2520
./2520/limpieza
./2520/nomina
./2530
./2530/limpieza
./2530/nomina
```

Mostrar los Vínculos

Los siguientes comandos se utilizan para visualizar los vínculos:

-i	**Incluir numero de inodes en el listado**
-l	**Mostrar los vínculos simbólicos y la cuenta de vínculos inodes**
-L	**Esconder vínculos simbólicos (seguir vinculo simbólicos al archivo original)**

```
$ touch usuario1
$ ln usuario1 usuario2
$ ln -s usuario1 usuario3

$ ls -il usuario [1-3]
230357 -rw-r--r-- 2 gnome2 ged2 0 jul 21 11:22 usuario1
230357 -rw-r--r-- 2 gnome2 ged2 0 jul 21 11:22 usuario2
230358 lrwxrwxrwx 1 gnome2 ged2 8 jul 21 11:22 usuario3 -> usuario

$ ls -ilL usuario3
230357 -rw-r--r-- 2 gnome2 ged2 0 jul 21 11:22 usuario3
```

Vínculos Duros (Hard Links)

Todos los atributos de los archivos creados con el comando de hard link son iguales, incluyendo su número de inode. La única diferencia es la referencia que usamos para indexar la tabla de inode a nombre de archivo. El número de nombres apuntando a la entrada del inode en la tabla se refleja en el campo de conteo del vínculo.

Solo existe un archivo real de data en disco y solo una entrada en la tabla de inode a lo que conocemos como dos archivos usuario1 y usuario2.

Vínculos Simbólicos

Un archivo por separado, usuario3, es creado, conteniendo un puntero al archivo original, usuario1. Entonces, existe un número de inode diferente, la "l" en la posición de tipo de archivo y un tamaño diferente del archivo que refleja el tamaño del archivo al cual apuntamos (no en tamaño de la data).

Los atributos del archivo usuario3, cuando se obtienen convencionalmente a través del comando ls -li, nos muestra los atributos del archivo puntero y no los del archivo original mismo. Si usas el comando ls -Lil, donde la -L significa seguir los vínculos, para obtener los verdaderos atributos.

ADMINISTRACION DE SISTEMAS DE ARCHIVOS

Los Sistemas de Archivos definen la estructura de los discos. Los discos en si se esconden del usuario. Todo acceso a los archivos es a través de la estructura de directorios y los permisos se utilizan para controlar acceso al sistema. Discos pueden ser formateados con diferentes sistemas de archivos y deben ser montados en la estructura de directorios para poder ser accesados. Un disco o una partición puede ser montada en cualquier directorio; este directorio es entonces referido como el punto de montaje.

El sistema de archivos de linux es una estructura de árbol invertida, con un directorio singular en la cima llamado raíz representado por una /. Todos los archivos y dispositivos son accesados a través de la estructura del sistema de archivos. Disco diferentes (ya sean fijos o removibles) y particiones con diferente o igual sistemas de archivos son mapeados en la jerarquía de archivos utilizando el comando mount.

El superbloque contiene la información acerca del sistema de archivos contenido en el disco o la partición. Esta información incluye una lista de los primeros bloques libres del disco y los inodes. Esta lista es suficiente para rendir las necesidades inmediatas del sistema operativo. En el momento que esta lista se agota el sistema operativo automáticamente la rellena pasando una inspección del listado de bloques libres o el listado de los inodes.

Los inodes definen los archivos en el sistema. Cada archivo es asignado un inode conteniendo toda la información pertinente al archivo excepto su nombre. El nombre es almacenado en el directorio que contiene el archivo.

El inode contiene la información de los diez primeros bloques asignados al archivo. Los números de los bloques subsiguientes se almacenan en un bloque separado señalado por el inode. Esta lista es llamada el bloque indirecto.

Los siguientes tópicos se discutirán en esta seccion:
- Tipos de Sistemas de Archivos
- Creando un Sistema de Archivos
- Montando un Sistema de Archivos
- Archivos de Configuración del Sistema de Archivos
- Espacio Libre en Disco
- Uso del Disco

Tipos de Sistemas de Archivos

Los sistemas de archivos tienen un tipo asociados con ellos. Los tipos soportados por un sistema Linux dependen de los que fueron compilados dentro del kernel en particular. Aunque Linux soporta un gran número de sistemas de archivos a veces no fueron compilados en el kernel por cuestión de ahorro de recursos. En práctica, las mayoría de sistemas solo implementan los sistemas de archivos mas comunes.

Un tipo de sistema de archivos define la estructura de data en el disco. Linux soporta muchos tipo de sistemas de archivos, es costumbre que un distro soporte doce o mas tipos de sistema de archivos. El estándar en todos los sistemas Linux es hoy día la Ext3 la cual recientemente reemplazo la Ext2. Todas distros incluyen soporte para Microsoft VFAT, FAT, FAT32 y NTFS (excepción de RedHat, aunque el kernel puede ser

recompilado para soportarlo) además de OS/2 y NFS.

Crear un Sistema de Archivos

El comando mkfs se utiliza para crear sistemas de archivos en discos y particiones. Creando sistemas de archivos es similar a formatear, como se demuestra en el siguiente ejemplo:

Formatear un disco como un disco de ext3:
 # mkfs -t ext3 /dev/hdb2 102400

Formatear un Disquete como un disco de DOS:
 # mkfs -t msdos /dev/fd0

En este ejemplo usamos -t para obviar el default del sistema de archivos. El comando requiere el nombre del dispositivo raw (/dev/fd0 y /dev/hdb2). Parámetros adicionales varían dependiendo en el tipo de sistema de archivos a implementar. El comando crea el superbloque y la lista de inodes. El listado de inodes es fijo en tamaño y no puede ser extendida, pero puede superar el número de inodes creados al inicio.

Si deseas cambiar el número de inodes, debes saber el tamaño físico del disco (bloques físicos) y convertirlo al número de bloques lógico. Típicamente, el bloque de un disco es de 512 bytes y el bloque lógico es de 1,024 bloques, entonces un disco de 64 MB en realidad tiene 64 Kb bloques lógicos (65,536) y 128 KB bloques físico (131,072). Utilice el número de inodes por defecto al menos que usted no este planificando crear muchos archivos bien pequeños o pocos archivos muy grande. Un tamaño de bloques de 2,048 le dará mejor rendimiento de disco pero desperdiciara espacio debido a bloques parcialmente llenos al final del archivo. Bloques de tamaño mas pequeños de 512 utilizan menos espacio pero en detrimento del rendimiento.

No tienes que crear el sistema de archivos del mismo tamaño que el disco. Cualquier espacio no asignado a un sistema de archivos puede ser asignado a otro sistema de archivos o puede permanecer sin asignación para su posterior uso.

Una vez un sistema de archivos ha sido creado, no se puede modificar su estructura sin destruir toda la data en el sistema de archivos. Para incrementar el número de inodes es requerido un back-up de la data, crear el nuevo sistema de archivos y restaurar la data del back-up.

Montar un Sistema de Archivos

Un sistema de archivos Linux solo puede ser accesado si esta montado en la jerarquía del sistema de archivos (el árbol). El directorio en el cual se monta el sistema de archivos es llamado el punto de montaje. Cualquier directorio puede actuar como el punto de montaje para cualquier sistema de archivos y sistemas de archivos montados pueden contener otros sistemas de archivos ya montados en ellos. Como al montar un sistema de archivos en un directorio esconde los archivos ya contenidos en el directorio de montaje, es costumbre montar los sistemas de archivos en directorios vacíos. El directorio /mnt se provee como un punto de montaje temporario y de referencia.

El comando mount es utilizado para agregar cualquier disco conteniendo un sistema de archivos a la estructura árbol de un sistema operativo.

```
# mount -t ext3 /dev/hda3 /mnt/disco
# ls /mnt/disco
descargas    video.mpg        Octave.pdf        PHP-MySQL -Tutoriales
```

Sistemas de archivos que no son requeridos pueden ser removidos del sistema de archivos utilizando el comando umount. Sistemas que contienen archivos en uso o que contienen otros sistemas de archivos mon-

tados en ellos no pueden ser desmontados. El comando fuser puede ser utilizado para determinar cuales procesos (y usuarios) están usando el sistema de archivos y puede ser utilizado para matar (KILL) estos procesos si es necesario.

```
# umount /mnt/disco
# ls /mnt/disco
```

Si el kernel entiende el formato (tipo) de sistema de archivos, entonces lo puedes montar en el sistema. Un sistema de archivos MS-DOS por ejemplo puede ser montado en un sistema Linux y ser manipulado transparentemente para el usuario. El mecanismo de mount es también el utilizado para manipular los sistemas de discos de redes, como lo son por ejemplo Network File System (NFS) y el Remote File System (RFS).

El comando mount, sin parámetros, lista los sistemas de archivos montados. El comando df muestra porcentajes de espacio en disco utilizado y disponible. El comando mount mantiene la información concerniente de los archivos montados en el archivo /etc/mtab. Si este archivo se corrompe dislocara los comandos mount y umount.

Archivos de Configuración del Sistema de Archivos

El archivo /etc/fstab es administrado por el administrador del sistemas y es utilizado para definir los discos a montar en sistema de archivos local. El guión de arranque y los comandos del sistema utilizan la información en este archivo para montar sistema de archivos automáticamente.

Entradas en el archivo /etc/fstab consisten en una entrada por sistema de archivos separada por campos de espacio o tab en la siguiente manera:

```
# /etc/fstab:
#
# Sistema data   Punto-Montaje  Tipo      Opciones                        dump pass
/dev/hda3        /              ext2      defaults,errors=remount-ro      0    1
/dev/hda1        none           swap      sw                              0    0
proc             /proc          proc      defaults                        0    0
/dev/fd0         /floppy        vfat      defaults,showexec,umask=022     0    0
/dev/scd0        /cdrom         iso9660   defaults,ro,user,noexec,noauto  0    0
/dev/hda2        /mnt/hda2      auto      noauto,user,exec                0    0
#/dev/hda3       /mnt/hda3      auto      noauto,user,exec                0    0
```

Utilice un menos (-) para los campos que no requieren los siguientes valores: no raw device, no fsck pass number, or no mount points. Líneas que empiezan con un símbolo de numero (#) son ignoradas hay utilizadas para comentarios de ayuda.

Los campos en el archivo /etc/fstab se listan en la siguiente tabla:

Campo	Descripción	Ejemplo
Block Device	Dispositivo de Bloque	/dev/hda1
mount point	Punto de Montaje	/home
file system type	tipo de Sistema de Archivos	Ext3
mount option	opciones de montaje	Auto,noexec
dump order	Orden de Vaciado	1 (yes) 0 0 (no)
Fsck order	1	(primero)

Espacio Libre en Disco

El comando df informa de la cantidad de espacio de disco usada y de la disponible en sistemas de archivos. Sin argumentos, df informa del espacio usado y del disponible en todos los sistemas de archivos monta-

dos actualmente (de todos los tipos). De otro modo, df informa sólo del sistema de archivos donde esté cada argumento archivo. Algunos ejemplos de opciones son: -m devuelve valores en megabytes, -k en kilobytes, -t si quieres referir a particiones por tipo de sistema de archivos.

El uso de df en sistemas nuevos, es muy util para tomar decisiones de asignacion; en un sistema funcionando estable, df, puede ser utilizado para monitorear uso del espacio en discos. Sistemas Linux modernos o sea versiones actualizadas incluyen utilidades ejecutandose bajo cron que informan cuando discos llegan a valores criticos configurable por el administrador (valores de 5 a 10%).

```
root@gnome2:/home/gnome2# df -k /dev/hda[1-2]
S.archivos  1k-blocks   Used      Available  Use%  Montado en
/dev/hda1   3043416     2702816   186000     94%   /
/dev/hda2   2887140     1988260   752216     73%   /mnt/hda2

root@gnome2:/home/gnome2# df -m -h /dev/hda[1-2]
S.archivos  Tamaño      Usado     Disp       Uso%  Montado en
/dev/hda1   2.9G        2.6G      181M       94%   /
/dev/hda2   2.8G        1.9G      734M       73%   /mnt/hda2
```

Disco en Uso

El comando du informa de la cantidad de espacio de disco usada por los archivos especificados y por cada directorio en las jerarquías cuyas raíces estén en los archivos especificados. Aquí, “espacio de disco usado” significa espacio usado por la jerarquía de archivos por debajo del archivo especificado. Sin argumentos, du informa del espacio de disco para el directorio en curso. Algunas opciones muy útiles son: - a lista todos los archivos, -s muestra solamente un total para cada argumento. La opción -h es muy útil ya que te da los resultados en formatos bien fácil de interpretar. Opción -k o -m te devuelve los resultados en kilobytes o en megabytes. Combinaciones múltiples son posibles como por ejemplo du -sch /dev/hdb3.

```
root@gnome2:/home/gnome2# du -s Charla/*
8   Charla/AlanTuring.sxw
4   Charla/esfera.txt
376 Charla/Hawking propone.sxw
40  Charla/INTRODUCCION-charla.doc
12  Charla/lapresentacion.sxw
336 Charla/personages.sxw
12  Charla/richardstallman.sxw
12  Charla/roboticaintro.sxw
4   Charla/scriptclameOctave
4   Charla/scriptclameOctave~
```

CUOTA EN DISCO

El sistema de archivos Linux implementa el mecanismo de cuota en disco. El administrador puede asignar cuota de espacio en disco a los usuarios, este control puede ser por el numero de inodes o bloques del disco que pueden utilizar en el sistema. Cuotas tienen dos tipos de limites: hard y soft. El límite denominado soft puede ser excedido por un periodo de gracia; desobediencia en reducir el espacio asignado antes que el tiempo se agote dispara la cuota. Limites Hard aplican inmediatamente. Los usuarios que excedan sus límites en espacio en disco son advertidos inmediatamente. Inclusive existen gatillos que se disparan cuando usuario se aproximan a sus cuotas (5 a 10%).

Este mecanismo puede ser activado y desactivado, a la interpretación del administrador. Una vez habilitado en un sistema de archivos puede ser aplicado a usuarios y/o grupos. Usuario no asignados cuota en disco no son incluidos en sistema. Es de especial importancia notar que un archivo creado por un usuario asignado cuota cuando cuota n esta habilitada no será contabilizado por el sistema de cuota.

Para iniciar el sistema de cuotas en un sistema de archivos, primero debe crear el archivo cuota (usuario, grupo o ambos) en el directorio raíz del sistema donde los usuarios tiene derecho a escribir (/home).

```
# touch /home/quota.user
# touch /home/quota.group
```

Editara las propiedades de las cuotas del grupo/usuario con el comando edquota.

```
# edquota ivelis
```

Para habilitar cuotas en un sistema de archivos lo hará con el comando quotaon:

```
# quotaon -v /home
```

Puedes habilitar cuotas para todo el sistema con la opción -a de esta manera:

```
# quotaon -a
```

Puedes deshabilitar cuotas en un sistema de archivos con el comando quotaoff:

```
# quotaoff /home
```

Los usuarios pueden examinar sus estados de asignacion de quotas, pero solo el root puede visualizar los niveles de quota de otros usuarios. A menos que un usuario no exceda sobre sus limites la salida del comando quota no retornara salida. Para informacion mas detallada podemos jecutar el comando de esta manera:

```
$ quota -v
Disk quotas for julio (uid 1006):
Filesystem  usage  quota  limit  grace  files  quota  limit  grace
/home       0      4222   6500          0      0      0
/Trabajos   3512   4222   6500          290    0      0
```

Para cada sistema de archivos que cuotas han sido definidas, el usuario actualmente en sesión recibirá una salida con los siguientes campos:

Sistema de archivos	Punto de Montaje del Sistema de Archivos bajo quota.
Uso	Monto de Bloques Usados
Quota	Numero de Bloques Permitidos (Soft)
Limit	Bloques Permitidos (Hard)
Grace	Aplicable solo si esta sobre su Limite
Archivos	Numero de Archivos Actualmente en Uso
Quota	Numero de Bloques Permitidos (soft)
Limit	Bloques Permitidos (Hard)
Grace	Aplicable solo si esta sobre su Limite

Ejercicio 5-2: Trabajar con Utilidades de Quota

Solución se proveen en el Apéndice A.

1.- Modifique esta línea del /etc/fstab para permitir cuotas para los usuarios y grupos:

```
/dev/hda1 /home ext3 defaults 1 2
```

2.- Examine esta salida del comando

```
# edquota -u ivelis:
Quotas for user ivelis:
/dev/hda1: blocks in use: 7502, limita (soft = 9600, hard = 12500)
inodes in use: 735, limits (soft = 2605, hard = 3600)
```

Escriba el comando para duplicar esta salida de cuota para todos los usuarios del grupo Operadores.

3.- ¿Cuál de los comandos el suite de cuotas produce listado para todas las cuotas en el sistema?

4.- ¿Cuál daemon permite obtener información de cuota de directorios montados remotamente?

Ejercicio 5-3: Sistema de Archivos

Solución se proveen en el Apéndice A.

Escriba los comandos para efectúar las siguientes tareas:

1.- Crear un sistema de archivos tipo ext3 en un disco de 200-MB hda3

2.- Monte este sistema de archivos recién creado en /mnt/hda3. Cree el directorio si no existe ya.

3.- Crear un sistema de archivos tipo ext2 en un disco de 150-MB hda5

4.- Monte este sistema de archivos en /mnt/hda5. Cree el directorio si no existe ya.

5.- Crear un sistema de archivos tipo Minix en un disco de 100-MB hda6 y montélo en /mnt/hda5/Minix. Cree el directorio si no existe ya.

6.- Desmonte los tres sistemas de archivos; Note como debe desmontar /mnt/hda5 antes de /mnt/hda5/Minix.

ARCHIVO DE CACHE DEL KERNEL

El kernel de Linux mantiene un archivo en memoria que utiliza como cache, en este archivo que se escribe al disco periódicamente, se incluye el superbloque, varios inodes y bloques de data. Esta cache mejora el rendimiento de Linux al reducir el acceso de data al disco ya que el kernel encontrara con frecuencia la data que busca en esta memoria cache. El contenido de este cache puede ser transferido a disco con el comando sync. Un método ya anticuado de apagar los sistemas Linux era con los siguientes comandos:

```
$ sync
$ sync
$ halt
```

El primer comando sync escribe la data al disco y el segundo comando sync le permite al (si el disco es lento) hardware del disco completar la operación de E/S. Y luego de estos dos comandos por ultimo podemos apagar seguro con el comando halt. El método moderno de hoy es con el comando shutdown, con las opciones -h para apagar y -r para reiniciar:

```
$ shutdown -h tiempo [mensaje de alerta a los usuarios que deseas mandarles]
$ shutdown -h now El sistemas se Apagara Ahora lo sentimos.
```

Si deseas practicar puedes utilizar la opción -k. Cuando un sistema se apaga inadecuadamente la data en cache se pierde y por esto el sistema de archivos se queda con ciertas inconsistencias. Para esto existe el comando fsck, comando que recupera la data perdida y la coloca en el directorio /lost+found, cual es el ultimo chance de recuperar la data perdida. En los sistemas linux que soportan el nuevo Journaling File System (JFS), la recuperación es automática, el sistema se recupera automáticamente al reiniciar.

Manipulando Sistema de Archivos Corrompido

El comando mkfs crea un directorio vació de nombre /lost+found para ser utilizado por el proceso fsck cuando recupera archivos en caso de corrupción del sistema de archivos. Como fsck no puede crear el direc-

torio /lost+found, ni puede aumentar el numero de bloques asignada en disco (recordemos que nuestro sistema de archivos esta corrompido), este directorio debe ser lo suficientemente grande para acomodar todos los archivos recuperados por el programa fsck. El directorio nunca tampoco puede ser borrado. Crear este archivo en el disco después del momento inicial por mkfs, es difícil encontrar por el tamaño considerable que hay que asignarle.

En cada ocasión que el sistema se inicia, el sistema de archivos es revisado por el fsck para verificar su integridad. Si el sistema de archivos no esta consistente, es modificado para volverlo aun estado estable. El directorio /lost+found entra en vigencia cuando el fsck encuentra inodes activas que no son referenciadas desde ningún directorio. Estos archivos perdidos son colocados en el directorio /lost+found. Al no existir entradas de directorios reflejando el nombre que se asignase a estos inodes, al ser almacenados se les asigna el nombre del número del inode. Puedes utilizar el comando file para idéntificar el tipo de archivo. Algunos archivos estaban tan corrompidos en el momento de recuperarlos que se perdían en su totalidad. Los directorios recuperados estarán vacíos.

```
root@gnome2:/home# /sbin/e2fsck -f -v /dev/hda2
e2fsck 1.27 (8-Mar-2002)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
9420 inodes used (1%)
17 non-contiguous inodes (0.2%)
# of inodes with ind/dind/tind blocks: 368/23/0
508601 blocks used (69%)
0 bad blocks
0 large files
8248 regular files
1156 directories
0 character device files
0 block device files
0 fifos
0 links
6 Symbolic links (6 Fast Symbolic links)
1 sockets
————
9411 files
```

La opciones -f es revisar aunque marcado clean/limpio y -v muy común en los comandos de unix significa verbose/mucha información a pantalla.

Ejercicio 5-4: Identificar los Archivos Recuperados

Solución se proveen en el Apéndice A.

Dado la siguiente información del directorio /lost+found:

```
# cd /lost+found
# file *
000973: ASCII text
000979: comands text
001256: iAPX 386 executable not stripped
001385: C source code
001576: data
```

¿Cuales comandos usarías para identificar el contenido de cada archivo?

1.- El archivo 00973
2.- El archivo 000979
3.- El archivo 001256
4.- El archivo 001385
5.- El archivo 001576

Ejercicio 5-5: Examinar y Revisar Sistemas de Archivos

Solución se proveen en el Apéndice A.

Escriba los comandos para ejecutar lo siguiente:

1.- Colocar el número máximo de veces encendido en /dev/hda1 antes de ejecutar fsck a 20.

2.- De dos comandos para revisar los bloques defectuosos (bad blocks) en /dev/hda2 (con 65,535 blocks)

3.- Muestre el progreso (sin la ejecución actual) para una revisión del sistema de todos los sistemas de archivos listados en el archivo /etc/fstab

4.- Ejecute una revisión de sistema de archivos en /dev/ hda3 con barra de progreso, especificando que el sistema de archivos es VFAT.

SISTEMAS DE ARCHIVOS DISTRIBUIDOS (DFS)

Un sistema de archivos distribuido almacena archivos en uno o más computadores denominados servidores y los hace accesibles a otros computadores denominados clientes, donde se manipulan como si fueran locales. Existen muchas ventajas en el uso de servidores de archivos: Los archivos están más accesibles si desde varios computadores se puede acceder a los servidores y compartir los archivos de una única localización es más sencillo que distribuir copias de los archivos a todos los clientes. Las copias de respaldo y la seguridad son más fáciles de manejar cuando sólo hay que tener en cuenta a los servidores. Los servidores pueden ofrecer un gran espacio de almacenamiento que sería costoso y poco práctico suministrar a cada cliente. La utilidad de un sistema de archivos distribuido se ve claramente cuando se considera a un grupo de empleados que tienen que compartir documentos. Por ejemplo, para compartir programas también es un buen candidato. En ambos casos, la administración del sistema se simplifica.

Sistemas de Archivos Distribuidos basados en Linux son utilizados para centralizar la administración de discos y proveer la facilidad de compartir archivos transparentemente en la red. Paquetes de Linux que proveen Dfs casi siempre incluyen el cliente y el server. Un servidor Dfs comparte archivos locales en la red; un cliente Dfs monta archivos compartidos localmente. Un sistema Linux puede ser tanto un cliente, un server, o ambos dependiendo en cual comando fue ejecutado.

NFS fué desarrollado por Sun Microsystems en el 1984 y licenciado a muchos vendedores. Esencialmente permite a servidores hacer disponible su sistema de archivos y directorios, típicamente aplicaciones y los directorios home, a clientes que montan desde el server. El cliente entonces ve la data como si fuese local; el montaje debe ser transparente para el cliente. NFS elimina la necesidad de instalar aplicaciones y múltiple copias de data idéntica en más de un computador.

Samba (Server Message Block, or SMB) fue desarrollado por Andrew Tridgell en 1991. Desarrollado para lograr que su PC, corriendo un Xserver en estatus de beta requisiera un protocolo de red (basado en NetBIOS) propiedad de la DEC, hablase con su estación de trabajo. Oficialmente el inicio NetBIOS para unix

a finales del 1993. Su software es el día de hoy incluida en casi toda distribución de Linux.

Discutiremos los siguientes tópicos:
- Un vistazo al NFS
- El Stack del Protocolo NFS
- Un vistazo al Samba
- El cliente NFS
- El servidor NFS
- Seguridad NFS

Análisis del 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. RedHat 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 RedHat Linux.

El Protocolo NFS

Visto desde el punto de vista de la pila, el NFS es una aplicación. Su capa de presentación es el XDR (External Data Representation) y la de sesión es el RPC (Remote Procedure Calls). Del lado del cliente, montamos el server utilizando en comando mount del sistema operativo. Del lado del server, el daemon mountd responde a estas llamadas, permitiéndolas 0 no.

La data es representada en una forma totalmente independiente y transparente de sistema operativo a través utilizando XDR y por esto es que NFS esta disponible en Linux, OS/2/ NetWare/ MVS, DOS y muchos más.

En los momentos que el cliente desea comunicarse con el server, el genera un RPC y el Daemon del Server (que siempre esta escuchando) responde a la llamada del cliente. Cierre de los record es implementada (Record locking) utilizando los servicios de los daemons lockd y statd, los cuales deberán estar ejecutándose en ambos el cliente y el server.

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 ser-

vidor o cuando se monten en un cliente. Haciendo esto reducirá los riesgos de seguridad NFS y protegerá mejor los datos en el servidor.

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 sólolectura 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 meta-caracteres 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 RedHat Linux.

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 podrán 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 RedHat 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.

Documentación Instalada

Existe documentación ya instalada en los sistemas GNU/Linux en el directorio /usr/share/doc/nfsutils<

version-number>/-Reemplace<versión-número> 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 -	Opciones comunes usadas en el archivo /etc/exports cuando exportamos sistemas de archivos NFS.

El Cliente y Opciones de Montar

Aparte de montar un sistema de archivos vía NFS con el comando mount 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 0 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.
rw/ro -	Read/Write Lectura y escritura o Read/Only solo lectura.
bg -	Si el primer intento de montar falla, los próximos intentos hazlo en el background.

Hay muchas más opciones en la página del manual de mount, incluyendo opciones para montar sistemas de archivos que no sean NFS.

El siguiente es un ejemplo del comando montar:

 # mount -t nfs -o bg, soft, intr nombre_servidor:/directorio/exporta /punto/montar

Este comando mount del cliente, contacta al rpcbind del server (portmapper) y le pide que en cual puerto esta el mountd escuchando. Este puerto ha de ser visto como la direccion de una aplicación. Una vez esto sucede el comando mount es aceptado o denegado, basado en los derechos definidos en el archivo de configuración /etc/exports.

Si la petición es aceptada por el mountd el server le pasa al cliente un identificador llamado un file handler, el cual es colocado por el kernel del cliente en su tabla de sistemas de archivos montados. En cualquier momento que el cliente trata de accesar este volumen el cliente simplemente le pasa este archivo (se puede ver como una llave) al server, indicandole que esta tratando de accesar data en este volumen.

Si desea montar volúmenes NFS desde el arranque del sistema, simplemente edite el archivo /etc/fstab, como lo hace con los volúmenes locales.

Asegurar NFS y el Servidor

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.

Para que un sistema GNU/Linux se convierta en un Servidor NFS, solo necesita estar ejecutando los siguientes servicios:

/sbin/portmap	El Mapaedor de puertos
nfsd	Daemon NFS
mountd	Daemon de mount

El sistema de archivos a compartir debe estar definido en el archivo /etc/exports. Los archivos guión (scripts) rc en /etc/rc.d/rc# deben ser modificados para que nfsd y mountd se ejecuten en el momento d inicio del server, si no deberemos iniciarlos manualmente. Aquí les damos un ejemplo de una maquina llamada cliente el permiso de Read/Write sobre el sistema de archivos a exportar desde el server en la ruta /mnt/compartidos/por/nfs

```
# cat /etc/exports
/mnt/compartidos/por/nfs cliente (rw)
```

Acceso al Sistema NFS

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 meta-caracteres 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.

Nota: Para más información sobre la seguridad en NFS, consulte el capítulo titulado Seguridad del servidor en el Manual de seguridad de RedHat Linux.

Analizando SAMBA

El protocolo SMB es usado por Microsoft Windows 3.11, NT y 95 para compartir discos e impresoras. Usando el paquete de herramientas Samba creado por Andrew Tridgell, las máquinas UNIX (incluyendo Linux) pueden compartir discos e impresoras con servidores Windows.

Hay cuatro cosas que uno puede hacer con Samba:
- Compartir una unidad de Linux con máquinas Windows.
- Compartir una unidad de Windows con máquinas Linux.
- Compartir una impresora de Linux con máquinas Windows.
- Compartir una impresora de Windows con máquinas Linux

Se requieren los dos demonios siguientes para el paquete Samba. Que suelen instalar en /usr/sbin y se pueden ejecutar tanto desde los scripts de arranque del sistema como desde inetd. Algunos scripts de ejemplo los puedes ver en Ejecutando los demonios.
- smbd (El demonio de SMB)
- nmbd (Provee un nameserver de NetBIOS para soporte de clientes)

Habitualmente, se instalan en /usr/bin los siguientes ejecutables de Samba, aunque la localización (como de costumbre) es opcional.

smbclient	(Un cliente SMB para maquinas UNIX)
smbprint	(Un script para imprimir a una impresora en un servidor SMB)
smbprint.sysv	(Como el de encima, pero para máquinas UNIX SVR4)
smbstatus	(Lista de las conexiones SMB en marcha en el servidor local)
smbrun	(Un script 'cola' para facilitar la ejecución de aplicaciones en servidores)

Los dos demonios de SMB son /usr/bin/smbd y /usr/sbin/nmbd. Puedes ejecutar los demonios de Samba desde inetd o como procesos independientes. Si estás configurando un servidor de archivos permanente, deberían ejecutarse desde inetd para que sean reejecutados si 'mueren'. Si solo quieres usar los servicios SMB de vez en cuando o como ayuda a la administración del sistema, puedes ejecutarlos con un script en /etc/rc.d/init.d o incluso a mano cuando los necesites.

Para ejecutar los demonios desde inetd, escriba las siguientes líneas en el archivo de configuración de inetd, /etc/inetd.conf:

```
# Servicios SAMBA NetBIOS (compartición de archivos e impresoras en PC)
netbios-ssn stream tcp nowait root /usr/sbin/smbd smbd
netbios-ns dgram udp wait root /usr/sbin/nmbd nmbd
```

Entonces reejecuta inetd con el siguiente comando:

```
kill -HUP  1
```

La configuración de Samba en un Linux (u otra máquina UNIX) es controlada por un solo archivo, /etc/smb.conf. Este archivo determina qué recursos del sistema quieres compartir con el mundo exterior y que restricciones deseas poner en ellos. Aquí les damos un ejemplo:

```
; /etc/smb.conf
; Reinicia el servidor cada vez que hagas cambios a este archivo, ej:
; /etc/rc.d/init.d/smb parar
; /etc/rc.d/init.d/smb empezar
```

```
[global]
    ; Quita el comentario a la siguiente línea si quieres cuentas de invitado
    ; guest account = nobody
    log file = /var/log/samba-log.%m
    lock directory = /var/lock/samba
    share modes = yes
[homes]
    comment = Directorios principales
    browseable = no
    read only = no
    create mode = 0750
[tmp]
    comment = Espacio de archivos temporales
    path = /tmp
    read only = no
    public = yes
```

Como se muestra en el archivo smb.conf anterior, compartir una unidad Linux con usuarios Windows es fácil. De todas maneras, como todo lo demás con Samba, puedes tener las cosas MUY controladas. Aquí tienes unos pocos ejemplos:

Para compartir un directorio con todo el mundo, crea una copia de la sección [tmp] añadiendo algo como esto al smb.conf:

```
[public]
    comment = Cosas publicas
    path = /home/public
    public = yes
    writable = yes
    printable = yes
```

Para que este directorio lo pueda leer todo el mundo, pero que sólo lo puedan cambiar gente del grupo 'laborales', modifica la entrada de esta manera:

```
[public]
    comment = Cosas publicas
    path = /home/public
    public = yes
    writable = yes
    printable = no
    write list = @laborales
```

Para aprender otros truquillos con que jugar con las unidades compartidas, mira la documentación de Samba o las páginas del man.

Ejercicio 5-6: Uso del Comando mount con NFS

Solución se proveen en el Apéndice A.

Escriba los comandos para ejecutar lo siguiente:

1.- Con que comando montaria usted el directorio /usr/share desde el equipo abiertos12 en el punto de montaje local /mnt/disco con la opcion de background y timeout ?

2.- Complete las opciones de mount para los siguientes dos archivos /etc/fstab:

```
# grep home /etc/fstab
ivelis:/home/ivelis /home/ivelis nfs 0 0
```

```
# rsh ivelis grep home /etc/fstab
fcld12:/home/ivelis /home/ivelis nfs 0 0
```

RAID

Antes de nada, debemos saber qué significan las siglas RAID y si realmente va a merecer la pena su inclusión dentro de nuestro querido GNU/Linux: las siglas RAID provienen de Redundant Array of Inexpensive Disks, lo que quiere decir "arreglo redundante de discos baratos", aunque, como veremos después, ni el array tiene que ser redundante ni los discos deben ser baratos.

RAID nos provee de un mecanismo para crear un único disco virtual a través de un conjunto de discos independientes, pudiéndose ganar en velocidad de acceso y/o seguridad en nuestros datos ante un fallo eventual de alguno de nuestros discos. Se puede implementar por hardware o en software, teniendo cada uno sus ventajas y desventajas:

Si la implementación se hace por hardware, necesitamos una controladora que lo incluya, las cuales no suelen ser nada baratas, además de tener que crear el array directamente con discos físicos, lo que va a limitar su campo de acción.

La implementación por software nos permite mucha más flexibilidad en su uso, pudiéndose crear los arrays sobre particiones en lugar de discos, pero, en contrapartida, este tipo de acceso a los discos consume más tiempo de procesador.

Discutiremos los siguientes tópicos a continuación:

- Niveles de RAID
- RAID Hardware
- RAID Software

Niveles de RAID

Hay cinco tipos de arrays de discos:

Linear: No es un tipo de RAID, pero al elegir esta opción Linux nos permite unir dos o más discos haciendo coincidir el final de uno con el principio del siguiente, simulando tener un disco de una capacidad igual a la que suman los discos unidos. Por supuesto, este método no proporciona ningún tipo de protección contra fallos de cualquiera de los discos, ni tampoco acelera el acceso a ellos, con lo que no es una opción muy interesante en la práctica.

RAID-0: Es semejante a linear, en el sentido de que nos permite unir discos sin proporcionar ningún mecanismo de protección contra fallos; sin embargo, utilizando este método se consigue una aceleración en el acceso a dichos discos, tanto en escritura como en lectura, mediante el entrelazado de ambos discos, el llamado data stripping, que va a ser una constante dentro de los tipos de arrays disponibles. Es, sin duda, el tipo de array más rápido.

RAID-1: También conocido como disk mirroring. Consiste en la unión de un par de discos idénticos para crear un único disco con la capacidad de uno de ellos, conteniendo ambos la misma información. Este mecanismo no conlleva el entrelazado de los discos, pero se obtiene un aumento del rendimiento en lectura debido a que se pueden leer ambos discos independientemente. Este tipo de RAID es el que mayor rendimiento proporciona dentro de los tipos redundantes, aunque a la vez es el método que más desaprovecha el espacio en los discos, ya que se utiliza realmente la mitad de la inversión en los mismos.

RAID-4/5: El funcionamiento de este tipo de array se basa en añadir a los datos almacenados en los discos un control de paridad que hará posible reconstruir la información perdida en caso de que alguno de los discos falle. La capacidad de un array de este tipo es de (N-1), siendo N el número de discos incluidos en el

array. La velocidad de lectura de estos datos es similar a la obtenida con RAID-1, pero se produce una degradación en la velocidad de escritura, ya que al tiempo empleado en la escritura de los datos en sí, hay que añadir el de lectura de los datos de paridad para recalcular y escribir nuevamente.

Equipos/Hardware RAID

Los sistemas de RAID basados en soluciones hardware manejan el subsistema RAID independientemente del sistema operativo, al que le presentan un solo disco por matriz. Un ejemplo de RAID por hardware podría ser uno que se conectase a una controladora SCSI y presentase las matrices de discos como un solo disco SCSI. Una caja externa contiene el controlador que proporciona toda la «inteligencia» de manejo del RAID que se encuentra en el subsistema externo de disco. El subsistema completo se conecta al ordenador principal a través de un controlador SCSI que es visto por el ordenador como un solo disco.

Algunos controladores de RAID consisten en tarjetas que se comportan como una controladora SCSI hacia el sistema operativo y manejan por su cuenta toda la comunicación con los discos. En ese caso, conectará los discos en la controladora del RAID del mismo modo que lo haría en una controladora SCSI, pero luego los añadirá a la configuración del controlador RAID y el sistema operativo nunca se enterará de la diferencia.

RAID SoftWare

Los RAID por software implementan los diferentes niveles de RAID en el código de dispositivo de bloque del núcleo. También proporcionan la solución más barata: No solamente porque prescinden de costosas controladoras de disco o chasis de intercambio en caliente, además los RAID por software funcionan tanto con discos IDE más baratos como con SCSI. Con los rápidos procesadores actuales, un RAID por software ofrece un rendimiento excelente comparado con uno por hardware.

El manejador de dispositivo MD en el núcleo Linux es un ejemplo de solución RAID que es completamente independiente del hardware. El rendimiento de una matriz RAID basada en software depende mucho del rendimiento y la carga del procesador.

Razones para usar un RAID:

- Velocidad
- Mayor capacidad de almacenamiento
- Mejor capacidad para recuperarse del fallo de un disco.

Debido a que los cambios son muchos, no es fácil relacionarlos todos. Brevemente, algunos de ellos son:

- Proceso de reconstrucción multihilo
- Configuración completamente basada en el núcleo
- Las matrices pueden cambiarse a otros computadores Linux sin reconstruirlas.
- La reconstrucción de matrices se efectúa en segundo plano, usando el tiempo inactivo del sistema
- Posibilidad de intercambio de discos en caliente
- Detección automática de tipo de procesador, para aprovechar ciertas optimizaciones

RESUMEN

En este capítulo usted fue introducido a los diferentes aspectos de la administración del espacio en disco y cuotas. Esto incluye:

- El Sistema de Archivos Linux de estructura de árbol jerárquico con un nodo singular denominado root
- El diseño del disco es ocultado del usuario y los archivos en discos son identificados por un número unico llamados inodes.
- Los directorios mapean nombres a números de inode que permiten crear vínculos de archivos.
- Acceso al sistema de archivos es manejado por un sistema de permisos.
- GNU/Linux tiene una estructura estándar de directorios utilizada por la mayoria de los usuarios.
- La estructura de linux esconde la distribución geométrica del disco.
- Los discos son utilizados para almacenar sistemas de archivos que son creados con el comando mkfs.
- Una gran cantidad de sistemas de archivos son soportados por GNU/Linux.
- Los discos tienen que ser montados dentro de la jerarquía de GNU/Linux para poder ser utilizados.
- Puntos de montajes son almacenados en /etc/fstab
- El kernel optimiza acceso a disco con un archivo cache en el cual mantiene información de inodes y data blocks.
- Un disco conteniendo un sistema de archivos sin consistencia no puede ser montado hasta no haberlo reparado con fsck
- GNU/Linux da soporte a compartir archivos a través de la red.
- El sistema más popular de compartir archivos de Linux a otros sistemas operativos es NFS de Sun.
- Clientes deben usar el comando mount para accesar volúmenes compartidos.
- Samba es la manera más popular de compartir archivos y recursos de Linux/Unix a Windows.
- Los clientes utilizan el comando smbclient para accesar recursos compartidos de Windows o a través del protocolo SMB.
- RAID provee almacenaje a prueba de fallos a través de duplicación de data en discos en una manera transparente.

PREGUNTAS POST-EXAMEN

Las respuestas a estas preguntas se proveen en el Apéndice A.

1.- ¿Qué debes hacer para usar disco para almacenar archivos en Linux?

2.- ¿Cuál es el proceso de borrar archivos con el comando rm?

3.- ¿Cuál es la diferencia entre quotas HARD y SOFT?

4.- ¿Para qué se usa SAMBA?

5.- ¿Para qué se utiliza el cache del KERNEL?

ADMINISTRACIÓN DE USUARIOS

OBJETIVOS

Al completar este capítulo, usted podrá:

- Crear diferentes cuentas de usuarios en Linux.
- Administrar cuentas de usuarios y grupos y archivos de sistemas relacionados.
- Configurar seguridad a nivel de usuarios y sistema.
- Personalizar y utilizar el ambiente del shell.
- Modificar los entornos del usuario y las variables del sistema.
- Como configurar el ambiente del usuario.
- Estar familiarizado con los diferentes archivos bash de usuario y sistema.
- Identificar los archivos principales en la configuración de los perfiles de usuarios.
- Detallar los pasos para agregar y remover usuarios.
- Describir el comando motd y su rol en la comunicación con los usuarios.

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1.- ¿Para manualmente crear una cuenta de usuario desde la línea de comandos, que archivos hay que editar?

2.- ¿Cómo puede un usuario agregar su teléfono al archivo /etc/passwd?

3.- Cuando un usuario nuevo es creado, algunos archivos son creados en su directorio home. ¿De dónde son estos archivos copiados?

4.- ¿Nombre los archivos de configuración que se pueden ejecutar en el momento de login de shell bash?

INTRODUCCION

Para que los nuevos usuarios puedan accesar un sistema ejecutando Linux se requiere una cuenta de usuario y un ambiente de trabajo ya creado. El administrador del sistema es responsable por estos pasos, esto incluye todo lo necesario para crear estas cuentas hasta la colocación del password del usuario la cual se almacena en el archivo /etc/passwd. Por razones de seguridad de ambos el administrador y el usuario, el passwd en si es almacenado en el archivo /etc/shadow. Este archivo almacena los passwd de todos los usuarios del sistema. Similarmente, los nombres de los grupos se almacenan en el archivo /etc/group y los passwords de los grupos en /etc/gshadow. Estos archivos los discutiremos mas adelante en este capítulo.

Un buen administrador debe proveer un buen perfil (profile) y guión (script de login) de ingreso para los nuevos usuarios, estos archivos plantillas se almacenan en /etc/skel y al momento de crear los usuarios su contenido es copiado al home del nuevo usuario. Por ejemplo si el usuario se llama ivellise entonces sucede lo siguiente: se crea un nuevo directorio en /home/ivellise el cual en realidad es una copia de /etc/skel y renombrada con el nombre del nuevo usuario en este caso ivellise. También el archivo de saludo a los usuarios al ingresar se controla con el archivo /etc/motd, aquí se puede incluir información del tipo de bienvenida y cuestiones de en caso de problemas a quien referirse.

Los pasos y las utilidades necesarios para llevar a cabo la creación de una nueva cuenta de usuario son simples de utilizar. La parte mas importante en agregar usuarios a un sistema es determinar su rol y insertarlo en una categoría de permisos y privilegios en la disponibilidad de servicios y recursos del sistema. La decisión de a cuales grupos pertenecerá, su numero de identidad en el sistema y el nombre de su usuario para ingresar.

USUARIOS Y GRUPOS

El control de los usuarios y grupos es un elemento clave en la administración de sistemas 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. El archivo /etc/group tiene el siguiente formato y contenido:

```
$ cat /etc/group
root:x:0:miguel
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:miguel
tty:x:5:
```

disk:x:6:miguel
lp:x:7:lp

Vamos a identificar cada campo:

Campo	Descripcion
x:	Campo del passwd, contraseña.
gid	ID Numerica
User1,user2	Lista usuario los usuarios permitidos acceso secundario a este grupo.

El Archivo /etc/passwd

Todo usuario del sistema debe tener una entrada en este archivo, para así poder acceder al sistema. Cada entrada define un nombre de login, un UID. Los otros campos tienen el password encriptado el directorio home, el programa de login y opcionalmente el nombre completo del usuario. El nombre del usuario (login) debe ser unico en el sistema, históricamente el archivo password contenía el password encriptados, pero en las ultimas versiones de linux esta convención a cambiado. Si su sistema tiene estos archivos, tendrían que correr el comando pwdconv en orden de poder mover su password al archivo /etc/shadow.

root:x:0:3:0000: -admin(0000): /root:/sbin/bash

El UID debe ser unico. Tenga mucho cuidado como administrador de que no haya un duplicado con el numero 0, esto puede significar que hay dos cuentas con el poder de superusuario. El programa a correr no necesariamente, tiene que ser un shell, podría ser por ejemplo una interfaz de base de datos: /bin/mysql.

Herramientas de Administración de Usuarios y Grupos

La gestión de usuarios y grupos ha sido tradicionalmente tediosa, pero en versiones recientes, 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 (ejemplos son: redhat-config-users, gnome users-admin, kde kusers). Las siguientes herramientas de línea de comandos también se pueden utilizar para manejar usuarios y grupos:

useradd, usermod, userdel-	Métodos estándar de la industria para añadir, eliminar y modificar cuentas de usuarios.
groupadd, groupmod, 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.

Cambiar los Atributos del Usuario

Esta herramienta se utiliza para cambiar el UID de un usuario, aunque esto no es recomendado. Note que el comando usermod cambiara el UID de los archivos almacenados en el directorio home del usuario, pero no cambiara los demás archivos (los del cron, del correo, etc.) aunque el valor será insertado en el archivo /etc/passwd.

El programa usermod puede ser utilizado para cambiar los atributos de vencimiento de cuenta, por ejemplo la fecha de expiración, la advertencia de vencimiento y fechas absolutos cuando la cuenta termina.

 # usermod -f 10 miguel

La opción -f indica que tiene derecho a 10 días de inactividad antes de cancelar la cuenta de forma permanent.

#usermod -e 07/29/2003 miguel

Indica la fecha de expiración de la cuenta de miguel.

Usuarios Estándar

La siguiente Tabla 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.

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
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

Grupos Estándar

La siguiente tabla lista los grupos estándar configurados por una instalación con "Todo". Los grupos son almacenados 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	
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
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

Grupos de Usuario Privado

GNU/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.

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

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

 # /usr/bin/gpasswd -d <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.

Ejercicio 6-1: Agregar y Modificar Usuario

Solución se proveen en el Apéndice A.

1.- Agregue un usuario llamado miguel

2.- Agregue un usuario llamado ivellise y especifique el shell sh.

3.- Agregue un usuario llamado alex, usando /home/alex, como su home directorio

4.- Agregue un usuario llamado peque especificando su UID de 400 en el grupo de staff

5.- Modifique el usuario alex para que pueda usar el shell bash

6.- Modifique el usuario alex para conseguir un nuevo UID de 401

CONTRASEÑA (PASSWORD)

Se le asigna el password a un usuario con el comando passwd. En la mayoria de los sistemas, el administrador, le permite a usuarios opcionalmente cambiar su password para así forzar a que el usuario la cambie en intervalos de tiempo a su preferencia. Estos usuarios del sistema pueden utilizar el comando password para cambiarla. Pero solo el root puede cambiar el password sin conocer la anterior. Los usuarios a menudo olvidan su contraseña y pueden ir a donde su administrador y pedirle que la cambie. El root puede asignar una contraseña nueva pero no puede obtener la contraseña anterior. Si el password tiene una fecha de expiración, el sistema le requerirá la próxima vez que el ingrese cambiar el password.

Cambiar la Contraseña (CONTRASEÑA)

Usuario muy a menudo eligen password inapropiada o demasiado simples. La mayoria de los sistemas insertan tener una cantidad mínima de seis (6) y enforzan a que la palabra no este contenida en ningún diccionario. Los siguientes ítems te ayudaran a colocar passwords más seguros:

- No utilice nombres
- Use letras y dígitos
- Incluya símbolos
- No utilices tu nombre de usuario como password

La siguiente es una lista de la política que te puede ayudar a mantener la seguridad y vida de tu sistema:

- Exija un número mínimo de caracteres
- Exija el uso de no alfanumérico
- Mantenga un diccionario actualizado de palabras no permitidas
- No use cuenta de invitado/guess
- Si alguien tiene que usar tu sistema, dale una cuenta con un password
- Utilice las características de vencimiento de password.

Contraseñas Shadow

En entornos multiusuario es muy importante utilizar contraseñas shadow (encriptadas) (proporcionadas por el paquete shadow-utils). Haciendo esto se mejora la seguridad de los archivos de autenticación del sistema. Por esta razón, la mayoria de las instalaciones de Linux activan 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 las 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 al 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

El Archivo /etc/shadow

Aunque el archivo /etc/password esta encriptado, aun existen riesgo de seguridad. Este archivo permite, que puedan ser leídos por todo usuario del sistema. Un usuario malicioso puede copiarlo a su directorio home y aplicarle utilidades que desencriptan a la fuerza las contraseñas. Los Software avanzado utilizados para romper los archivos encriptados, son muy eficientes.

La representación encriptada del password es almacenada en el archivo /etc/shadow, este archivos no contiene los UIDs. La relación entre el archivo password y el shadow, es usuario posición. Por esto es muy importante que estos archivos mantengan su orden si son editados a mano. En este archivo shadow es que se mantiene la información de expiración de las contraseñas.

Nombre: password (contraseña encriptada):cant_ min_caracteres:cant_max_caracteres:warning_de tiempo: tiempo_inactividad:tiempo_expiracion: flag

Es importante mantener una alta seguridad a este archivo, especialmente en sistema antiguos, que utilizaban encriptación de 56bits, la cual es mucho más fácil de descubrir, porque los hackers maliciosos utilizan cada día equipos mas sofisticados.

Seguridad de las Cuentas

Los usuarios representan el riesgo más alto para un sistema linux. Para mejorar la seguridad es necesario aconsejar a los usuarios del uso sensibles de las contraseñas. Note que solo los usuarios con la contraseña de root, pueden tener accesos a los archivos de encriptados de contraseñas. Aun con contraseñas encriptados es posible que un empleados con accesos a esta contraseñas, disgustado pueda tener acceso al sistema operativo y causar daños. Aunque los sistemas modernos utilizan encriptaciones de 128 bits, superior a los de 56, no quiere decir que la encriptación no pueda ser violada.

La siguiente es una lista de prácticas exitosas de administradores de sistema:

- Cuando un miembro de staff administrativo, se marcha, cambie los password a los que el tenia acceso.
- Cambie todas las contraseñas.
- Nunca tenga copias escritas de las contraseñas del root; NUNCA.
- Permita el acceso a la contraseña del root a los menos usuarios posibles.
- Cambien la contraseña del root regularmente.
- Siempre informe a los usuarios y a los equipos del cambio de contraseña.

Ejercicio 6-2: Seguridad de Cuentas de Usuarios

Solución se proveen en el Apéndice A.

1.- Agregue un passwd a miguel

2.- Exija a miguel que cambie su contraseñas, la próxima vez que se haga un login

3.- Habilite característica de vencimiento al passwd de alex (min. 21, max 31, advertencia 7)

4.- La fecha de expiración de ivellise a la fecha 24 de abril del 2003-07-24

5.- Cierre (bloque) la cuenta de ivellise.

6.- Vuelva a abrirla (desbloquearla).

ELIMINAR USUARIOS

Cuando un usuario no es requerido en el sistema debe ser removido. El administrador debe limpiar las cuentas y el sistema de correo, es mejor dejar los archivos de registro, para que su UID y su nombre de ingresos no sean reasignados. Archivo no necesario deben ser removido, pero la información requerida por otros usuarios, que era mantenida por el usuarios saliente, debe ser recuperada.

Eliminar la Cuenta de un Usuario

Los archivos de los usuarios deben ser almacenados por un corto tiempo para asegurarse de no borrar de manera permanente archivos, importantes. El correo del usuario debe ser reenviado a una cuenta de un administrador. Alternativamente, el administrador puede usar el comando vacation para enviar un mensaje a esos correos informándoles de la situación y que hacer al respecto. También debe revisar si el usuario dejo tareas ejecutadas programadas con cron o at. Si remueve los archivos de un usuarios con el comando find también borrara estas tareas, lo siguiente es una lista de los pasos sugeridos para borrar este usuarios.

Expira la cuenta y clausura la contraseña
 # chage -E /fecha-hoy(25/07/2003) alex

Cree y proteja el directorio almacenar los archivos del usuarios
 # makedir /almacenar; chmod 000 /almacenar

Cree un archivo comprimido de todos los archivos del usuarios (formato cpio). Note que hemos cambiados a root y ejecutados el comando find en el directorio actual (.) deliberadamente, para ahorrarnos rutas relativas.

Esto no asegurara que cuando restauremos archivos para inspeccionarlos, ellos no serán insertados en un lugar diferente.
 # cd /; find . –user alex -print | cpio -ov | compress > /almacenar/alex

Ahora que hemos salvados los archivos, podemos borrarlo, note que no hemos trabajado con ningún directorio, que sean del usuario, pero que contenga archivos que le pertenezcan a otros.
 # find . –user alex -type f -exec rm -f {} \;
 # find . –user alex -type d -exec rm -f {} \;

El correo entrante puede ser redireccionado (enviado) a otro usuario. Esto funcionara aunque hayamos borrado la cuenta del usuario del sistema.
 # su - alex -C “mail –F admin”

En práctica los administradores utilizan el comando userdel ya que este puede ser utilizado para borrar los directorios home, pero dejan otros, como ejemplo el correo, intacto. El comando para borrar a alex seria:
 # userdel -r alex

Ejercicio 6-3: Administrar los Usuarios

Solución se proveen en el Apéndice A.

1.- Utilice el comando useradd para agregar el usuario juan (nombre completo Miguel Antonio) con el UID de 355. No olvides utilizar la opción –m para crearle su directorio home.

2.- Corregirle el nombre a miguel a Miguel Antonio y agrégale su shell de /bin/bash.

3.- Como root, utiliza el comando chage -l para mostrar el estatus de la protección del password de juan y cambiárselo a lo siguiente (Max numero de días = 7, Min número de días = 2, Advertencia numero = 7).

Luego ingresa como miguel y trata de cambiar el password. Si no puedes, su a root y arregla el problema. Salga del shell de root y cambia el password de miguel.

4.- Cree un nuevo grupo y llámelo “estudiantes”. Modifique a juan que sea un miembro del grupo estudiante; pero no modifique es grupo default de juan. En un comando, agregue un nuevo usuario ivelis con su nombre completo de Ivelis A González, UID 359 y miembro suplementario del grupo estudiante (no cam-

bies el default de su grupo). Agréguele el password a luis y salga del shell. Ingrese como ivelis y como el, entre los siguientes comandos:

```
$batch
date
^ D
```

Esto se asegurara que exista un mensaje de correo para ivelis. No lea este mensaje aun; queremos que permanezca el la caja de correo.

5.- Remueva la cuenta ivelis, incluyendo su directorio home. Ahora busque si ivelis es propietario de algunos archivos en el sistema.

Utilice el comando useradd para crear de nuevo el usuario ivelis con los mismos parámetros de antes.

Usted no podrá porque el sistema no le permitirá reusar el UID de ivelis (digamos que el 421) por otros 20 días.

¿Qué opción de comando debieras especificar para asegurarte que reuse este UID de 421?

6.- Liste todos los grupos de los cuales root es miembro (no olvide el grupo defecto).

Ejercicio 6-4: Administrar home de Usuarios y Directorios

Solución se proveen en el Apéndice A.

1.- Cree un directorio de nombre /home/esqueleto, el cual contiene todos los archivos de /etc/skel (con todos los archivos incluyendo los ocultos). Cree dos directorios vacíos llamados libros y mascotas en /home/esqueleto. Cree un nueva cuenta para mike (nombre completo michael foster, UID 511) con su archivo skel en /home/esqueleto. Verifique que los archivos correctos existen en /home/mike.

2.- Agregue una cuenta nueva para jazzy (nombre Maria Francisca Arias, UID 575) pero no le cree el directorio home. Asígnele una contraseña y verifíquela:

```
# su - jazzy
# pwd
```

¿Cuál es su directorio actual mostrado por el ultimo comando? Porque no es /home/jazzy? Salga de esta cuenta y intente ingresar como jazzy al sistema. Usted no podra, ya que el directorio home de jazzy no existe.

Ingrese como root al sistema y manualmente cree el directorio home/jazzy utilizando una copia desde /etc/skel.

Ingrese como jazzy y verifique que puede guardar un archivo en /home/jazzy/tarea.

3.- Cambie el UID de juan de 355. Y ingrese el siguiente comando:

```
#ls -la /home/juan
```

¿Cómo se ve todo, bien? ¿Qué se le olvido?

RESTRICCIONES

El interprete interactivo de Linux es mejor conocido como el SHELL. Se llama el shell porque nos lo podemos imaginar como una concha protectora, alrededor de las funciones del sistema que se involucran en las llamadas de los comandos y así protegen al usuario de tener que enfrentarse con mucho detalle. El shell

es la forma en que el usuario puede accesar las muchas utilidades que linux ofrece. El shell lee las instrucciones del usuario y ejecuta los programas.

Una característica importante de GNU/Linux es que el shell no es parte del kernel. El shell es solo un programa más de muchos. No es ni siquiera un programa privilegiado. Por esto es que se puede reemplazar el shell default de Linux con otro shell cualquiera. De hecho las distros recientes de GNU/Linux proveen una elección libre de cual shell deseas utilizar. Los shells son diferentes en su modo de operación, pero básicamente, la funcionabilidad es equivalente.

Bourne-Again Shell (bash)	Mejora del Bourne Shell original. Implementa características de csh y ksh.
Tcsh (tcsh)	Mejora del Shell C original
Public Domain Korn Shell (pdksh)	Implementación del Dominio Publico del shell popular Korn Shell.

Restringir el Acceso de ROOT

En un ambiente de red local, darle la contraseña de root a un usuario ordinario a su Workstation, abre a impredecibles situaciones inseguras a la red completa. Para esto el administrador debe crear la cuenta pseudo-root. Una manera de darle privilegios de root a un usuario normal, es crear cuentas especiales con la cual a través de un script usuarios pueden ejecutar comandos de root. Por ejemplo se puede crear un cuenta especial de shutdown (apagar), la cual apagara el sistema, si alguien ingresa a esta cuenta automáticamente se inicia el shutdown. El método de SUDO es mejor y mas seguro que este método.

Variables de Entorno y Archivos Relevantes

Pero, ¿Cómo definir variables de entorno?. Dentro de nuestro sistema hay infinidad de estas variables cuya finalidad es configurar de alguna manera el entorno de nuestra shell, como pueden ser marcar rutas de acceso para algunos programas. Podemos asignarles valores de la siguiente forma:

aperpinan@codigolibre:~# export http_proxy='http://192.168.2.254:3128'

Al referirse al contenido de una variable en Unix es necesario anteponer el símbolo $ al nombre de la variable. Para conocer el contenido de una variable en concreto se usará el comando 'echo':

aperpinan@codigolibre:~# echo $ http_proxy
http://192.168.2.254:3128

Para ver el contenido de las variables de entorno de nuestro sistema Linux en un momento determinado podremos usar el comando set. Obtendremos un listado de todas las variables y sus valores. Cuidado porque la lista es larga y fácilmente ocupará más de una pantalla así que les recomendamos usar: set | more para controlar el flujo de datos.

Variables del Entorno

Las variables del entorno almacenan valores que describen las propiedades del entorno de trabajo. Un usuario puede definir sus propias variables o modificar las ya existentes.

Para asignarle valor a una variable en el shell se emplea el operador de asignación tradicional entre el nombre de la variable y el valor asignado (no deben haber espacios intermedios). Ejemplo:

$ MENSAJE = "Hola Santo Domingo"

Para acceder al valor de una variable en el shell se emplea el carácter $ seguido por el nombre de la variable. Para imprimir en la terminal el valor de una variable se puede utilizar el comando echo. Ejemplo:

$ echo $MENSAJE
Hola Santo Domingo

Dentro de un shell se pueden ejecutar otros shells que serían hijos del primero (subshells) heredando todo o parte del entorno de trabajo del padre. Para que una variable mantenga su valor en los shells hijos es necesario indicarlo explícitamente mediante el comando export. Ejemplo:

$ export MENSAJE

Tanto la asignación del valor como el exportar una variable se pueden hacer a la vez:

$ export MENSAJE = "Hola Santo Domingo"

Para ver las variables del entorno definidas se puede emplear el comando set. Este además se relaciona con las opciones que es otra forma de regir el comportamiento del shell. Las opciones se activan (on) o desactivan (off). Estas se utilizan para indicar propiedades del shell muy específicas por lo que no nos vamos a detener en ellas. Si se desea conocer más al respecto se puede hacer $ help set | less.

Para eliminar el valor de una variable se emplea el comando unset. Ejemplo:

$ unset MENSAJE

Algunas variables del entorno en bash son:

PATH: guarda la secuencia de caminos donde el shell busca los programas que se intenten ejecutar en la línea de comandos cuando no se utilizan los caminos absolutos. Estos caminos se separan por el carácter ``:"

Ejemplo:

$ echo $PATH
/bin:/usr/bin:/usr/X11R6/bin:/usr/local/bin

Para añadir un nuevo camino puede hacerse:

$ export PATH = $PATH: /bin

USER: contiene el login del usuario actual.

PAGER: almacena el lector utilizado por defecto por algunos programas. Por ejemplo, el comando man utiliza esta variable para determinar que paginador empleará, aunque primero chequea otra variable llamada

MANPAGER y si esta no tiene valor entonces acude a PAGER, que de no tener valor tampoco, asumirá al less como paginador por defecto. También asociada a man existe la variable MANPATH donde se especifican los caminos de los manuales que despliega man y LANG para indicar el idioma.

Ejemplo:

$ export PAGER=more
$ man bash

HOME: guarda el directorio base del usuario actual.

Ejemplo:

$ echo $HOME
/home/usuario

EDITOR: contiene el editor por defecto del usuario actual. De no tener valor asociado se utiliza vi. En entornos gráficos se pueden indicar editores visuales aunque para ello se prefiere emplear la variable

VISUAL. Más adelante veremos un ejemplo del uso de EDITOR.

PS1: almacena la estructura del prompt principal del usuario. Permite una serie de macros.

Ejemplos:
\d : contiene la fecha del sistema.
\h : contiene el nombre de la máquina.
\T : contiene la hora del sistema.
\u : contiene el login del usuario.
\w : contiene el nombre completo del directorio de trabajo actual.
\W : contiene la base del nombre del directorio actual (Ej: para /home/usuario/doc base es doc).
\$: si el ID del usuario es 0 (root) contiene el valor # y sino, $.
\# : contiene el número del comando actual desde la creación del shell.

El prompt principal por defecto tiene la forma: "[\u@\h \W]\$ "

Ejemplo:
```
$ export PS1="[\T,\u\#]\$"
[14:12:15,usuario 315]$
```

PS2: guarda el prompt secundario. Este es el que se presenta cuando el shell no puede interpretar lo que se ha escrito hasta el momento. Normalmente el shell interpreta lo tecleado cuando se pulsa retorno. En caso de que no se haya completado una estructura interpretable bash muestra el prompt secundario, que por defecto es "> ". Para forzar a que bash no interprete algo después del retorno se escribe el carácter ``\" antes del retorno.

Ejemplos:
```
$ ls \
> ..
$ echo Esta es una línea largaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa\
> aaaaaaaaaaaa
# for i in "find /tmp -type f -size +1024k"
> do
> echo ————————————————————————
> stat $i
> rm -i $i
> done
```

En este ejemplo se muestra como se emplean las estructuras de control de bash, particularmente para un lazo tipo for. Como ya se ha mencionado existen otras estructuras de las cuales se podrá encontrar información en el manual de bash. Esta secuencia de código permite buscar recursivamente en el directorio /tmp todos los Archivos regulares cuyo tamaño exceda 1M y para cada uno de ellos se ejecutan los comandos stat y rm en forma interactiva. También se imprimen líneas divisorias.

HISTFILE: almacena el nombre absoluto del Archivo que contiene el historial de comandos para cada usuario. También existen HISTFILESIZE, HISTCMD e HISTSIZE. Esta última indica el tamaño del historial de comandos, que por defecto es de 1000 comandos.

Ejemplo:
```
$ echo $HISTFILE
/home/usuario/.bash_history
```

PWD: contiene el directorio de trabajo actual. Esta variable la pueden emplear algunos programas para saber desde donde se invocaron.

SECONDS: almacena la cantidad de segundos transcurridos desde la invocación del shell actual.

Archivos .bashrc, .bash_profile y .bash_logout

Cuando se invoca el shell, ella lee dos archivos para definir el perfil del usuario. Los archivos /etc/profile y /etc/bashrc, son los archivos globales de todos los usuarios en el que en su home no se define uno de estos en /home/usuario/.bashrc o .bash_profile. Estos archivos nos los encontramos en cada cuenta o directorio de usuario de cualquier sistema Linux. Son los archivos que nos permiten personalizar nuestra sesión en el sistema. El archivo .bash_profile se ejecuta al inicio de cada sesión y el archivo .bash_logout se ejecuta al final de la misma. Por tanto, podemos editarlo a nuestro gusto según queramos que se realicen unas tareas al iniciar o al finalizar nuestra sesión. El archivo .bashrc es el archivo que nuestra shell usará cuando iniciemos en sesión y donde podemos crear nuestros alias o exportar nuestras variables de entorno, entre otras cosas. Así pues, su finalidad es parecida a la del .bash_profile.

Como ejemplo, el contenido de este archivo en esta máquina como usuario gnome2 es:

```
.....
    eval `dircolors`
    alias ls='ls —color=auto '
    alias l='ls -la'
    alias cp='cp -i'
    alias rm='rm -i'
    alias mv='mv -i'
    export http_proxy='http://192.168.2.254:3128'
......
```

Algo común es llamar al archivo .bashrc desde el .bash_profile de la siguiente forma:

```
if [ -f ~/.bashrc ]; then
    source ~/.bashrc
fi
```

Además, en este archivo también se declaran variables propias del usuario, como las vistas en el apartado anterior referentes al http_proxy o cualquier otro programa.

Uso del comando "alias"

Usar el comando "alias" es una forma de definir sinónimos a los comandos del sistema o incluso redefinir los existentes con alguna forma más complicada. Si queremos definir un nuevo comando que borre archivos, lo podemos hacer de la siguiente forma:

```
aperpinan@codigolibre:~# alias borrar='rm'
```

Donde rm es el comando de Unix para borrar archivos.

Para eliminar alguna definición de alias, se deberá de utilizar el comando "unalias", especificando el alias que deseamos eliminar:

```
# unalias borrar
```

Concepto de la Variable PATH

La variable PATH guarda las direcciones o rutas de directorios donde buscar un programa ejecutable. Por ello, no es necesario situarse en una determinada ruta donde se encuentra el comando de dicho programa, sino que dentro de todo nuestro sistema podremos ejecutarlo sin ningún tipo de confusión. El archivo en donde el sistema guarda todos los valores asignados a la variable PATH es /etc/profile.

Recordar que los archivos binarios que son compilados e instalados se asignarán directamente al PATH cuando se instalen, pero por el contrario, hay programas que no aparecen en el PATH, su ejecución deberá realizarse de la siguiente forma:

```
# ./nombre_ programa
```

Mensaje del Día MOTD

El archivo /etc/motd contiene un mensaje configurable por el root. /etc/motd no es más que la pantalla de bienvenida que aparece al entrar en un servidor. Puedes configurarla a tu medida editando motd.txt con el vi o cualquier editor de texto, si quieres puedes borrar todo lo que pone dentro y si quieres puedes poner las características del servidor, o el nombre del administrador y su dirección, en fin, lo que se te ocurra.

Existen otros dos programas de mensaje similar (issue y issue.net) a motd. Issue es el que despliega los mensajes encima del login. El default es de costumbre algo referente a Linux mas la Distros que usas, versión del kernel y el nombre de la maquina que usas. Este mensaje se cambia idéntico al motd; utilizando un editor. Issue.net es también similar solo que es el mensaje que vez al entrar desde la red, a través de telnet.

Cuentas Guest/Invitados

Cuentas guest son un peligro para la seguridad. Trate de nunca crearlas o utilizarlas. Estas cuentas son muestras de indisciplina de los administradores para no crear las cuentas y condiciones o privilegios correctos. Esta cuenta de guest no debe ser confundida con las de anónimos de ftp o las de los servidores web.

Directorios Compartidos de Grupos

El group ID es una identificación de alto nivel la cual es compartida con otros usuarios del sistema, que permite (o prohíbe) a un cierto número de usuarios leer y escribir sobre los mismos archivos. Asignar el ID de grupos a directorios a compartir es muy útil cuando un grupo almacena y trabaja en los archivos de un directorio. El bit de SGID enciende el bit del grupo (4 o 010) sobre el sistema de permisos. Esto coloca una "s" en el lugar del bit indicando que el SGID esta encendido.

```
# chmod 4770 Directorio
# ls -l Directorio
drwsrwx--- 5 cristhian admin 170 14 Feb 22:01 dir
```

Ejercicio 6-5: Ejemplo de Variables de Ambiente

Solución se proveen en el apéndice A.

1. ¿Cuál de estos archivos es mantenidos por el administrador?
   /etc/profile
   /etc/bashrc
   $HOME/.bashrc
   $HOME/.bash_profile

Ejercicio 6-6: Variables de Ambiente del Usuario

Solución se proveen en el Apéndice A.

1.- Agregue un nuevo usuario llamado apagar con el comando useradd, cual llama el programa /sbin/shutdown como su shell de login. Recuerde que shutdown debe ser ejecutado desde el directorio raíz (/), y debe ser ejecutado por root (UID=0). Recuerde que debes poder apagar el sistema con simplemente ingresar con el usuario apagar.

** Necesitara utilizar la opción –o del comando useradd para especificar el UID de "0".**

2.- Agregue un nuevo usuario llamado fecha con el comando useradd, cual llama el programa /bin/date como su shell de login. Asigne un password en blanco y pruébelo con su.

3.- Ingrese al sistema como miguel e investigue el valor de su mascara de default. ¿Dónde se establece este valor?

Ejecute los siguientes comandos:

```
$ mkdir prueba
$ cd prueba
$ umask 0
$ touch um000
$ umask 022
$ touch um022
$ umask 077
$ touch um077
$ umask 770
$ touch um770
$ ls -l
```

Analice los diferentes permisos de archivos. ¿Cuales permisos le gustaría utilizar de default para crear archivos?

Asigne un umask correctamente para que en su próximo login este sea su umask default. Haga este cambio y salga del sistema e ingrese de nuevo para verificar sus cambios.

¿Qué hay de extraño en los permisos del archivo um770?

Ejercicio 6-7: Ambiente Restringido del Usuario

Solución se proveen en el apéndice A.

1.- Modifique los archivos de ambiente de miguel, para que el no pueda cambiar o borrar su perfil de usuario.

2.- Pruebe sus cambios ingresando como miguel y verifique que el no puede editar o borrar su .bash_profile.

LOGGING/INGRESAR AL SISTEMA GNU/LINUX

El programa getty es tradicionalmente el programa que inicia el proceso de ingreso de Unix. Fija la velocidad del terminal (baud rate) y proporciona el prompt del Login. Utiliza el programa mingetty, el cual es un mini programa getty para utilizar en consolas virtuales. Después de haber leído el nombre, entonces ejecuta el programa de login, cual lee el usuario y password y los verifica antes de proporcionar un shell. También existe un programa uugetty, el cual trabaja de Unix-a-Unix Copy (UUCP).

Discutiremos lo siguiente en la siguiente seccion:

- **Utilizando mingetty**
- **Características default del Login**
- **Trabajar con terminales**
- **Corregir Problemas de Puertos**
- **Base de Datos Terminfo**
- **Network Informacion Service (NIS)**
- **Lightweight Directory Access Protocol (LDAP)**
- **Módulos Autentificable de Redes (PAM)**

Utilizar Mingetty

El programa mingetty es utilizado para ajustar las características del terminal, para permitir que el programa Login/Ingreso imprima el prompt y luego recoja el nombre y el password del usuario. En GNU/Linux, mingetty es ejecutado por init, init lee la configuración desde el archivo /etc/inittab. Al hacer cambios al

archivo /etc/inittab, es necesario utilizar el comando init q para informarle al proceso init que vuelva a leer el archivo inittab. Linux suple al usuario con el uso de consolas virtuales en un sistema singular. Al presionar ALT+F [1-6], puedes cambiar entre las diferentes consolas. Es convenio no utilizar la consola tty7 ya que es la predeterminada del Sistema X Window.

```
# Run gettys in standard runlevels
1:12345:respawn:/sbin/console tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6
```

Defaults del Login

Las características predeterminadas o los defaults se definen en /etc/login.defs. Este archivo de configuración del paquete shadow password. Solo el root debe tener acceso a este archivo ya que en el se guarda los defaults de login/ingreso al sistema y además las políticas de vencimiento de las cuentas. Algunas opciones como el numero máximo de reintentos para las fallas de login, el nombre del archivo de registro (log) para los login fallidos, ubicación de los archivos de correo del usuario, los registros (logs) del su (esto no funciona con distros de redhat mas modernas), y la configuración de los timeouts de login.

Trabajar en los Terminales

Una vez ya ha ingresado en el sistema, los usuarios pueden cambiar la configuración actual de la línea serial utilizando el comando stty. Existe un gran número de opciones aplicables a este comando, muchas de las cuales tienen utilidades especiales. El principiante administrador de sistema solo tiene que conocer unas cuantas de estas opciones. Información bien amplia es ofrecida en las páginas del man.

Para verificar las preferencias del terminal ejecute el comando stty sin ninguna opción. El shell de bash contiene muchas opciones disponibles por default. Para conocer las opciones de otros shell, documéntese en las páginas man de la respectiva shell.

```
# stty -a
speed 9660; line = 1;
intr.=*c; quit=^\; erase=^h; kill=^u; eof=^d; susp=^z; …
```

Para saber en que terminal te encuentras solo tienes que ejecutar el comando tty y te responderá con el nombre del shell. Veamos un ejemplo:

```
# tty
/dev/console
```

Implementaciones modernas utilizan sistemas pts/x (o pseudo terminales) para cada conexión de red.

Corregir Problemas de Puertos

Corregir errores y problemas de terminales (troubleshooting) en Linux no es nada nuevo. Antes de todo debes revisar lo siguiente:

- Revise la conexión física, cables, etc.
- Asegurese que el terminal se reinicie a la emulación correcta.
- Revise la configuración de la línea serial del terminal, cosas como el baud rate, etc.

Sólo después de asegurarse de que todo esta bien físicamente analizaras las configuraciones de los puertos del monitor, configuraciones del stty y el valor de la variable TERM.

A menos que existan terminales separadas conectadas a la computadora en GNU/Linux, esto no debe ser un posible problema. Los mismos conceptos previamente mencionados aplicarían cuando falla una de las consolas virtuales; usted puede simplemente presionar ALT+F [1-6] hacia otra consola y desde ahí corregir el problema. Terminales pueden ser reiniciadas escribiendo "sane" o "reset". Si un terminar se inhibe al punto que no puede aceptar entrada desde el teclado tendrás que ir a otro terminar y corregir el problema desde esta terminar.

```
# stty sane </dev/tty01
# stty </dev/tty01
speed 9600; line = 1;
intr=^?; quit=^\; erase=#; Hill = @; eof=^D; susp=^Z;
…
# stty intr "^C" erase "^H" kill "^U"
```

En el peor de los casos, a la terminal se le puede enviar la señal KILL desde la otra consola ejecutando el comando "kill -KILL < el numero PID del mingetty>".

La Base De Datos TermInfo

UNiX empezó con terminales tontas y el tipo de terminal en realidad no importaba y cuando las terminales cursos-redireccionable se popularizaron, UNiX tuvo que desarrollar soporte para ellos, muchos sistemas propietarios (DEC y IBM) pudieron constringir a usuarios a cierto tipo de terminales. Como un sistema abierto, unix no pudo obligar este tipo de constreñimiento y, por esto, adoptaron una actitud diferente. Linux continúa con esta tradición.

La base de datos TermInfo contiene definiciones detalladas de las características de muchas terminales. La variable TERM especifica cual conjunto de característica a usar con la configuración apropiada, programas como el vi pueden trabajar correctamente con cualquier terminal. Muchos no proveen el fuente del texto de la base de datos TermInfo; esos que si lo proveen están colocados en /usr/src/terminfo.

En los sistemas linux un archivo ASCII /etc/termcap puede remplazar la base de datos terminfo. En la mayoria de los casos ambos, el archivo termcap y la base de datos pueden estar presentes.

Network Information Service (NIS)

NIS es una herramienta que permite a grupos de computadora entrar a una red y compartir archivos. El mejor ejemplo de los archivos a compartir es el /etc/passwd. Utilizando NIS en solo una computadora, el NIS server, o master, solo uno de ellos almacena el archivo /etc/passwd y las otras computadoras lo aceptan directamente como si fuese un archivo local.

Las ventajas de utilizar NIS en una red amplia deben ser obvias. Contraseñas pueden ser sincronizadas a través de una red de computadora. Imagínese una red con 100 computadoras. Si un administrador del sistema quisiese agregar un usuario al sistema sin utilizar NIS, tuviese que agregarlo a cada computadora por separado. Lo mismo aplica cuando un usuario quisiese cambiar su password. Con NIS el administrador simplemente agrega el usuario al servidor NIS master.

Lightweight Directory Access Protocol (LDAP)

LDAP es un protocolo para compartir información a través de una red ampliada (Ej.: una red WAN corporativa). Usa servidores con base de datos distribuidos, el cual replica y sincroniza. El directorio puede contener cualquier tipo de información pero su mejor uso es para mantener información acerca de usuario y recursos de redes.

Las principales distribuciones de GNU/Linux proveen servidores LDAP para proveer con la información del login de los usuarios. Poner en funcionamiento un LDAP es un poco complicado y se reserva para empresas muy grades. El servidor LDAP más común en linux es OpenLDAP.

Pluggable Authentication Modules (PAM)

Tradicionalmente, UNiX siempre ha utilizado contraseñas almacenadas localmente en /etc/passwd. Nuevos métodos de confirmación de información de password han sido introducidos requiriendo que estos programas sean reescritos para manejar estos nuevos métodos. Llego un momento que se saturo el numero de métodos de autentificación y dificulto rescribir todos los programas cada vez que un nuevo método fue desarrollado. Sun MicroSystem desarrollo el sistema PAM para permitir a programas de autentificación, como el login, usar métodos de autentificación programables.

Utilizando sistema PAM cada programa de autentificación tiene una configuración en /etc/pam o en el directorio /etc/pam.d. La configuración le dice cual método de identificación usar con cada programa de autentificación. Por ejemplo puede tener que el programa login busque en /etc/passwd y que el programa SU use NIS o LDAP. PAM también puede utilizarse al cambiar o asignar contraseñas.

PAM provee una manera fácil de personalizar los métodos de autentificación que usted usa en su sistema. Le da al administrador flexibilidad y permite a los programas sacar de su código los métodos de autentificación. Todo lo que se requiere es que el programa este escrito para utilizar PAM para todos sus métodos de autentificación.

EJERCICIO 6-8: Trabajando con Tipos de TERM

No se proveen soluciones para este ejercicio.

1.- ¿Qué despliega el prompt del usuario después que ejecute la siguiente secuencia de comandos del shell bash?

```
$ tput rev
$ tput rmso
$ REV=$(tput rev)
$ NRM=$(tput rmso)
$ PS1=$REV$(uname -n)$NRM: ‘echo $PWD:’
```

EJERCICIO 6-9: Login y Terminales.

Para este ejercicio no se dan soluciones.

1.- Cambia los default del login global para que el valor de la umask sea 077.

2.- Ingrese con alex y escriba la configuración de las teclas especiales descrita por stty?

```
erase?
intr.?
kill?
eof?
```

Cambia su tecla de “intr” Que sea CTRL+A y que tu tecla de kill sea CTRL+B. Digite un comando a medias y presione CTRL+C. Describa lo que paso.

Presione CTRL+A y Describa lo que paso ahora.

Digite un comando a medias y presión CTRL+B

3.- Ingrese como Alex y determine cual es su carácter de quit es probablemente CTRL+\, borrar el siguiente

comando que debe tomar un buen rato y presiona las teclas de quit. $ ls -lR /
Busque el archivo core que se genero.

Entre los siguientes comandos:

$ rm core*
$ ulimet -c 0

Trate la primera parte de la pregunta y fíjese la diferencia en el tamaño del core.

4.- ¿Cuál es su tipo de terminar en uso?

Entre los siguientes comandos:

$ OTERM=$TERM
$ TERM=wyse50
$ vi
:q! #salga del vi puesto que todo salio mal.
$ TERM=$OTERM

¿Porque no pudo vi dibujar en la pantalla?

5.- Digite los siguientes comandos (NOTE que los primeros 3 usan paréntesis y los ultimos llaves):

$ BOLD=$(tput bold)
$ BLINK=$(tput blink)
$ NORM=$(tput sgr0)
$ PS1='${BOLD}Yes ${BLINK}Master${NORM}? '
$ tput clear

6.- Asegúrese a colocar todo como estaba antes de comenzar este ejercicio.

RESUMEN

En este capítulo usted fue introducido los siguientes conceptos:

- Crear y administración de cuentas de usuarios
- Asegurar y regular acceso a cuentas de usuarios
- Personalización del ambiente del usuario
- Configuración (Setup) de terminar

PREGUNTAS POST - EXAMEN

Las respuestas a estas preguntas están al final de este libro en el Apéndice A

1.- ¿Qué es un terminar virtual? ¿Cómo puedes distinguir en cual terminar estas?

2.- ¿Describa como corregir un terminar que no responde?

3.- ¿Describa el rol de mingetty en los Login de los usuarios?

4.- ¿Cómo puede un usuario redefinir las opciones del terminar?

5.- ¿Describa como y porque un administrador de sistema debe configurar un directorio con el bit SGID?

CAPITULO 7

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

PROGRAMAR TAREAS Y ADMINISTRAR BACKUPS

OBJETIVOS

Al completar este Capítulo, usted podrá:

- Automatizar tareas administrativas del sistema programando los trabajos para ser ejecutados en el futuro.
- Administrar una estrategia de Backup efectiva.
- Compare y contraste las siguientes aplicaciones de backup:
  •Amanda •KBackup •UNiBACK •Taper •Arkeia
- Conocer el sistema cron y estar en capacidad de configurar la programación de trabajos utilizando cron.
- Identificar la implementación del daemon cron en backup de archivo del sistema.
- Conocer utilitarios utilizadas para archivar y la diferencia entre los programas

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1.- Nombre los cinco campos del archivo crontab

2.- ¿Cómo edita usted una entrada de crontab?

3.- ¿Cómo puede usted programar para que un programa se ejecute una sola vez ?

4.- ¿Qué programa debe estar ejecutandose para que una entrada de crontab se ejecute ?

5.- ¿Qué debe usted hacer después que ha efectuado el backup de un archivo ?

6.- Cuáles son las ventajas y desventajas de los backups sobre la red ?

7.- ¿Cuáles son las diferentes herramientas disponibles de GNU/Linux para hacer backups de sus sistemas ?

8.- Liste 2 diferentes maneras de desempaquetar archivos tar con gzip. Ej. archivo.tar.gz

INTRODUCCION

La idea detrás del uso de computadoras, es para automatizar tareas que toman mucho tiempo o son tediosas para los humanos. Administrar un sistema puede convertirse un poco tedioso asi que buscamos maneras de automatizar tareas cotidianas. Una forma es a traves de los scripts del shell, los cuales combinan una serie de comandos y lo convierten en un solo comando. Otra manera de automatizar tareas de administración del sistema es programar la computadora que ejecute comandos automáticamente a tiempos especificos. Los mecanismos que proveen linux para lograr esto son los sistemas cron y at. Los sistemas cron y at pueden ser utilizados para ejecutar un simple comando o una serie de comando dentro de un shell script.

El backup es una de las tareas mas importantes del administrador del sistema siempre es una buena idea estar preparado para lo peor y mantener una copia de toda su data en caso de catástrofe. Dependiendo de la importancia de su data y el cambio dinámico de la información tendrás que desarrollar estrategias para su backup. Usted puede o hacer backup completo de sus sistemas o backup incrementales el solo cambia los archivos que han sido modificado desde el último backup. Existen diferentes herramientas en Linux para hacer su backup puede utilizarlas para hacer su backup en gran variedad de medios.

Este capítulo le introducirá a los fundamentos de programar tareas con cron y at hacer backup y utilizar herramientas populares de GNU/Linux para lograr su objetivos.

El cron

Cron es un sistema para programar procesos que van a ser ejecutados con regularidad. Uso común de cron son para empezar un backup, para rotar archivos de registro y inicializar script del sistema. El sistema cron consiste de un daemon y un archivo de configuraciones por usuarios. Cada archivo de configuración es denominado una tabla cron o mejor dicho crontab. Una entrada en un archivo crontab se llama un evento o trabajo. Existen por lo menos dos versiones utilizadas en los sistemas GNU/Linux. Mientras que los conceptos básicos son lo mismo, algunos particulares varían. Siempre revise su documentación y que entienda como utilizar cualquier herramienta de administración de sistema. Los siguientes tópicos se discuten en esta sección:

- El daemon de cron
- Los archivos crontab

Vamos a identificar cada campo:

```
*       *       *       *       * /commando/a/ejecutar
-       -       -       -       -
|       |       |       |       |
|       |       |       |       +----- día de la semana (0 - 6) (dom = 0)
|       |       |       +------- mes (1 - 12)
|       |       +--------- día del mes (1 - 31)
|       +------------ hora (0 - 23)
+-------------- minuto (0 - 59)
```

El Daemon cron

Como la mayoría de los servicios del sistema la funcionabilidad de cron es proveída por un sistema daemon cron o crond dependiendo de la distribución que usted utiliza. El Cron lee el archivo de configuración para determinar cuales comandos ejecutar cuando. Cada un minuto cron lee todos los archivos crontab para ver que comando debe ejecutar. Si el encuentra una entrada que considera con la hora actual; el ejecuta el comando correspondiente, ejecutándolo con el UID del dueño del archivo crontab. Por esto solo podrás ejecutar a través del cron lo mismo que tus permisos te permiten. Cron puede ser configurado para permitir o

denegar a usuarios específicos la habilidad para programar eventos. Los archivos para especificar quien puede utilizar cron son /etc/cron.allow y /etc/cron.deny. Si el archivo cron.allow existe, solo los usuarios listados en el pueden utilizar cron si el archivo no existe solo usuarios que no aparecen en el archivo cron.deny. Un archivo cron.deny vacío significa que todo los usuarios pueden utilizar cron.

Si los archivos no existen el programa puede permitir o a todos los usuarios o solo a root dependiente de la configuración y la distribución en uso. El cron mantiene un directorio en spool para almacenar los archivos crontab. Casi siempre este directorio se encuentra en /var/spool/cron y contiene un archivo crontab para cada usuario que tiene un trabajo programado. También existe un archivo crontab del sistema en /etc/crontab el crontab usa una sintaxis un poco diferente del de los usuarios, con un campo agregado que especifica bajo el cual el trabajo del usuario ha de ser ejecutado. Normalmente la salida del trabajo ejecutado por cron es enviado por correo al usuario. Esto puede ser cambiando y redireccionar la salida del archivo o especificar un usuario diferente para enviar el correo.

Los Archivos crontab

El archivo crontab le dice al daemon cron que programas usted quiere empezar y cuando empezarlo. Cada usuario tiene un archivo crontab y también existe un archivo crontab en el directorio /etc. El programa utilizado para administrar los archivos crontab también se llama crontab. Existen dos tipos de entrada: definición de variables de ambientes y eventos. Una definición de variable de ambiente se dice a cron colocar una variable para que un programa que el ejecuta debido a un evento del cron. Para asignar esta variable, entre el nombre de la variable con un símbolo de "=" y el valor de la variable. Hay una variable de ambiente especial denominada MAILTO que especifica donde se debe de enviar la salida. Si lo deja en blanco (MAILTO =””) toda la salida es ignorada. Por defecto la salida es enviada correo al dueño del archivo crontab.

La mejor parte de las entradas en el archivo crontab son eventos, un evento tiene dos partes: el tiempo que el evento se ejecuta y que hacer cuando llegue el momento de ejecutarlo. Cinco campos en la entrada del evento representan tiempos. En este orden, ellas son: minuto, hora, día del mes, mes, día de la semana. Los campos son separados por espacios o tab. Recuerde que las horas se representas en horas militar de 24 horas. En el campo del día de la semana puede utilizar o cero o siete para domingo y los otros días en orden numérico.

Un evento se ejecuta cuando el tiempo del campo coincide con el tiempo actual. El cron revisa una vez por minuto para encontrar entradas que deben ser iniciada, para un evento, ser iniciado cada campo de tiempo debe coincidir con el tiempo actual excepto el día de la semana y el día del mes; solo uno de estos dos campos necesita coincidir.

Además de escribir un valor simple en cada campo puedes utilizar un comodín, un rango de valores, una lista de valores o un incremento, el "*" es el carácter del comodín que coincide con cualquier valor. Un rango se delimita por un símbolo "-"indicando que cualquier valor entre estos valores es correcto. Una lista es separada por comas. Un incremento es indicado utilizando el carácter "*" seguido por (/), seguida por un numero (*/2). El campo de incrementar cuando el valor es un múltiple del número después de la barra.

Después de los 5 campos se encuentra el comando a ser ejecutado el comando toma el resto de la línea después de los campos del tiempo y puede incluir espacio. La mayoría de las versiones de cron requiere un campo adicional del nombre del usuario antes del campo del comando para el crontab del sistema, /etc/crontab. Este es el usuario bajo el cual el comando se ejecutara, casi siempre este usuario será root o deny o nobody.

Los siguiente es un ejemplo del archivo crontab.

```
MAILTO = root
0       *       *       *       *       echo “Ejecuta cada hora.”
0       1.2     *       *       *       echo “Ejecuta a la 1 AM y 2 AM.”
13      2       1       *       *       echo “Ejecuta a la 2:13AM el día 1st de cada mes.”
9       17      *       *       1-5     echo “Ejecuta a la 5:09PM todos los días de semana.”
0       0       1       1       *       echo “Feliz año Nuevo!”
0       6       */2     *       *       echo “Ejecuta a la 6AM los días de fecha par.”
```

Tenga especial cuidado de que todos los campos de tiempo estén correctos. Ya que no recibirás ningún mensaje de error si cometes cualquier falta al digitar sus entradas, pero su trabajo no ejecutara como espera.

Comandos del crontab

Usted no debe editar sus archivos crontab directamente en el directorio /var/spool/cron. Debe utilizar siempre el comando:

```
# crontab -e
```

Este comando utilizara el editor por defecto definido en las variables de ambiente VISUAL o EDITOR. Si ningunas están definidas utilizara el vi cual es el editor por defecto de las mayorías de distros de Linux.

Puedes desplegar sus crontab actual con el siguiente comando:

```
$ crontab -l
```

Para remover todo el contenido de su crontab, utilice el comando:

```
$ crontab -r
```

Puedes especificar a crontab que lea las tablas desde un archivo:

```
$ crontab /home/usuario/mi_crontab
```

Cuídese de no confundir el mecanismo que cron utiliza para determinar cual archivo crontab debe utilizar, así que utilice la opción –u para especificar de cual usuario es el archivo crontab que se va a afectar:

```
$ crontab -u ivellise /home/ivellise/su_crontab
```

Trabajos Preconfigurados del cron

La mayoría de las distribuciones vienen preconfiguradas con varios trabajos cron activo. Éstos se proporcionan para ejecutar mayormente trabajos de la limpieza del sistema. Generalmente, ejecutan llamadas a scripts de nombre /etc/cron.weekly, /etc/cron.daily, /etc/cron.hourly y /etc/cron.mensual. El administrador del sistema entonces puede modificar estos scripts para hacer cualquier trabajo que el desee hacer regularmente. A excepción de los scripts programado a correr cada hora, estos scripts se fijan para ejecutarse tarde en la noche en horas de menos volumen de trabajo.

Cada vez más, las distribuciones cambian hacia un sistema donde estos archivos shell scripts se substituyen por directorios de los shell scripts. El directorio tiene el mismo nombre que el shell script, pero en vez de funcionar un solo script todo los shell scripts dentro del directorio son ejecutados. Esto es similar al sistema usado para iniciar y detener servicios al cambiar de runlevels.

Se utilizan los trabajos típicos del cron para poner al día bases de datos del sistema. Un trabajo común del cron llama el comando logrotate. Este hace un backup de los archivos del registro (logs) y los recorta de modo que no sean demasiado grandes. Otro comando que normalmente se ejecuta a diario es el comando updatedb. Este comando crea una lista de todos los archivos en el sistema para poder utilizar el comando locate de localizar y encontrar archivos sin tener que buscar el sistema de archivos entero. Otros trabajos comunes del cron son los de funcionar chequeos de seguridad en el sistema.

At y batch

El comando at se utiliza para ejecutar un comando o un script en un tiempo especificado. Diferente a cron, estos trabajos funcionan solamente una vez, no sobre una base que se repite. El comando batch es como el comando at, pero él específica que los comandos se ejecuten cuando el promedio de la carga este por debajo de cierto nivel. El límite medio de la carga por defecto es 0.8, pero un cualquier límite puede ser especificado por el superuser.

Los trabajos sometidos vía y los comandos at y batch son manejados por el daemon atd. Este programa trabaja como el daemon del cron pero mantiene una cola de los trabajos y de los tiempos que se suponen ser ejecutados. El daemon atd no necesita comprobar para saber si hay trabajos nuevos; solo esperas hasta que es hora de comenzar un trabajo ya en cola o en el comando at le dice que agregue algo a la cola. El sistema at tiene la estructura de archivo permitir y denegar (/etc/at.deny (/etc/at.allow) y el control de quien puede utilizarlo. Los campos funcionan exactamente como los del sistema cron. El directorio spool para el atd está generalmente en el directorio/var/spool/at. La única opción probablemente del atd que usted necesitará utilizar es la opción -1, que especifica el promedio de la carga máxima en el cual los trabajos se permiten ejecutar.

Para programar un trabajo en at, utilice el comando at y de el tiempo que usted desea comenzar el trabajo. Cuando usted presione ENTER, usted entrará un modo que le permitirá digitar los comandos que usted desea poner a funcionar en el trabajo. De entrada a los comandos y presione CONTROL+D para terminar la entrada:

```
[root@www /root]# at 8 AM
Warning: commands will be executed using (in order) a) $SHELL b) login shell c) /bin/sh
at> echo “Buenos Días!”
at> <EOT>
job 5 at 2003-07-27 08:00
```

Alternativamente, usted puede especificar un archivo para funcionar usando - la opción de f:

```
[root@www /root]# at 9 AM -f /etc/init.d/atalk
warning: commands will be executed using (in order) a) $SHELL b) login shell c) /bin/sh
job 7 at 2003-07-26 09:00
```

Observe el mensaje de alerta de utilizar /bin/sh. Esto puede ser significativo si usted usa el tcsh en vez del bash porque los dos tienen un sintaxis un poco diferente. Cualquier salida del comando se envía al usuario por e-mail.

El comando at es bien flexible en aceptar formatos de fecha y hora. Usted puede especificar el tiempo en formato de hh:mm o simplemente la hora. También funciona la palabra clave now para especificar el tiempo actual. Usted puede especificar la fecha en formato de mm/dd/yy o de mm/dd o escribir el nombre del mes. Usted puede también especificar un tiempo seguido por un signo de más (+} con un incremento en un cierto número de minutos, de horas, de días, o de semanas. Los nombres de los días y las palabras claves today y tomorrow están también disponibles para especificar una fecha. La cosa principal a recordar es que el tiempo es requerido y debe estar en formato de 24 horas a menos que usted especifique a.m. o p.m.

Los ejemplos siguientes son para ilustrar algunos de estos formatos:

```
$ at now + 3 hour
$ at 6:10pm + 1 days
$ at 5:30 tomorrow
$ at 2am
$ at 8pm 12/23/2003
$ at 9:35 Jul 26
```

```
$ at 6 Saturday
```

Usted puede echar un vistazo a la cola en una de las dos siguientes maneras:

```
[root@www /root]# at -l
1    2003-07-26 20:00 a root
2    2003-07-26 20:00 a root
3    2003-07-27 08:00 a root
5    2003-07-27 08:00 a root
6    2003-07-26 09:00 a root
7    2003-07-26 09:00 a root

[root@www /root]# atq
1    2003-07-26 20:00 a root
2    2003-07-26 20:00 a root
3    2003-07-27 08:00 a root
5    2003-07-27 08:00 a root
6    2003-07-26 09:00 a root
7    2003-07-26 09:00 a root
```

Ambos hacen la misma cosa, solo cambia la sintaxis del comando. Si usted es superusuario, usted verá los trabajos enumerados de todos los usuarios. La mayoría de las versiones de at no dejarán que usuarios normales listen los trabajos de otros usuarios.

Para ver exactamente qué comandos en particular ejecutara un trabajo, uso at -c le dará una lista de todos los comandos que serán ejecutados. Puede demostrar muchas definiciones de variable de entorno y de cambio de directorio porque at ejecuta los trabajos en el mismo ambiente que fueron creados.

Se utiliza el comando atrm para quitar/remover trabajos en cola. Solo liste el número del trabajo en la cola. Usted puede también utilizar la opción -d (o en algunas versiones -r) del comando:

```
$ atrm 13
$ at -d
```

Ejercicio 7-1: Usos de at y cron

Solución se proveen en el Apéndice A.

1. Usted debe fijar las variables de entorno VISUAL o EDITOR antes de usar crontab ya que puede ser que le habrá un editor de textos que usted no sepa utilizar. Los editores de textos recomendados son pico, jed, emacs y vi. (Note que no todas las distribuciones tendrán todos éstos instalado)

2. Edite su tabla del cron usando el comando crontab

3. Cree un trabajo del cron para enviarse un saludo por E-mail en su cumpleaños. Crear otro trabajo del cron de enviarse un E-mail 10 minutos de ahora. Cerciorarse de que usted reciba ese mensaje.

4. Utilice at para enviarse un mensaje por E-mail 5 minutos de ahora. Cerciorarse de que usted lo reciba.

   Pruebe con varios formatos de tiempo.

5. Liste la cola de at y quite algunas entradas

BACKUP Y RESTAURAR

Los Backups son importantes porque proporcionan medios de almacenaje alternativos para los datos importantes. Backups no serían necesarias si los sistemas informáticos fuesen 100% confiable. Pero en realidad, todos los sistemas informáticos pueden fallar tarde o temprano. La mayoría de los fallos del sistema dan lugar a la pérdida o a la corrupción de datos. Una política de backups regular permite al administrador del sistema recuperar los datos del sistema al estado que era cuando se efectúo el último backup. Los Backups también se utilizan para transferir datos entre las máquinas no interconectadas por redes.

Los usuarios corren el riesgo de accidentalmente por equivocaciones, suprimir archivos importantes de datos pensando que ya no eran necesarios. Una buena política de backups permitirá que el administrador del sistema recupere los datos almacenados en el backup en estas situaciones, que de hecho se darán. Los Backups son inútiles si los datos en ellos no pueden ser recuperados. Para que sirve un backup si posteriormente no se puede restaurar.

Los siguientes tópicos serán discutido en esta sección:

- Porque hacer BACKUP
- A que hacerle BACKUP
- Donde almacenar el BACKUP
- Cuando hacer el BACKUP
- Restaurar

Cuando Hacer Backup

Hacer copia de respaldo de su sistema es la única manera de poder repararlo si sufre un daño severo, o incluso si borra por accidente algunos archivos importantes del sistema, o si alguien irrumpe en su sistema y borra intencionalmente algunos archivos. También debería hacer copia de respaldo de los datos que usa a diario (audio comprimido, imágenes, documentos de oficina, correo-e, libreta de direcciones, etc.) para estar seguro.

Debería hacer sus copias de respaldo usando un soporte apropiado y mantenerlas en un lugar seguro. Tal lugar debería estar fuera del lugar en el que Usted trabaja usualmente, si es posible. Incluso puede tener dos copias de respaldo, una en el lugar de trabajo y otra fuera del mismo. En general, debería asegurarse que podrá recuperar dichas copias de respaldo si desea que todo esto realmente sirva para algo.

Preparar su Sistema

Probablemente ya tiene todo lo que necesita instalado en su sistema. También debería tener un disquete de arranque a mano (hizo uno, ¿cierto?). En realidad, puede hacer copias de respaldo usando sólo a tar y una herramienta de compresión tal como gzip o bzip2. Más adelante les damos un ejemplo de backup utilizando tar.

Como alternativa, puede usar programas de copia de respaldo especializados, tales como KBackup, Taper, Time Navigator, Arkeia, etc.

Donde Almacenar el BACKUP

Almacene las cintas del backup fuera de la oficina, siempre y cuando las políticas de su trabajo se lo permitan. Si un fuego destruye sus equipos el seguro los reemplazara, no será igual con su data. Recuerde que los sistemas de BackUps de Linux que aquí describimos no tienen ningún sistema de seguridad asociados con ellos, así es que cuide bien su data y quien tiene acceso a estas cintas de backup.

¿Qué Debe Incluir en su BACKUP?

Bueno, esta puede ser otra pregunta difícil que cada administrador del sistemas se pregunta cuando llega la hora de hacer la copia de respaldo. La respuesta depende de cosas tales como: ¿sólo está respaldando sus datos personales, sus archivos de configuración, o todo su sistema? ¿Cuánto tiempo y/o espacio va a tomar? ¿Restaurará su copia de respaldo en la misma máquina/versión de sistema operativo, o en una diferente?

Tendrás que hacerle backups al sistema completo sistemáticamente, especialmente antes de cualquier actualización del sistema y antes de mantenimiento preventivo, o si se piensa mover el equipo. Al hacerle backups a sistemas de base de datos tenga cuidado, sistemas de base de datos involucran varios archivos que deben ser consistentes. Detenga el servicio de la base de datos antes de hacerle el backup. Las mayorías de los sistemas de base de datos tienen módulos de backup, para así no tener que detener el servicio. Base de datos que utilizan almacenamiento RAW para escribir al disco proveen sus propios métodos de backup.

Debido a que esto es una guía de solución de problemas, trataremos de concentrarnos en hacer una copia de respaldo tal que nos permita restaurar rápidamente nuestro sistema al estado en el cual estaba antes que ocurra esa cosa terrible que lo inutilizó. Por supuesto, necesitará hacer copia de respaldo de sus datos personales si no desea perderlos, pero... esa es otra historia.

Como regla general, necesitará hacer copia de respaldo de los directorios siguientes:
/etc **/home** **/root** **/var**

Si hace una copia de respaldo completa de estos directorios, habrá guardado no sólo sus configuraciones, sino también sus datos (en caso que se esté preguntando dónde están sus datos, están en el directorio /home/su_nombre_de_usuario/). Por favor, tenga presente que esto puede tomar un tiempo largo en completarse, pero es la apuesta más segura.

Un esquema más sofisticado sería hacer copia de respaldo sólo de los archivos de configuración que han cambiado, dejando de lado los que no han cambiado. Esto llevaría más tiempo de "planificación", pero llevará a tiempos de copia de respaldo más cortos (y también a tiempos de restauración más cortos) y a copias de respaldo que son "más fáciles" de portar de una máquina/versión de sistema operativo a otra.

A continuación, se le presentará una lista de los archivos a los cuales debería prestarles la mayor atención. Note que estas listas no son exhaustivas en absoluto, en especial si ha hecho un montón de cambios en su sistema.

En el directorio /etc:
/etc/lilo.conf

Contiene la configuración del cargador de arranque LILO. Si, en vez de LILO, Usted usa grub, los archivos a incluir en la copia de respaldo son los que están en el directorio /boot/grub.

/etc/fstab
Contiene la configuración de las tablas de partición de los discos y los puntos de montaje asociados.

/etc/modules.conf
Contiene los módulos a cargar y sus parámetros de acuerdo al hardware de su sistema. Puede no ser de utilidad si se restaura en una máquina muy diferente, pero de todas formas puede proporcionar algunas pistas.

/etc/isapnp.conf
Contiene las configuraciones de ISAPnP si es que lo usa para configurar el hardware ISA plug'n'play.

Nota: Con el núcleo 2.4.x puede no necesitar más este archivo, ya que el hardware plug'n'play se configura usando el sistema de archivos DevFS.

/etc/X11/XF86Config-4 y/o /etc/X11/XF86Config

Contiene las configuraciones de X. X es el "corazón" gráfico de GNU/Linux y todos los entornos de escritorio y administradores de ventanas del mismo.

/etc/cups

Contiene las configuraciones de CUPS. CUPS es el sistema de impresión predeterminado de Mandrake Linux. Si no utiliza CUPS y usa el sistema de impresión lpr, entonces tiene hacer copia de respaldo de

/etc/printcap

/etc/bashrc

Configura al shell bash, para todo el sistema.

/etc/profile

Configura el entorno del sistema y algunos programas que se ejecutan al iniciar el sistema.

/etc/crontab

Configura los jobs de cron a ejecutar periódicamente, por ejemplo, para las tareas de mantenimiento del sistema.

/etc/rc.d/*

Configura los distintos niveles de ejecución del sistema. Usualmente, no necesitará hacer copia de respaldo de estos, excepto si agregó algún nivel de ejecución personalizado o cambio uno de los predeterminados.

/etc/inittab

Configura el nivel de ejecución predeterminado con el cual arranca su sistema.

/etc/ssh

Contiene las configuraciones de ssh. Si utiliza el acceso remoto seguro, es sumamente importante incluir este archivo.

Si tiene un servidor web, un servidor FTP, u otros servidores, también haga una copia de respaldo de sus respectivos archivos de configuración. Note que no podemos listarlos a todos aquí ya que los mismos dependen del servidor que utilice.

En el directorio /root y en el directorio personal de cada usuario /home/nombre_de_usuario, los directorios siguientes:

~/.gnome/* y ~/.gnome2/*

Configuraciones para el entorno de escritorio GNOME.

~/.kde/*

Configuraciones para el entorno de escritorio KDE.

~/.mozilla/*

Configuraciones para la familia de programas Mozilla. Los marcadores de Navigator, los filtros de correo de Messenger, etc. Note que este directorio también contiene todos sus mensajes de correo-e y de sus grupos de noticias. Definitivamente no quiere perder estos, ¿cierto?

~/Mail/*

Si usa kmail este directorio contiene todos sus mensajes de correo-e. Definitivamente no quiere perder estos, ¿cierto?

~/.ssh/*
Contiene las configuraciones personalizadas para el uso de ssh. Si utiliza a ssh, es obligatorio realizar copia de respaldo de este.

Tampoco quisiera perder de vista a los archivos siguientes:
~/.bash_profile y ~/.bashrc
Contiene las variables de entorno, los alias y más configuraciones para el shell bash.

~/.cshrc
Contiene las variables de entorno, los alias y más configuraciones para el shell CSH.

~/.tcshrc
Contiene las variables de entorno, los alias y más configuraciones para el shell tcsh.

Por favor, note que no mencionamos todos y cada uno de los posibles archivos de configuración debido a que hubiésemos necesitado un libro entero sobre el tema. Por ejemplo, si Usted no utiliza Mozilla no necesita hacer copia de respaldo de los archivos y directorios relacionados con Mozilla, si Usted no utiliza ssh no necesita hacer copia de respaldo de las cosas relacionadas con ssh y así sucesivamente.

Resumiendo, haga copia de respaldo de todos los archivos de configuración de los programas que usa y de todos los archivos de configuración que Usted ha modificado. También haga copia de seguridad de todos los archivos de datos personales (y de los usuarios de su sistema) No se va a arrepentir.

Como Hacer Copias de Respaldo/Backup

Cuándo realizar un backup es una pregunta difícil de contestar. La respuesta generalmente es “tan a menudo como sea necesario”. Una manera mejor es la importancia de los datos. ¿Puede usted permitirse perder un año de datos? ¿Talvez un mes de datos? ¿Y una semana o quizás un día? La realidad del asunto en la mayor parte de las empresas es que no pueden perder ni un solo día de datos, así es que la repuesta es DIARIO.

Utilice a cron para programar la ejecución de los scripts de BACKUP que se ejecuten tarde de la noche, cuando el volumen de trabajo se encuentra bien reducido. Utilice backups incrementales. Nunca utilice la misma cinta de backup todos los días, si fracasa el intento de backup perderá la de ayer y el de hoy.

Hay muchas políticas para las agendas de copia de seguridad. Aquí le presentaremos algunas. Por favor, tenga presente que estas no son obligatorias, ni son las mejores, ni son las únicas. Simplemente son guías que querría seguir al planificar su propia agenda de copia de seguridad.

Las distintas estrategias de copia de respaldo que existen dependen del soporte que Usted utilice, de cuan seguido cambian sus datos y de cuan críticos son sus datos para Usted o para su organización. Por ejemplo, una estrategia dice que debería hacer una copia de respaldo completa cada fin de semana y una incremental (sólo las cosas que cambiaron) cada día; luego hacer una copia de respaldo completa cada mes y guardar esa última en al menos dos lugares. Esta estrategia puede resultar ser útil para una empresa pero no para una computadora personal. Para sus copias de respaldo personales puede pensar en algo como esto: hacer una copia de respaldo semanal en su disco rígido y cada mes transferir esas copias a CD-R/DVD+RW o cinta.

Ejemplo de Copia de Respaldo Usando tar

Seguidamente, le presentaremos un pequeño script de copia de respaldo que usa a tar para hacer una copia de respaldo completa de su directorio personal.

Ejemplo:

Necesitará permiso de lectura sobre los archivos y permiso de lectura y ejecución sobre los directorios que va a incluir en la copia de respaldo, de no ser así la operación de copia de respaldo fallará.

```
#!/bin/bash
# Crear una copia de respaldo comprimida de su directorio
# personal en el archivo backup.tar.gz o backup.tar.bz2
# dependiendo del esquema de compresión usado.

BACKUP_DIRS=$HOME

# Quitar comentario de la línea siguiente si desea comprimir con GZIP
#tar cvzf backup.tar.gz $BACKUP_DIRS

# Aquí comprimimos con BZIP2...
tar cvjf backup.tar.bz2 $BACKUP_DIRS
```

Como puede ver este es un script de copia de respaldo muy simple que sólo hace una copia de respaldo de su directorio personal y pone el resultado en el mismo directorio. Vamos a mejorarlo un poquito.

```
#!/bin/bash
# Crear una copia de respaldo comprimida de todos los directorios
# especificados y poner el archivo resultante en un directorio
# de nuestra preferencia.

BACKUP_DIRS="$HOME /etc /etc/rc.d"
BACKUP_FILENAME=`date '+%b%d%Y'`
BACKUP_DEST_DIR="/backups"

# Quitar comentario de la línea siguiente para usar GZIP, dejarlo
# para usar BZIP2

# tar cvzf $BACKUP_DEST_DIR/$BACKUP_FILENAME.tar.gz $BACKUP_DIRS

# Aquí usamos BZIP2 para comprimir...
# Comentar la línea siguiente para usar GZIP, quitar comentario
# para usar BZIP2
tar cvjf $BACKUP_DEST_DIR/$BACKUP_FILENAME.tar.bz2 $BACKUP_DIRS
```

Como puede ver en este último ejemplo, hemos añadido algunos directorios más a nuestra copia de respaldo y hemos usado un esquema de nombres para agregar la fecha de la copia de seguridad al nombre del archivo resultante.

Por supuesto, puede mover el archivo tar.bz2 o tar.gz resultante a cualquier soporte que desee. Incluso puede hacer copia de respaldo directamente sobre el soporte que desea si lo monta y cambia la variable BACKUP_DEST_DIR del script adecuadamente. Siéntase libre de mejorar este script y hacerlo tan flexible como desee.

Para restaurar las copias de respaldo hechas de esta forma, por favor véase mas adelante en la sección Restaurar.

Restaurar

La restauración de la copia de seguridad depende del programa, soporte y agenda que Usted utilizó para hacerlo. Aquí no cubriremos todos los casos de restauración, sino que sólo mencionaremos que Usted se tiene que asegurar de restaurar los archivos y/o directorios en los mismos lugares donde se encontraban cuando hizo la copia de respaldo para poder recuperar sus configuraciones y sus archivos de datos.

Ejemplo de restaurar un Backup usando tar

Ahora, veremos un pequeño script para restaurar la copia de respaldo que hicimos con tar usando el script que se presentó antes en "Ejemplo de copia de respaldo usando tar"

Ejemplo:

Necesita permisos de escritura sobre los archivos y directorios que va a restaurar, de no tenerlos, la operación de restauración fallará.

```
#!/bin/bash

# Extraer una copia de respaldo comprimida de todos los
# Directorios especificados poniendo los archivos en sus lugares
# Originales.

BACKUP_SOURCE_DIR="/backup"
RESTORE_FILENAME=$1

# Quite el comentario de la línea siguiente si está comprimido
# con GZIP

#tar xvzf $BACKUP_SOURCE_DIR/$RESTORE_FILENAME

# Restaurar una copia de respaldo comprimida con BZIP2...
tar xvjf $BACKUP_SOURCE_DIR/$RESTORE_FILENAME
```

Como puede ver, este script es bastante simple. Todo lo que tenemos que hacer es pasarle el nombre del archivo de la copia de respaldo que deseamos restaurar como parámetro (sólo el nombre del archivo, no la ruta completa), y el script restaura los archivos de la copia de respaldo en sus ubicaciones originales.

¿Dónde Hacer Copia de Respaldo?

La otra gran pregunta a responder. Esto depende de cuanta información desea incluir en la copia de respaldo, cuan rápido desea hacer sus copias de respaldo, cuan fácil es el acceso al soporte de la copia de respaldo y una larga lista de etcéteras.

En general, necesitará soportes que tengan a lo sumo tanta capacidad como la cantidad de información que desea incluir y sean suficientemente rápidos como para que el proceso completo termine en un tiempo razonable.

Soportes Para Copias de Respaldo

Acontinuación, le proporcionaremos una descripción breve de las opciones de soportes de copia de respaldo disponibles. Estas varían en capacidad, confiabilidad y velocidad. No se dan en orden particular alguno, simplemente como vienen a la mente. Por favor, note que el software que Usted utiliza para hacer copia de respaldo puede o no soportarlos a todos.

> *Nota: Esta lista no pretende ser un análisis exhaustivo de los diferentes soportes de almacenamiento disponibles en el mercado. De hecho, algunas de las cosas que se describen a continuación pueden cambiar en el futuro. Cosas tales como el tiempo de vida esperado del soporte fueron tomadas de los sitios web de los fabricantes y/o de la experiencia personal y de la comunidad. También, puede haber muchos puntos de vista personales sobre temas tales como el precio o la velocidad, por ejemplo.*

Disquete

Su capacidad llega hasta 1.44 MB. Se pueden llevar de un lado a otro con facilidad pero, para las necesidades actuales, pueden no tener espacio suficiente. La mejor forma de llevar archivos pequeños. Lento. Barato. Disquetera estándar en virtualmente cada computadora que existe. Lectura/Escritura. El tiempo de vida esperado es de 4 o 5 años.

Aviso: Por favor, tenga presente que los disquetes no son muy confiables.

Disquete LS120

Su capacidad es 120 MB. Dimensiones físicas idénticas al disquete pero con casi diez veces más la capacidad. No muy baratos. Necesita una disquetera especial pero esta disquetera también puede leer/escribir disquetes estándar. Puede ser un buen reemplazo para los disquetes pero su velocidad queda atrás con respecto a la de las unidades ZIP. Lectura/Escritura. El tiempo de vida esperado es más o menos el mismo que para los disquetes ZIP

Disquete ZIP

Su capacidad llega hasta 750 MB. Aunque no son tan delgados como los disquetes, también se pueden llevar de un lado a otro con facilidad y son más adecuados para las necesidades de hoy día. Buen balance de características, aunque pueden ser un poco caros. Lectura/Escritura. El tiempo de vida esperado es de diez años para las unidades de 100 MB, Talvez sea mayor para las de 250/750 MB.

CD-R

Su capacidad llega hasta 700 MB estos días, aunque el estándar es 650 MB. Soporte muy barato y confiable. Hoy día muchos comentan que su capacidad no es suficiente, pero en realidad 650 MB son suficientes para la mayoría de las personas. Su característica más fuerte es que casi todas las computadoras de la Tierra tienen una unidad de CD-ROM, por lo tanto se pueden leer casi en cualquier lugar. Se pueden escribir sólo una vez. Se pueden leer tantas veces como lo desee (bueno, en realidad, tantas como pueda...). El tiempo de vida esperado es 20 años, Talvez más si se almacenan en un lugar seguro y no se leen muy seguido.

CD-RW

Las mismas consideraciones que con los CD-R, pero se pueden formatear y volver a escribir hasta 1000 veces. En general, es un soporte barato y confiable. El tiempo de vida esperado es 15 años, Talvez más si se almacenan en un lugar seguro y no se leen muy seguido.

DVD Grabable/Regrabable

Esta es una de las adiciones más nuevas a los soportes de almacenamiento disponibles. La capacidad del mismo es de 4,7 GB para discos DVD grabables de una sola cara. Las grabadoras son un poco caras, pero eso se compensa en gran medida por el hecho de poder almacenar 4,7 GB en un disco único. El tiempo de vida esperado es 15 años, Talvez más si se almacenan en un lugar seguro y no se leen muy seguido.

Cinta

Su capacidad va desde 120 MB (¿alguien tiene unidades de cinta tan antiguas?) hasta varios gigabytes. Es un soporte caro y no muy confiable (después de todo son cintas magnéticas). No obstante, su capacidad los hace ideal para hacer copias de respaldo de servidores y cosas por el estilo; si desea hacer copia de respaldo de todo su disco rígido en una sola pieza de soporte, la cinta puede ser la única opción. Su desventaja más importante es que el acceso a la cinta es secuencial y esto tiene un gran impacto en el desempeño, pero

las unidades de cinta SCSI son lo suficientemente rápidas para las necesidades de hoy día y tienen muchos gigabytes de lugar para almacenar sus archivos. Lectura/Escritura. El tiempo de vida esperado es 30 años para las tecnologías de cinta nuevas.

Disco Rígido

Los precios de los discos han caído de forma tal que también pueden ser considerados seriamente como soporte para copias de respaldo. Son relativamente baratos, tienen amplio espacio (hasta 120 GB al momento de escribir este manual), son muy confiables y el soporte más rápido de todos los que se presenta en esta lista. Si tiene un sistema portátil esto puede no ser una opción, pero en sus sistemas de escritorio agregar una segunda unidad de disco sólo para propósitos de copia de respaldo puede ser una buena opción. En realidad, puede no necesitar agregar una unidad de disco rígido nueva y hacer copias de respaldo en el único disco rígido que tiene; pero esto puede no ser una idea muy buena ya que no lo protegerá si se le arruina el disco rígido.

Otros Soportes Removibles

Existen otros soportes removibles (El ORB de Castlewood y el JAZ de IOMEGA me vienen a la mente) que tienen un buen balance de precio/características y son adecuados para hacer copias de respaldo. Algunos incluso fueron publicitados como “reemplazos de su disco rígido” (JAZ por ejemplo) pero cuando se usan como discos rígidos pueden no durar mucho debido a restricciones de diseño (No son unidades de disco rígido). Sin embargo, en estos temas Su Experiencia Puede Ser Distinta, simplemente asegúrese de elegir con sabiduría (use el sentido común) de acuerdo a sus necesidades y... ¡buena suerte!

Directorios Remotos

Bueno, estrictamente hablando, estos pueden no ser considerados como “soporte”, pero mencionaremos algo sobre ellos porque son una buena opción para copia de respaldo siempre y cuando tenga espacio y ancho de banda suficientes.

Si su ISP le proporciona algo de espacio, usted puede usar ese espacio para colocar sus archivos junto con sus páginas web. En la web puede encontrar muchas ofertas de servicios de almacenamiento remoto en línea. Si tiene una red con dos o más máquinas pueden hacer copias de respaldo en alguna máquina “remota” en la red (por supuesto, una máquina distinta a la que Usted está intentando hacer copia de respaldo...)

En realidad, el hecho de hacer copias de respaldo “remotas” puede ser un problema de seguridad, entonces no mantenga en una copia de respaldo remota sus archivos ultra secretos ni sus archivos más importantes. Recuerde que, en caso de una falla severa del sistema, Talvez ni pueda conectarse a ese sitio remoto para recuperar los archivos...

Por favor, tenga presente que también puede combinar soportes de copia de respaldo de acuerdo a su estrategia para realizar las copias de respaldo, por ejemplo: cintas y CD-R, disco rígido y cintas, disco rígido y CD-R, etc.

Terminología de Backup de GNU/Linux

Los siguientes términos le serán útiles en su investigación de Copias de Seguridad en Linux:

Archive	Un backup completo y removido del sistema en línea (borrado una vez el backup ha sido completado
Image Backup	Una copia completa (byte por byte) de un disco o partición
Full backup	Completo backup del directorio
Differential Backup	Backup de los archivos que han cambiado desde el último backup completo
Incremental Backup	Backup de los archivos que han cambiado desde el último backup completo o incremental.

Dump levels (0-9) Backup de los archivos que han cambiado desde el ultimo backup en el mismo o mas bajo nivel del dump (Donde nivel 0 es igual a backup completo).

Utilidades Manuales de Backup

Las existen numerosas utilidades para hacer sus backups. Dependiendo del tipo de backup que usted desea hacer, usted puede utilizar diversas utilidades. La mayoría de las herramientas descritas en esta sección están diseñadas para ser puestas en un guión/script conjuntamente con otras para lograr nuestros objetivos. Son los pasos necesarios a dar para que las tareas de backup que usted se crearía. Las utilidades más completas e integradas serán cubiertas en la sección siguiente.

• Utilidades Orientada a Archivos

cpio Copy to I/O
afio Versión actualizada de cpio
tar Archiva a Cinta y Restaura

• Utilidades Nivel de Dispositivos

dd Direct Device access (Image backups)

Algunas distribuciones proporcionan sus propias utilidades propietarias y muchos sitios de Internet llevan la información sobre los usos de tercera persona para GNU/Linux.

Los siguientes tópicos se discuten en esta sección:

• Tape Archive y Restaurarlo (tar)
• Copiar a I/O (cpio)
• afio
• Direct-Device Access
• Usar DD para identificar el tipo de archivo
• Nombres de Dispositivo De Cinta de Linux
• Manipulación de cintas con mt
• Trabajo con disquetes del DOS con mtools
• Ponerlos junto con compress
• Backups de red con rsh
• Haciendo un CD-ROM de recuperación

Tape Archive y Restaurarlo (tar)

Utilizar el alquitrán para las reservas rápidas y simples. No es bastante flexible para las estrategias de reserva sofisticadas. Los archivos se escriben al archivo en formato del alquitrán y toda la información siguiente también se almacena sobre los archivos:

• Ruta del directorio e información del inode
• Usuario, grupo, e información del permiso
• Tiempos de la creación y de la modificación

Digámosle que deseas hacerle un backup al directorio /home/trabajos. Los comandos para él utilizarían la opción -c (crear):

```
# cd /home/trabajos
# tar cvf /dev/ftape
```

Esto hiciese un backup del directorio/home/trabajos en /dev/ftape del dispositivo de la cinta.

Para recuperar estos archivos, usted puede utilizar el comando tar con la opción -x (extraer):

```
# cd /tmp
# tar xvf /dev/ftape
```

Esto extraerá todo los archivos desde /home/trabajos a /tmp.

Copiar a I/O (cpio)

Puedes utilizar el cpio para hacer backups del sistema. Soporta estrategias flexibles y sofisticadas de backups. Los ficheros se escriben al archivo en formato cpio y toda la siguiente información se almacena acerca de los archivos:

- Ruta del directorio e información del inode
- Usuario, grupo, e información del permiso
- Tiempos de la creación y de la modificación

Opciones de cpio

Las opciones operacionales son:

- i extrae Archivos
- o Crea el archivo
- p Crea el archivo en el mismo sistema de archivos

Las opciones generales son:

- v modo de mensajes activo
- B Utiliza bloques grandes
- Cn Utiliza bloques de tamaño n bytes
- c Utiliza archivos headers tipo ASCII (Siempre utilice esta opción)

Especificar los dispositivos de I/O con las opciones siguientes:

- O archivo Para tener un mejor manejo de medios multivolúmenes al archivar
- I archivo Para tener un mejor manejo de medios multivolúmenes al restaurar

Opciones de entrada (restaurar):

- t Listas más bien la tabla de contenido que los archivos del restore de las
- d Crea directorios si está necesitado
- u Incondicional restaura archivos
- m Conserva tiempos de la modificación del archivo

afio

El comando afio es otra utilidad de archivar, usando archivos del formato cpio pero creando los archivos cabezales (headers) del tipo ASCII para la mejora de compatibilidad {similar al cpio -c). Algunas características del afio incluyen:

- El manejo interactivo para los archivos
- Compatibilidad con cpio
- Más opciones disponibles que en cpio
- Archivos comprimidos de afio son más seguros que archivos comprimidos con tar o cpio

Opciones de afio

Las opciones operacionales son:

- i Extraer Archivos

- o Crea el archivo
- t Lista Contenido de la tabla del archivo
- r Verifica el archivo contra sistema de ficheros

Opciones Generales de afio son:

-v Modo de Mensajes
-Z Utiliza gzip para comprimir/descomprimir ficheros de un archivo

Opciones para tape y floppy son:

-s volumesize	El tamaño del volumen (defecto es 5,120) pero también reconoce el tamaño seguido por k, m, y g para el KB, el MB y el GB
-c n	Coloca n número de bloques entre cada operación de I/O
-F	Dispositivo es un Floppy; -s se requiere para esta opción

Opciones de Input (Restaurar) son:

-n No sobre escribe archivos mas recientes
-k Salta data corrompida al principio de un archivo

Preparar un CD-ROM de Recuperación

Hay una manera de estar preparado en caso de un "desastre total", y es hacer una copia de respaldo completa de su sistema. Los programas como mkCDrec pueden ser muy útiles para que Usted pueda recuperarse y estar operativo en cuestión de minutos.

La utilidad mkCDrec le permite hacer volúmenes de CD-ROM múltiples, clonado de discos (copiar todo el contenido de un disco o partición a otro con características similares – al menos el mismo tamaño), y muchas cosas más.

Para poder restaurar un sistema con mkCDrec simplemente tiene que arrancar con el primer CD-ROM de los múltiples CD-ROM del volumen y seguir las instrucciones en pantalla.

Ejercicio 7-2: Uso de cpio

Solución se proveen en el Apéndice A. Explique que sucede al ejecutar los siguientes comandos.

1.- # find . -print \ afio -ovb > /dev/ftape

2.- # afio -tb < /dev/ftape

3.- # cd /tmp;

afio -ivab < /dev/ftape

4.- # cd /;

find etc home var -print | afio -ovb > /dev/ftape

5.- # cd /tmp;

afio -ivaby etc/init.d /dev/ftape

Acceso Directo a Dispositivo

El dd se utiliza para tener acceso a un dispositivo directamente. Se utiliza a menudo para copiar pedazos grandes de datos, por ejemplo en backups. Escribe el contenido en formato raw de sus archivos al dispositivo. Algunos parámetros útiles para el comando de la DD son:

of=file	Nombrado en vez del stdout del archivo del al de Escribe
if=file	Lee desde el archivo nombrado en vez de stdin
bs = size	Especifica tamaño del bloque (también los ibs y los obs)
count=n	Copia solo n numero de record
conv-ascii	Convierte Extended Binary-Coded Decimal Interchange Code (EBCDIC) a ASCII
Conv=ebcdic	Convierte ASCII a EBCDIC
conv=ibm	Un poco diferente convierte del ASCII al EBCDIC
conv=swab	Intercambiar cada par de bytes (big endian/little endian)

El siguiente comando copiará los datos en /dev/ftape a /home/restaurar en pedazos de 4.095 bytes:

```
# dd if=/dev/ftape of=/home/restaurar bs=4095 conv=ibm.swab
```

Ejercicio 7-3: Copiar un Disco

Solución se proveen en el Apéndice A.

1.- ¿Cómo copiarías un disquete preformateado?

Utilizar dd para Identificar el tipo de Archivo

Imaginarse que le dan un disco de backup marcado con un nombre de archivo pero ninguna información sobre el comando usado para crearlo. A usted se le pide restaurar los datos de es backup. ¿Cuál utilidad usted utiliza?

Aquí es en donde dd demuestra su gran utilidad:

```
# dd if=/dev/fd0 count=1 of=test1 ***leera solo un block desde el device y lo almacena en un archivo.
# file test1
test1: tar
```

Esto es mejor que probar cada utilidad alternadamente para ver si trabaja. El comando dd es muy versátil y vale la pena aprender a usarlo.

En este ejemplo, lo utilizamos para leer un bloque de datos de un dispositivo. Nos confiamos en el hecho de que cualquier utilidad que utilizaron para almacenar la información que se nos pidió extraer debe colocar sus archivos cabeceras (headers) al principio del medio de almacenaje, para así poder identificarse. Así es cómo cada comando sabe si los datos están en su formato y rechazarán la lectura del dispositivo si el header no lo comprende.

Pero dd, sin embargo, no lee estos archivos headers, ya que es una copiadora de imagen y leerá cualquier cosa si el medio en el que se almacena es leíble.

Nombres en Linux de Dispositivos de Cinta (Tape Device)

Los sistemas GNU/Linux usan convenciones simples. Los dispositivos de la cinta son almacenados en /dev. Los dispositivos son rft* y nrft* (sobre nombres al ftape y al nftape, respectivamente). Los dispositivos de cinta comienzan con r (rebobinado/rewind) o nr (no rebobinar/no-rewind).

Usted puede manipular cintas magnéticas con el comando mt. Se reconocen muchos parámetros útiles, permitiendo que usted enrolle y que rebobine la cinta a un punto específico así como la verificación del estado de la cinta. Un programa llamado ftape está también disponible y es más rico en característica que mt. Los nombres de dispositivo de cinta magnética utilizan el más diverso de las convenciones de nombramiento. El factor común es incluir r o nr en el nombre para indicar rebobinado o no-rebobinado de los dispositivos.

Manejar de Cintas con mt

Usted puede manipular cintas magnéticas con el comando mt. Usted especifica un dispositivo particular de cinta con –f. Los parámetros útiles que el comando reconoce incluyen:

Rewind	Rebobina al comienzo de la cinta
Offline	Rebobina y saca fuera de línea (dispositivo puede expulsar la cinta)
fsf n	Salta hacia adelante n archivos
bsf n	Salta hacia atrás n archivos
com	Salta adelante al fin de la cinta
status	Imprime la información del estado de la unidad de cinta

Por ejemplo:

```
# cd /
# mt -f /dev/ftape rewind
# echo “Sistema de Archivos; /et.c\nDate: ‘date’” | dd of=/dev/ftape
# find etc -print j | cpio -ocv >/dev/rmt/On
# mt -f /dev/ftape rewind
# mt -f /dev/ftape fsf 1
# cpio -itvc </dev/ftape
# mt -f /dev/ftape offline
```

El comando mt es útil para trabajar con las cintas cuando múltiples archivos van a ser escrito a la misma cinta.

El ejemplo anterior escribe los mismos datos a la cinta (una etiqueta y un backup con cpio) pero hace esto en una manera menos propensa a producir error. Observe que la cinta es tomada fuera de línea después del backup para no poderle sobre escribir accidentalmente.

Trabajar con Disquetes DOS con el mtools

GNU/Linux brinda soporte para disquetes formateados con DOS. Para más información sobre el asunto, usted puede revisar las páginas man para saber más sobre mtools. Algunos de los comandos de mtools son:

Mdir directorio	lista directorio al estilo DOS
mcopy archivo archivo	Copia archivos a y desde Disquete
mdel archivo…	Borra el file(s)
mformat drive	Da formato tipo FAT DOS a la unidad especificada

Los nombres de archivo del disquete se pueden especificar de la siguiente forma:

```
drive:ruta ruta              puedes utilizar / o \
$ mdir a:
CARTA      TXT          1,226    26/07/2003      19:58
OFICIO     DOC          4,463    26/07/2003      16:56
FOTO       GIF          779      26/07/2003      11:10
3 files 6468 bytes
16,848,817 bytes free
```

mcopy a :/carta.txt .

El paquete de mtools incluye casi todas las herramientas del DOS para manipular el sistema de archivos: **mattrib, mbadblocks, mcd, mcopy, mdel, mdeltree, mdir, mdu, mformat, mkmanifest, minfo, mlabel, mmd, mmount, mmove, mpartition, mrd, mren, mshowfat, mtooltest, mtype, mzip, xcopy….**

Para los otros refierase a la página man mtools y mucha información más.

Poner Todo Junto con compress

Usted puede acelerar los backups y reducir espacio de almacenaje comprimiendo. Utilice la utilidad de la comprimir de Linux para reducir tamaño del archivo. Para una compresión más eficiente, usted puede utilizar los formatos gzip y bzip2.

Para bloques más grandes, utilizar el comando de la dd:

 # find . –print | cpio -ocvB | compress | dd of=/dev/ftape bs=65536
 # dd if=/dev/ftape bs=65536 | uncompress | cpio -itvcB

A veces técnicas especiales de backups se requieren para los sistemas. Éstos están generalmente más allá del alcance del interfaz de la administración del sistema y requieren más conocimiento y capacidad del administrador del sistema.

Los tamaños de los archivo se pueden reducir dramáticamente usando la utilidad de compresión a través de una tubería con los comandos de backup. Ajustado con un bloque más grande (del dd), un backup grande se puede reducir de tamaño y convertido en uno más pequeño. El coste es, por supuesto, en términos del tiempo y de la carga de la CPU del sistema. El un volumen más pequeño de datos de backup reducirá probablemente el tiempo total también.

Ejercicio 7-4: Usar tar, gzip y compress

Solución se proveen en el Apéndice A.

Este ejercicio se ocupará del uso de tar, gzip y de la utilidad de comprimir compress para archivar y de comprimir varios archivos para crear un backup.

1.- ¿Cuál es la manera más simple de colocar el contenido del directorio /bin en un solo archivo tar y nombrado bin.tar?

2.- ¿Usando el gzip, cómo se puede comprimir un archivo para ahorrar el mayor espacio?

3.- ¿Cómo se puede comprimir el archivo usando compress?

4.- ¿Cómo se pueden combinar estos pasos de una manera más simple? ¿Si es así, cómo?

5.- ¿Cómo se puede descomprimir el archivo gzippped? ¿ Cómo se puede descomprimir el archivo compress?

6.- ¿Qué comando expandirá el archivo comprimido a un archivo sin comprimir?

7.- ¿Cómo se pueden ejecutar los comandos uncompress y untar al mismo tiempo?

Backups de Redes con rsh

Usted puede también hacer sus backups sobre la red. Usted puede utilizar el comando rsh para lograr esto. Usted tendrá que montar sistemas de archivos de la red usando el NFS o algo similar antes de comenzar. Se puede hacer backup de los datos de cualquier sistema de archivos, incluyendo drives conectados vía red. Si usted está utilizando drives mapeados en la red, debe tener cuidado que usted de no le haga backup a sistemas de archivos que no desea ni repita datos porque el sistema de archivos este mapeado en otro sistema de archivos ya incluido en su esquema de backup. Usted puede utilizar el comando find -mount para restringir la búsqueda de los archivos.

Los sistemas networked de Linux se les puede fácilmente hacer un backup usando utilidades de red estándares, tales como drives networked compartidas en la red o la ejecución de comandos remotos.

En un ambiente networked (mapeado), no todas las máquina requieren tener su propio drives, con tal que tenga acceso a una máquina en la red que si lo tenga. Los ejemplos siguientes demuestran cómo hacer un backup sobre una red a un drive en otra máquina.

```
# find . –print | cpio -ocvB | rsh miguel dd of=/dev/ftape
# rsh miguel dd if=/dev/ftape | cpio -itvcB
# rsh gnome2 'find . –print | cpio -ocvB' | dd of=/dev/ftape
# dd if=/dev/ftape | rsh gnome2 cpio -itvcB
```

El primer ejemplo muestra un backup desde la máquina local a la máquina remota rosie (la cual tiene el drive); el segundo asume que el drive está en el sistema local, y se le está haciendo el backup a la máquina remota, de nombre gnome2.

NOTA: Por razones de seguridad, usted debe utilizar el ssh en lugar de rsh.

Ejercicio 7-5: Backups y Restaurar

Solución se proveen en el Apéndice A.

1.- Abra una sesión como root y ejecute el siguiente comando:

```
# touch /home/backup
```

2.- Cambie al directorio /usr y hágale un backup al directorio dict a disquete usando tar. El dispositivo del disquete se encuentra en /dev/fd0.

¿Por qué piensa usted que se le pidió cambiar al directorio y hacer el backup a los discos de backup en dos acciones separadas? ¿Cómo puede usted verificar que el backup trabajara?

Restaure el backup en el directorio /tmp/dict.tar.

3.- Repita el backup anterior y restaurarla con el cpio (restaure en /tmp/dict.cpio).

4.- ¿Si usted no supiera qué formato de datos fue escrito al disquete, cómo usted lo descubriría?

Ejercicio 7-6: Backups Programados

Solución se proveen en el Apéndice A.

1. Repita las operaciones de backup de los pasos 2 y 3 en el ejercicio 7-5, primero tar use y entonces cpio, pero mida el tiempo de la operación usando el comando time como en los siguientes ejemplos:

```
# cd /usr
# time tar cvf del /dev/fd0 dict
# time (find dict -print | cpio -ocv > /dev/fd0 )
```

¿Por qué usted nos piensa utilizó paréntesis en el segundo ejemplo?

2. Repita los dos backups otra vez, pero esta vez, de salida a los archivos a /tmp/tar y /tmp/cpio más bien que a disquete.

   Repita otra vez, pero esta vez haga que tar y cpio escriban a la salida estándar, pase la salida por tuberías al comando compress y redireccione todo a los archivos /tmp/tar.Z y /tmp/cpio.Z. Los ejemplos siguientes demuestran lo qué se requiere:

```
# cd /dict
# tar cvf - dict | compress > /tmp/tar.Z
# find dict -print | cpio -ocv \ compress > /tmp/cpio.Z
```

   Ahora vamos a comparar los tamaños de los cuatro ficheros archivados.

3. ¿Cómo haría usted un backup de todos los archivos que se han modificado desde que /home/backup fue creado?

Ejercicio 7-7: Tecnicas de Backup

Solución se proveen en el Apéndice A.

1. Ponga el disquete que contiene un backup en la disquetera y ejecute los siguientes comandos:

```
# dd if=/dev/fd0 count=1 of=/tmp/floppy
# file /tmp/floppy
```

   ¿Qué piensan usted que acaba de hacer?

2. De formato a un disquete con un sistema de archivos DOS.

   Copie los archivos hosts a este disquete. Ejecute un ls para probar que esta hay. Ahora copie este archivo de nuevo a /tmp y llámelo DOS.hosts. Analícelo para asegurarse que es valido este archivo.

   Copy el disquete entero a un archivo y llámelo /tmp/dos.fd. Ejecute el siguiente comando para corromper el disquete del DOS:

```
# dd if=/usr/bin/date of=/dev/fd0
```

   ¿Puede usted conseguir un listado del directorio del DOS disquete?

   Ahora copie la imagen de su /tmp/dos.fd al disquete.

   ¿Puede usted ahora conseguir un listado DOS del disquete?

Utilidades de Backup Integradas

En el pasado, crear un sistema de backup significaba que un administrador necesitaría fabricar un sistema de scripts (guiones) que utilizarían un puñado de otras herramientas más pequeñas. Esto sería específico a cada sistema y construir este el sistema de backup a mano era uno de los deberes principales del administrador de sistema. Mientras que la importancia del tener un buen sistema de backup sigue siendo de vital importancia, las opciones de herramientas han crecido al punto donde no es siempre necesario fabricar el sistema de backup desde cero. Esta sección cubre un número de utilidades de backups integradas que puedan hacer backups del sistema mucho más fáciles lograr que en el pasado. Las siguientes utilidades serán dis-

cutidas en esta sección:

- Amanda
- KBackup
- UNiBACK
- Taper
- Arkeia

Amanda

El programa de Amanda es el resultado del proyecto Advanced Maryland Automatic Network Disk Archiver en la universidad de Maryland. Sirve como sistema de backup que permite que un solo servidor cree backups de cinta de una multiplicidad de anfitriones/hosts a través de una red. Este programa se utiliza más comúnmente para hacer backups de cajas UNIX, pero las versiones recientes le han dado la capacidad de soportar otros sistemas operativos usando la samba.

Características de Amanda

Desarrollado y lanzado como software libre sin ninguna ayuda de los Desarrolladores. Por esta razón, Amanda puede aparentar o dar la impresión de carecer en características como sus contrapartes comerciales, pero éste no es el caso. Amanda es la herramienta disponible de backup de redes de UNIX más completa. Las características de Amanda incluyen:

- Arquitectura basada cliente/servidor
- Capacidad de realizar backups de la red completa
- Mantiene una base de datos para localizar las cintas
- Informes generados para cada sesión de Backups
- Los datos se comprimen antes de ser enviados a la cinta
- Utiliza programas estándares de backup de UNIX
- Cintas activas no se sobre escriben
- Puede realizar reservas desatendidas con un automounter
- Uso eficiente de la cinta

Amanda trabaja con un sistema de backup incrementales. Consecuentemente, solamente algunas reservas completas se realizan cada noche. Amanda funciona de tal manera en cuanto a hace el uso más eficiente de espacio disponible de la cinta. Este proceso implica no solamente backups incrementales, pero utiliza la capacidad de Amanda de variar su horario de backup basado en la cantidad de datos a respaldar y el espacio disponible en cinta.

Disponibilidad

El paquete de Amanda, se incluye en la gran mayoría e distros de Linux (RedHat, Mandrake, SuSE, Debian). El Home Page de Amanda es http://www.amanda.org. Esa página debe dirigirle a ftp://ftp.amanda.org/pub/amanda de donde pudiese descargar el paquete en caso de que su distro no lo incluya o que desee actualizarlo a la última versión. Después de localizar y de descargar el archivo más reciente de Amanda en formato tar, deberá utilizar el programa tar de LINUX, luego untar el archivo y entonces lea el Readme e instalar los archivos. El método para compilar y configurar una instalación de Amanda en un sistema RedHat será explorado de este punto adelante. Para la información sobre la compilación y el configuración en un cierto otro sistema, usted puede ser que intente recurrir a la documentación proporcionada por el programa o mire hacia algunos de los recursos citados el final de esta sección.

Programas Requeridos

La instalación de Amanda podía requerir un número de paquetes auxiliar. Para hacer la instalación lo más simple posible, localizar e instalar cada uno de los paquetes siguientes antes de continuar:

- GNU tar 1.12 o mayor (http://www.gnu.org)
- Perl 5-004 o mayor (http://www.perl.org)
- GNU rcadline 2.2.1 o mayor (http://www.gnu.org)
- GNU awk 3.0.3 o mayor (http://www.gnu.org)
- Gnuplot 3.5 o mayor (ftp://ftp.dartmounth.edu/pub/gnuplot)

Crear Reservas/BackUPs con Amanda

Una vez que esté configurado correctamente, Amanda requiere muy poca intervención para crear los backups. Durante el proceso de la configuración, los programas principales de Amanda se deben haber configurado para funcionar bajo cron en intervalos de momentos específicos. Durante la operación normal Amanda comprobará para asegurarse de que la cinta correcta esté cargada cuando el cron ejecuta el comando amcheck. Si Amanda no encuentra ningún problema, esperará hasta la ejecución del programa. Sin embargo, si Amanda detecta un problema, le envía un E-mail al operador pidiéndole que corrija el problema antes de la programada ejecución del backup esa noche.

Agregar Discos

Para informarle a Amanda acerca de discos nuevos que se pondrán en su rotación de backups, el archivo disklist debe ser actualizado. Después de poner al día el archivo, Amanda comenzará automáticamente a incluir el disco nuevo en su rutina de backup.

KBackup

KBackup es un programa basado en el shell que se ha desarrollado para manejar backups en cintas, discos duros y sistemas de archivos. Si usa un sistema Slackware Linux, instalar el afio en su máquina para poder ejecutar KBackup. Los programas utilizados por KBackup en un sistema GNU/Linux son:

- Dialog
- gtar
- afio
- grep
- fgrep
- sed

KBackup provee a los usuarios con un programa de backup de gran alcance mientras que mantiene una interfaz simple y fácil de aprende. Algunas de las características más versátiles de KBackup son su facilidad del manejo de los menús de manipulación y configuración. KBackup utiliza tar o afio para su manejo de archivo y provee a los usuarios con una simplificación de las numerosas características de backup. KBackup provee de usuarios control agregado en cinco áreas importantes. Incluyen la configuración del archivo, programar operaciones, utilización del programa de MULTIBUF y de crear y de restaurar backups. Aquí están algunos aspectos únicos de KBackup:

- Cada archivo en una cinta contiene un header que contiene las opciones elegidas, parámetros, formato de compresión, fecha, tipo de backup, etc.
- Al usar las cintas o los dispositivos en modo bloque montables, todos los ficheros en el archivo son almacenados en un directorio al principio del archivo.
- Si existen datos sensibles, usted puede comprimir y cifrar los archivos usando cifrado público de encrip-

tación.
- Usted puede programar los backups o restaurar las operaciones que se ejecutarán en cualquier momento específico.
- Usted puede incluir y/o excluir archivos usando patrones estándares del shell.
- Hay una opción entre backups completos o incrementales.
- KBackup soporta las cintas, floppies y los medios removibles.
- Cada tarea de backup puede tener diversos archivos configuración basadas en menú.
- Usando el programa MULTIBUF, KBackup detecta automáticamente la longitud de las cintas.
- KBackup proporciona acceso a los dispositivos remotos de para backup de LAN.
- KBackup permite que las configuraciones sean nombradas para permitir formatos variados de archivo.

Programar Operaciones

Un punto fuerte en el arsenal de KBackup es la capacidad para el usuario programar backups. Después de elegir la función de programar backups, KBackup termina y realizará la reserva en el tiempo especificado. Después de elegir la opción de programar restaure, KBackup tiene acceso a la información en el archivo especificado, requiriendo el archivo estar disponible. Después de que un usuario haga selecciones, el KBackup termina y los archivos especificados serán restaurados cuando se le ha llegado su tiempo.

MULTIBUF

La puesta en práctica del programa MULTIBUF es otra ventaja del sistema de KBackup. MULTIBUF es un programa escrito especialmente para el uso dentro de KBackup. Se piensa para ser un manejador para los archivos multivolúmenes, substituyendo la necesidad del afio y de tar.

Crear un Backup

Crear un backup utilizando KBackup es un proceso bastante simple. Fije el directorio padre para el backup como el directorio que su contenido será almacenado, lo cual puede ser hecho en el menú de configuración. El usuario debe entonces seleccionar el comando backup en el menú principal. Por ejemplo, si usted necesita hacer un backup de su directorio home, fije sus opciones y guardelas bajo el nombre que desee. Si hay una necesidad repetir esta tarea, después para incorporar el menú de la configuración y para incorporar el nombre bajo el cual salvo la configuracion para restaurar ajustes previamente elegidos. Al usar backups incrementales, la fecha y la hora se utilizan de comparar los archivos mientras que crea los backups incremental y son dependientes en la configuración del usuario. La fabricación de un backup con la configuración A no afecta la fecha y la hora para los backups incrementales usando la configuración B.

Restauración de un Backup/Copia de Respaldo

Antes de que se restaure cualquier archivo, el usuario debe entender que los archivos son restaurados relativos al directorio actual. KBackup incita a usuario para su autentificación de cambios antes de que ocurra cualquier cambio. Se permite al usuario también cambiar el directorio por defecto que el restore se hace adentro. Para restaurar archivos más viejos multivolume, restauración manual usando tar o afio debe ser realizada.

UNiBACK

UNiBACK es siglas para el backup de UNIX y ahora está disponible para Linux. UNiBACK para Linux es similar a UNiBACK para UNIX. Es una solución rápida, fácil y confiable de backup de la red. Son ambos el mismo producto ahora llevado al ambiente de Linux. UNiBACK es un producto comercial que los funcionamientos en la versión RedHat 5-2 en adelante y son un producto del software de ORBiT SoftWare, que se asocio con RedHat. Existe versión Demo del programa de UNiBACK disponible para probar por 45 días. Una

copia de trabajo completa de UNiBACK para Linux para un solo servidor cuesta US$200.

Taper

Taper es un programa para archivar diseñado para el uso con cintas, aunque trabajará con cualquier archivo en cualquier dispositivo a el cual Linux pueda escribir. Taper es más de uso fácil que tar porque almacena la información sobre los archivos que son archivados al principio del proceso para la recuperación fácil. Encontrar archivos con taper es, por lo tanto, mucho más rápido que con tar, taper también almacena archivos de información sobre los archivos que son hechos en el directorio home del usuario.

Arkeia

Arkeia es una aplicación de backup que permite a los administradores LAN y WAN y los usuarios backup y restaurar fácilmente sus archivos. Arkeia puede centralizar y manejar los backups de una amplia gama de sistemas al mismo tipo de medios. Usando un control central, Arkeia puede backup y restaurar todos los sistemas distribuidos conectados a él. Utilizando una base de datos central corriente del archivo backed-up, puede fácil y eficientemente no perder de vista todos sus archivos y facilita el proceso de la restauración.

RESUMEN

En este capítulo, se introdujo lo siguiente:

- Utilidades y Conceptos de Programar Backups
- Comandos de backups tales como tar, cpio, y dd
- La carencia de las restricciones incorporadas de control de acceso en los backups de Linux
- Usar backups para mantener integridad del sistema
- Usar almacenaje fuera del equipo cuando es posible
- Utilidades para ahorrar espacio de Almacenaje en los backups conjuntamente con el comando de compress.
- BackUps sobre la red usando el comando del rsh
- Medios de backups comunes tales como cintas y CDs (CD-R, CD-RW)
- GNU/Linux y los comandos especiales que trabajan con los disquetes DOS

PREGUNTAS POST - EXAMEN

Las respuestas a estas preguntas están al final de este libro en el Apéndice A

1.- ¿Cuál del siguiente es entradas válidas del archivo del crontab?

A.- * * * * echo "Hola" > /tmp/hola.out
B.- # Mes Dia Hora Minuto Comando
C.- 0 2/4 * * * echo "Hola"
D.- * * 0 * echo "Hola"
E.- 11111 echo "Hola"
F.- 13 13 13 13 * echo "Hola"
G.- */4 * * * * echo "Hola"

2.- ¿Cuál de los siguientes es una válida representaciones de time/date?

A.- 1
B.- next friday
C.- last tuesday
D.- 07/26/1961
E.- 7/26/2003 9am
F.- 1 tomorrow
G.- now
H.- 4:11pm+ 2 days
I.- 10:00 am July 31
J.- noon + 2 hours
K.- midnight +1 week

3.- Muestre una entrada de crontab que ejecute uptime a la 1:00 de la mañana cualquier martes y cualquier día del mes que termine en un cero.

4.- ¿Cómo programarías para ejecutar un programa a la 1:23 a.m. cada lunes y las 2:34 a.m. cada martes?

5.- Ejecute un backup completo el domingo. Ejecute backups incrementales en las noches de lunes y martes.

Usted necesita restaurar un archivo la mañana del miércoles. ¿Cuántas cintas usted necesita utilizar para restaurar el archivo si el archivo se modifica todos los días?

6.- ¿Qué utilidad de backup puede usted utilizar para encontrar archivos específicos para hacer su backup?

De un ejemplo del comando que usted utilizaría.

7.- ¿Cuáles son algunos de los directorios importantes en un sistema Linux que cambia con frecuencia y tienen la necesidad de ser backup regularmente?

CAPITULO 8

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

CONFIGURAR UNA IMPRESORA

OBJETIVOS

Al completar este Capítulo, usted podrá:

- Administrar Impresoras y los trabajos en cola.
- Instalar y configurar impresoras locales y de Redes.
- Entender el proceso de Impresión de Linux/UNiX.
- Citar las opciones de configuración de impresoras bajo Linux.
- Poder Configurar Impresoras bajo modo Gráfico y desde la Línea de Comando

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1.- ¿Dónde se van los trabajos en lo que esperan su turno a la impresora?

2.- ¿Qué utilidad grafica se puede utilizar para configurar la impresora GNOME/KDE?

3.- ¿Qué necesita estar instalado para imprimir por impresoras conectadas a una maquina de Windows?

4.- ¿Cuál es el comando para ver la cola de impresión?

5.- Enviaste tres trabajos a imprimir por Laser1; los archivos se encuentran en el queue (sus números son 1,2 y 3), ¿Cómo puedes hacer que el 3 imprima primero que todos?

INTRODUCCION

Este capítulo cubre lo básico de imprimir bajo Linux. La impresión es llevada a cabo bajo Linux por varias aplicaciones/utilidades archivos de configuración, el nombre del conjunto es LPR (Berkeley Line Print). Discutiremos la estructura de este software, como usarla para crear impresoras disponibles en la Red, como interactuar con maquinas bajo redes de otros sistemas operativos.

Imprimir bajo GNU/Linux

Esta sección comenta cómo imprimir ficheros, examinar la cola de impresión, eliminar trabajos de la cola, formatear ficheros antes de imprimirlos y configurar tu entorno de impresión.

Lo básico de Imprimir

La forma más simple (con mucho) de imprimir en el sistema operativo Linux es enviar el fichero a ser impreso directamente al dispositivo de impresión. Una manera de hacer esto es usar el comando cat. Como usuario root, uno puede hacer lo siguiente:

```
# cat tesis.txt > /dev/lp
```

En este caso, /dev/lp es un enlace simbólico al verdadero dispositivo de impresión (una matricial, láser, tipográfica o plotter). Mira la página del man ln(1) para más información acerca de enlaces simbólicos.

Para el propósito de la seguridad, sólo el usuario root y los usuarios de su mismo grupo como el demonio de impresión son capaces de escribir directamente a la impresora. Es por esto por lo que se tienen que usar comandos como lpr, lprm y lpq para acceder a la impresora.

Por esto, los usuarios tienen que usar lpr para imprimir un fichero. El comando lpr es responsable de preocuparse por el trabajo inicial para imprimir un fichero, pasando entonces el control a otro programa, lpd, el demonio de las impresoras de líneas.

Este demonio le dice entonces a la impresora cómo imprimir el fichero.

Cuando lpr es ejecutado, primero copia el fichero a un cierto directorio (el directorio de spool) donde el fichero permanece hasta que lpd lo imprime. Una vez se le dice a lpd que hay un fichero para imprimir, creará una copia de sí mismo (lo que los programadores llaman un ‘fork’). Esta copia imprimirá nuestro fichero mientras la copia original queda esperando otras peticiones. Esto permite que haya múltiples trabajos a la vez en una cola.

Las sintaxis de lpr (1) es bastante familiar:

```
$ lpr [ opciones ] [ nombre_archivo ... ]
```

Si no se especifica un nombre de fichero, lpr asume que la entrada será efectuada por la entrada estándar (normalmente el teclado o la salida de otro programa). Esto permite que el usuario redirija la salida de un programa al dispositivo de impresión. Por ejemplo:

```
$ cat tesis.txt | lpr
```

o algo más potente, como

```
$ pr -l60 tesis.txt | lpr
```

El comando lpr acepta varios argumentos en la línea de comandos que permiten al usuario controlar cómo funciona. Algunos de los argumentos más ampliamente usados son: -P printer especifica la impresora a usar, -h suprime la impresión de la página, burst, -s crea un enlace simbólico en lugar de copiar el fichero completo al directorio de spooling (útil para ficheros grandes), y -#num especifica el número de copias a imprimir.

Un ejemplo de interacción con lpr podría ser algo como:

```
$ lpr -#2 -sP dj tesis.txt
```

Este comando crearía un enlace simbólico al fichero tesis.txt en el directorio de spool de la impresora llamada dj, donde debería ser procesado por lpd. Además debería imprimir una segunda copia de tesis.txt. Para ver un listado de todas las opciones que reconoce lpr, ver la página del man lpr(1).

Viendo la cola de Impresión

Algunas veces es útil saber qué trabajos están actualmente en una cola de impresión particular. Esta es la única tarea del comando lpq. Para ver qué hay en la cola de la impresora por defecto (definida por /etc/printcap), se usa:

```
$ lpq
lp is ready and printing
Rank    Owner  Job  Files       Total Size
active  mwf    31   tesis.txt   682048 bytes
```

Cancelar un Trabajo de Impresión

Otra útil característica para cualquier sistema de impresión es la capacidad de cancelar un trabajo que ha sido 'encolado' anteriormente. Para hacer esto, usa lprm.

```
$ lprm -
```

El comando anterior cancela todos los trabajos de impresión que son propiedad del usuario que envió el comando. Se puede cancelar un trabajo de forma individual obteniendo primero el número del trabajo usando lpq, dando entonces el número a lprm. Por ejemplo:

```
$ lprm 31
```

Cancelaría el trabajo 31 (tesis.txt) en la impresora por defecto.

Elementos Misceláneos

Esta sección comenta algunos de las cosas de utilidad general que puedes querer saber sobre imprimir bajo GNU/Linux.

Formatear

Como la mayoría de los ficheros ASCII no están formateados para la impresión, es útil formatearlos de alguna manera antes de que sean realmente impresos. Esto puede incluir poner un título y número en cada página, poner márgenes, espaciado doble, sangría, o imprimir el fichero en múltiples columnas. Una forma común de hacer esto es usar un preprocesador de impresión como pr.

```
$ pr +4 -d –h "Ph.D. Thesis, 2nd Draft" -l60 tesis.txt | lpr
```

En el ejemplo de antes, pr tomará el fichero tesis.txt y saltaría las primeras tres páginas (+4), pondría la longitud de página en 60 líneas (-l60), doble espacio de la salida (-d), y añadiría la frase "Ph.D. Thesis, 2nd Draft" al principio de cada página (-h). lpr imprimiría entonces la salida de pr. Mira la página del manual para más información acerca de cómo usar pr.

La Variable de Entorno PRINTER

Todos los comandos del sistema de impresión de Linux aceptan la opción -P. Esta opción permite que el usuario especifique que impresora usar como salida. Si un usuario no especifica la impresora a usar, entonces se asumirá que la impresora por defecto es el dispositivo de salida.

En lugar de tener que especificar la impresora a usar cada vez que imprimes, puedes poner en la variable de entorno PRINTER el nombre de la impresora que quieres usar. Esto se hace de diferentes maneras por cada shell. Para el bash puedes hacerlo con:

$ PRINTER=”nombre_de_impresora”; export PRINTER

En csh, lo puedes hacer con:

 # setenv PRINTER “nombre_de_impresora”

Estos comandos pueden ser situados en tus scripts de login (.profile o .cshrc), o enviados en la línea de comandos. (Leer bash(1) y csh(1) para más información sobre las variables de entorno.)

Imprimir Ficheros PostScript

Imprimir ficheros PostScript en una impresora que tiene un intérprete PostScript es sencillo; simplemente usa lpr y la impresora se ocupará de todos los detalles por ti. Para aquellos que no tienen impresoras con capacidades PostScript, nos veremos obligados a usar otros medios. Por suerte, hay programas disponibles que pueden entender el PostScript y traducirlo a un lenguaje que la mayoría de las impresoras pueden comprender. Probablemente el más conocido de estos es Ghostscript.

La responsabilidad de Ghostscript es convertir todas las descripciones de un fichero PostScript a los comandos que la impresora entienda. Para imprimir un fichero PostScript usando Ghostscript, podrías hacer algo como:

$ gs -dNOPAUSE -sDEVICE=deskjet -sOutputFile=|lpr tesis.ps

Tenga en cuenta que en el ejemplo anterior hemos enviado la salida de Ghostscript hacia el comando lpr usando la opción -sOutputFile.

Ghostview es un interfase de Ghostscript para el Sistema X Window. Te permite previsualizar un fichero PostScript antes de que lo imprimas. Ghostview y Ghostscript pueden ser bajados desde ftp://prep.ai.mit.edu/pub/gnu/.

Imprimir Ficheros TeX

Una de las maneras más fáciles de imprimir los ficheros TeX es convertirlos a PostScript y entonces imprimirlos usando Ghostscript. Para hacerlo, primero necesitas convertirlos de TeX a un formato conocido como DVI (siglas de DeVice-Independent, independiente del dispositivo). Puedes hacerlo con el comando tex (1). Entonces necesitas convertir el dispositivo DVI a PostScript usando dvips. Todo esto debería ser de la siguiente manera cuando lo escribas.

$ tex tesis.tex
$ dvips tesis.dvi

Ahora ya estás preparado para imprimir los ficheros PostScript resultantes tal como se describe anteriormente.

Imprimir Archivos Formateados con troff

$ groff -Tascii tesis.tr | lpr

o, si lo prefieres,

$ groff tesis.tr > tesis.ps

Y entonces imprimir el fichero PostScript como se describió anteriormente.

Respuestas a Preguntas Frecuentes (FaQ)

P1. ¿Cómo puedo prevenir el efecto de escalera (staircase effect)?

R1. El efecto de escalón ocurre por la manera en que algunas impresoras esperan que se acaben las líneas. Algunas impresoras quieren líneas que terminen con un retorno_de_carro/avance_de_línea CR/LF (estilo DOS) en lugar de con la secuencia por defecto de los sistemas tipo UNiX (sólo un avance de línea, LF). La manera más sencilla de solucionar esto es mirar si tu impresora puede conmutar entre ambos estilos de alguna manera (un interruptor DIP o mandando una secuencia de escape).

Para hacer la segunda necesitas crear un filtro (ver P2 y Foster95b). Una forma rápida de arreglarlo es usar un filtro en la línea de comandos. un ejemplo de esto podría ser:
$ cat tesis.txt | todos | lpr

P2. ¿Qué es un filtro?

R2. Un filtro es un programa que lee de la entrada estándar (stdin), realiza alguna acción sobre esa entrada y escribe en la salida estándar (stdout). Los filtros se usan para montón de cosas, incluyendo el procesado de textos.

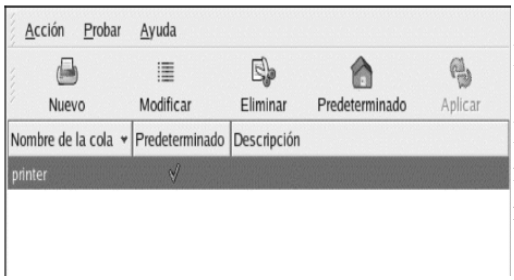
P3. ¿Qué es un filtro mágico?

R3. Un filtro mágico es un filtro que realiza una acción basada en el tipo de un fichero. Por ejemplo, si el fichero es puro texto, entonces simplemente imprimirá el fichero usando los métodos normales. Si el fichero es PostScript, o cualquier otro formato, podría imprimirlo usando otro método (ghostscript).

Configurar una Impresora desde el X

La Herramienta de configuración de impresoras permite a los usuarios configurar una impresora en Red Hat Linux. Esta herramienta ayuda a mantener el archivo de configuración de la impresora, los directorios spool y los filtros de impresión.

Desde la versión 9, Red Hat Linux, CUPS es el sistema de impresión predeterminado. Sin embargo, todavía se proporciona el sistema de impresión por defecto anterior, LPRng. Si el sistema fue actualizado desde una versión anterior de Red Hat Linux que usaba LPRng, el proceso de actualización no reemplaza LPRng con CUPS; el sistema continuará usando LPRng.



Si un sistema fue actualizado desde una versión anterior de Red Hat Linux que usaba CUPS, el proceso de actualización mantiene las colas configuradas y el sistema continuará usando CUPS. La Herramienta de configuración de impresoras configura ambos sistemas de impresión CUPS y LPRng, dependiendo de cual se configure a usar en el sistema. Cuando aplique los cambios, configurará el sistema de impresión activo.

Para usar la Herramienta de configuración de impresoras debe tener privilegios como root. Para iniciar la aplicación, seleccione Botón de menú principal (en el Panel) => Configuración del sistema => Impresión, o escriba el comando redhat-config-printer. Este comando determina automáticamente si ejecutará la versión gráfica o la versión basada en texto dependiendo de si el comando es ejecutado desde el ambiente gráfico X Window o desde una consola basada en texto.

Puede forzar a la Herramienta de configuración de impresoras a ejecutarse como una aplicación basada en texto usando el comando redhat-config-printer-tui desde el intérprete de comandos.

Importante: No modifique el archivo /etc/printcap o los archivos en el directorio /etc/cups/. Cada vez que el demonio de impresión (lpd o cups) es iniciado o reiniciado, se crean dinámicamente nuevos archivos de configuración. Los archivos son creados dinámicamente cuando se aplican cambios con la Herramienta de configuración de impresoras también.

Si está usando LPRng y desea añadir una impresora sin usar la Herramienta de configuración de impresoras, modifique el archivo /etc/printcap.local. Las entradas en /etc/printcap.local no son desplegadas en la herramienta de configuración de impresoras pero son leídas por el demonio de impresión. Si actualizó su sistema desde una versión anterior de Red Hat Linux, su archivo de configuración existente fue convertido al nuevo formato usado por esta aplicación. Cada vez que se genera un nuevo archivo de configuración, el archivo viejo es guardado como /etc/printcap.old.

Si está usando CUPS, la Herramienta de configuración de impresoras no despliega las colas o compartidas que no hayan sido configuradas con la Herramienta de configuración de impresoras; sin embargo, no las eliminará de los archivos de configuración.

Se pueden configurar los siguientes tipos de colas de impresión:

Conectada-localmente- Una impresora directamente conectada al computador a través de un puerto paralelo o USB.

Conectada CUPS(IPP)- Una impresora conectada a un sistema CUPS diferente que puede ser accesada sobre una red TCP/IP (por ejemplo, una impresora conectada a otro sistema Red Hat Linux corriendo CUPS en la red).

Conectada UNIX(LPD)- Una impresora conectada a un sistema UNIX diferente que puede ser accesada sobre una red TCP/IP (por ejemplo, una impresora conectada a otro sistema Red Hat Linux corriendo LPD en la red).

Conectada Windows(SMB) Una impresora conectada a un sistema diferente el cual está compartiendo una impresora sobre una red SMB (por ejemplo, una impresora conectada a una máquina otro OS).

Conectada Novell(NCP)- Una impresora conectada a un sistema diferente el cual usa la tecnología de red Novell NetWare.

Conectada JetDirect- Una impresora conectada directamente a la red a través de HP JetDirect en vez de a un computador.

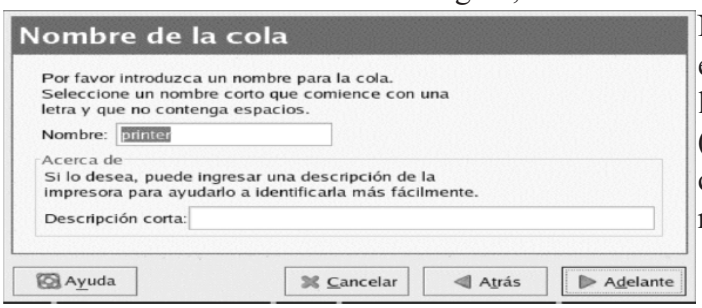
Importante: Si agrega o modifica una cola de impresión, debe aplicar los cambios para que tomen efecto.

Al hacer clic en el botón Aplicar guarda cualquier cambio que haya realizado y reinicia el demonio de impresión. Los cambios no son escritos al archivo de configuración hasta que el demonio de impresión no sea reiniciado. Alternativamente, puede seleccionar Acción => Aplicar.

Añadir una Impresora Local

Para añadir una impresora local, tal como una conectada al puerto paralelo o USB en su computador, haga clic en Nuevo en la ventana principal de la Herramienta de configuración de impresoras para mostrar la ventana en la Figura. Haga clic en Siguiente para proceder.

En la ventana mostrada en la Figura, introduzca un nombre único para la impresora en el campo de texto Nombre. El nombre de la impresora no puede contener espacios y debe comenzar con una letra. El nombre de la impresora puede contener letras, números, guiones (-), y rayas (_). Opcionalmente, introduzca una descripción corta para la impresora, la cual puede contener espacios.



Después de hacer clic en Siguiente, aparecerá la

Figura. Seleccione Conectado localmente desde el menú Seleccionar el tipo de cola y seleccione el dispositivo. El dispositivo es usualmente /dev/lp0 para una impresora paralela o /dev/usb/lp0 para una impresora USB. Si no aparece ningún dispositivo en la lista, haga clic en Volver a escanear dispositivos para revisar nuevamente la máquina o haga clic en Dispositivo personalizado para especificarlo manualmente. Haga clic en Siguiente para continuar.

Añadir una Impresora IPP

Una impresora de red IPP es una impresora conectada a un sistema Linux diferente en la misma red ejecutando CUPS o una impresora configurada para usar IPP en otro sistema operativo. Por defecto, la Herramienta de configuración de impresoras navega la red en busca de impresoras compartidas IPP. (Esta opción se puede cambiar seleccionando Acción => Compartir desde el menú.) Cualquier impresora IPP compartida aparecerá en la ventana principal.

Si tiene un cortafuego (firewall) configurado en el servidor de impresión, este debe ser capaz de enviar y recibir conexiones en el puerto de entrada UDP 631. Si tiene un cortafuego configurado en el cliente (la computadora enviando la petición de impresión), se le debe permitir enviar y aceptar conexiones en el puerto 631.

Si desactivó la característica automática de navegación, todavía puede agregar una impresora de red IPP haciendo clic en el botón Nuevo en la ventana principal de la Herramienta de configuración de impresoras para desplegar la ventana. Haga clic en Siguiente para proceder.

En la ventana que se le muestra, introduzca un nombre único para la impresora en el campo de texto Nombre. El nombre de la impresora no puede contener espacios y debe comenzar con una letra. El nombre de la impresora puede contener letras, números, guiones (-), y rayas (_). Opcionalmente, introduzca una descripción corta para la impresora, la cual puede contener espacios.

Después de hacer clic en Siguiente, aparecerá unafigura similar a las anteriores de SAMBA. Seleccione Conectada CUPS (IPP) desde el menú Seleccionar un tipo de cola.

Aparecen los campos de texto para las opciones siguientes:

Servidor- El nombre o dirección IP de la máquina remota a la cual la impresora está conectada.
Ruta- La ruta de la cola de impresión en la máquina remota.

Haga clic en Siguiente para continuar.

El próximo paso es seleccionar el tipo de impresora.

Importante: El servidor de impresión de red IPP debe permitir conexiones desde el sistema local. Consulte la Sección más adelante para más información.

Añadir una Impresora UNiX (LPD) Remota

Para agregar una impresora UNiX remota, tal como una conectada a un sistema Linux diferente en la misma red, haga clic en el botón Nuevo en la ventana principal de la Herramienta de configuración de impresoras. Aparecerá la ventana mostrada en la figuras anterior de elección del tipo de impresora. Haga clic en Siguiente para proceder.

Al igual que en la ventana mostrada anteriormente, introduzca un nombre único para la impresora en el campo de texto Nombre. El nombre de la impresora no puede contener espacios y debe comenzar con una letra. El nombre de la impresora puede contener letras, números, guiones (-), y rayas (_). Opcionalmente,

introduzca una descripción corta para la impresora, la cual puede contener espacios. Seleccione Conectada UNIX (LPD) desde el menú Seleccionar el tipo de cola y haga clic en Siguiente.

Aparecen los campos de texto para las opciones siguientes:

Servidor- El nombre o la dirección IP de la máquina remota a la cual la impresora está conectada.

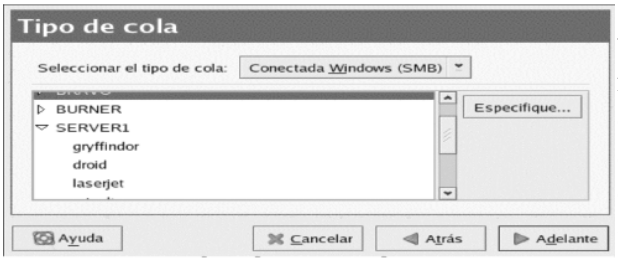
Cola- La cola de impresión remota. La impresora por defecto es usualmente lp.

Haga clic en Siguiente para continuar. El próximo paso es seleccionar el tipo de impresora.

Importante: El servidor de impresión remoto debe poder aceptar trabajos de impresión desde el sistema local.

Añadir una Impresora Samba (SMB)

Para añadir una impresora que es accesada usando el protocolo SMB, haga clic en el botón Nuevo en la ventana principal de la Herramienta de configuración de impresoras. Aparecerá la ventana mostrada en la Figura. Haga clic en Siguiente para proceder.



En la ventana mostrada en Figura, introduzca un nombre único para la impresora en el campo de texto Nombre. El nombre de la impresora no puede contener espacios y debe comenzar con una letra. El nombre de la impresora puede contener letras, números, guiones (-), y rayas (_). Opcionalmente, introduzca una descripción corta para la impresora, la cual puede contener espacios.

Seleccione Conectada a Windows (SMB) desde el menú Seleccionar un tipo de cola y haga clic en Siguiente. Si la impresora está conectada a un sistema Microsoft Windows, seleccione este tipo de cola.

Como se muestra en la Figura, las comparticiones SMB son detectadas y listadas automáticamente. Haga clic en la flecha al lado de cada nombre de compartición para ampliar la lista. Desde la lista ampliada, seleccione una impresora.

Si la impresora que está buscando no aparece en la lista, haga clic en el botón Especificar a la derecha. Aparecerán los campos de texto para las siguientes opciones:

Grupo de trabajo- El nombre del grupo de trabajo Samba para la impresora compartida.

Servidor- El nombre del servidor compartiendo la impresora.

Compartir- El nombre de la impresora compartida en la cual desea imprimir. Este nombre debe ser el mismo que el definido como la impresora Samba en la máquina Windows remota.

Nombre de usuario- El nombre de usuario con el que debe conectarse para acceder a la impresora. Este usuario debe existir en el sistema Windows y el usuario debe tener permiso para accesar la impresora. El nombre de usuario predeterminado es típicamente guest para los servidores Windows, o nobody para los servidores Samba.

Contraseña- La contraseña (si se necesita) para el usuario especificado en el campo Nombre de usuario.

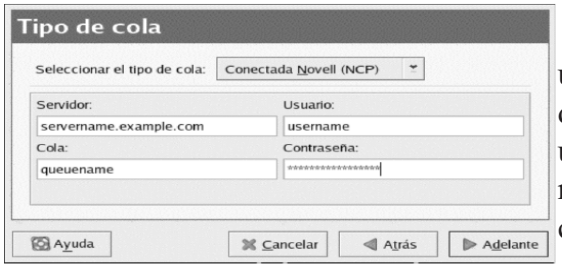
Haga clic en Siguiente para continuar. La Herramienta de configuración de impresoras luego intenta conectarse a la impresora compartida. Si la impresora compartida requiere un nombre de usuario y contraseña, aparecerá una ventana de diálogo pidiéndole que proporcione un nombre de usuario válido y contraseña. Si se especificó un nombre de compartición incorrecto, puede cambiarlo aquí también. Si un nombre de grupo de trabajo es requerido para conectarse a la compartición, se puede especificar en esta caja de diálogo. Esta ventana de diálogo es la misma que la mostrada cuando se hace clic sobre el botón Especificar.

El próximo paso es seleccionar el tipo de impresora.

Aviso: Si requiere un nombre de usuario y una contraseña, estos son almacenados descifrados en archivos que sólo pueden ser accesados por root y lpd. Por tanto, es posible para otros conocer el nombre de usuario y la contraseña si ellos tienen acceso root. Para evitar esto, el nombre de usuario y la contraseña para accesar a la impresora deberían ser diferentes del nombre de usuario y la contraseña usados por la cuenta de usuario en el sistema local Red Hat Linux. Si son diferentes, entonces el único riesgo de seguridad posible será el uso no autorizado de la impresora. Si hay compartici-ones de archivo del servidor, se recomienda que ellos también tengan una contraseña diferente de la de la cola de impresión.

Añadir una Impresora Novell NetWare (NCP)

Para añadir una impresora Novell NetWare (NCP), haga clic en el botón Nuevo en la ventana principal de la Herramienta de configuración de impresoras. Aparecerá la ventana mostrada en Figura. Haga clic en Siguiente para proceder.



En la ventana mostrada en Figura, introduzca un nombre único para la impresora en el campo de texto Nombre. El nombre de la impresora no puede contener espacios y debe comenzar con una letra. El nombre de la impresora puede contener letras, números, guiones (-), y rayas (_). Opcionalmente, introduzca una descripción corta para la impresora, la cual puede contener espacios.

Seleccione Conectada Novell (NCP) del menú Seleccionar un tipo de cola.

Aparecerán campos de texto para las opciones siguientes:

Servidor- El nombre de la máquina o dirección IP del sistema NCP al cual la impresora está conectada.
Cola- La cola remota para la impresora en el sistema NCP.
Usuario- El nombre del usuario que debe conectarse para accesar la impresora.
Contraseña- La contraseña para el usuario especificado en el campo Usuario.

El próximo paso es seleccionar el tipo de impresora. Vaya a la Sección 27.7 para continuar.

Aviso: Si requiere un nombre de usuario y una contraseña, estos son almacenados descifrados en archivos que sólo pueden ser accesados por root y lpd. Por tanto, es posible para otros conocer el nombre de usuario y la contraseña si ellos tienen acceso root. Para evitar esto, el nombre y la contraseña del usuario para accesar la impresora deberían ser diferentes al usados por la cuenta de usuario en el sistema local Red Hat Linux. Si son diferentes, entonces el único riesgo de seguridad posible será el uso no autorizado de la impresora. Si hay comparticiones de archivo del servidor, se recomienda que ellos también tengan una contraseña diferente de la de la cola de impresión.

Añadir una Impresora JetDirect

Para agregar una impresora JetDirect, haga clic en el botón Nuevo en la ventana principal de la Herramienta de configuración de impresoras. Aparecerá la ventana con una Figura igual que la anterior. Haga clic en Siguiente para proceder.

En la ventana mostrada en Figura, introduzca un nombre único para la impresora en el campo de texto Nombre. El nombre de la impresora no puede contener espacios y debe comenzar con una letra. El nombre de la impresora puede contener letras, números, guiones (-), y rayas (_). Opcionalmente, introduzca una descripción corta para la impresora, la cual puede contener espacios.

Seleccione Conectada JetDirect desde el menú Seleccionar un tipo de cola y haga clic en Siguiente.

Aparecerán los campos de texto para las siguientes opciones:

Impresora- El nombre de la máquina o dirección IP de la impresora JetDirect.

Puerto- El puerto en la impresora JetDirect que está escuchando por trabajos de impresión. El puerto predeterminado es 9100.

El próximo paso es seleccionar el tipo de impresora.

Elección del Modelo de la Impresora

Después de seleccionar el tipo de cola de impresión, el próximo paso es seleccionar el modelo de la impresora. Aparecerá una ventana similar a la que hemos visto hasta ahora. Si no fue detectado automáticamente, seleccione el modelo de la lista. Las impresoras son divididas por fabricantes. Seleccione el nombre del fabricante desde el menú. Los modelos de impresoras son actualizados cada vez que un nuevo fabricante es seleccionado. Seleccione el modelo de impresora de la lista.

El controlador de la impresora recomendado es escogido basado en el modelo de impresora seleccionado. El controlador de la impresora procesa los datos que desea imprimir en un formato que la impresora pueda entender. Puesto que hay una impresora local conectada a su computador, necesita un controlador de impresora para procesar los datos que son enviados a la misma.

Si está configurando una impresora remota (IPP, LPD, SMB, o NCP), el servidor de impresión remoto usualmente tiene su propio controlador de impresión. Si selecciona un controlador de impresión adicional en su computador local, los datos son filtrados múltiples veces y convertidos a un formato que la impresora no puede entender.

Para asegurarse de que los datos no son filtrados más de una vez, primero trate de seleccionar Genérico como el fabricante y Cola de impresora sin formato o Impresora PostScript como el modelo de impresora. Después de aplicar los cambios, imprima una página de prueba para probar la nueva configuración. Si la prueba falla, el servidor de impresión remoto puede que no tenga un controlador de impresora configurado. Intente seleccionando un controlador de acuerdo al fabricante y modelo de la impresora remota, aplique los cambios e imprima una página de prueba.

Sugerencia: Puede seleccionar un controlador de impresora diferente después de añadir una impresora iniciando la herramienta de configuración de impresoras, seleccione la impresora desde la lista y haga clic en Modificar. Luego haga clic en la pestaña Controlador, seleccione un controlador diferente y luego aplique los cambios.

Confirmar la Configuración de la Impresora

El último paso es confirmar la configuración de su impresora. Haga clic en Aplicar para agregar la cola de impresión si las configuraciones son correctas. Haga clic en Anterior para modificar la configuración de la impresora.

Presione el botón Aplicar en la ventana principal para guardar sus cambios y reiniciar el demonio de impresión. Después de aplicar los cambios, imprima una página de prueba para asegurarse de que la configuración es correcta. Si necesita imprimir caracteres fuera del conjunto ASCII básico (incluyendo aquellos usados por idiomas tal como el japonés), debe revisar las opciones de su controlador y seleccionar Preparar PostScript. También puede configurar opciones tales como el tamaño del papel si modifica la cola de impresión después de haberla agregado.

Imprimir una Página de Prueba

Después de haber configurado su impresora, debería imprimir una página de prueba para asegurarse de que su impresora funciona perfectamente. Para imprimir una página de prueba, seleccione la impresora que

desea probar desde la lista de impresoras, luego seleccione la página de prueba apropiada desde el menú Probar.

Si cambia el controlador de la impresora o modifica las opciones de la impresora, debería imprimir una página de prueba para verificar la nueva configuración.

Modificar Impresoras Existentes

Para borrar una impresora existente, seleccione la impresora y haga clic en el botón Eliminar en la barra de herramientas. La impresora será eliminada de la lista de impresoras. Haga clic en Aplicar para guardar los cambios y reiniciar el demonio de impresión.

Para establecer la impresora por defecto, seleccione la impresora desde la lista y presione el botón Predeterminado en la barra de herramientas. Aparecerá el icono de la impresora por defecto en la columna Predeterminado de la impresora predeterminada en la lista.

Después de agregar una impresora, las propiedades se pueden modificar seleccionando la impresora desde la lista de impresoras y haciendo clic en el botón Modificar. La ventana contiene los valores actuales para la impresora seleccionada. Efectúe los cambios necesarios y luego pulse el botón OK. Haga clic Aplicar en la ventana principal de la Herramienta de configuración de impresoras para guardar los cambios y reiniciar el demonio de impresión.

Nombre de la Cola

Para renombrar una impresora o cambiar su descripción, cambie el valor en la pestaña Nombre de la cola. Presione OK para volver a la ventana principal. El nombre de la impresora debería cambiar en la lista de impresoras. Haga clic en Aplicar para guardar los cambios y reiniciar el demonio de impresión.

Tipo de Cola

La pestaña Tipo de cola muestra el tipo de cola que fue seleccionada cuando se agregó la impresora y sus propiedades. El tipo de cola de la impresora se puede cambiar o simplemente las propiedades. Después de realizar las modificaciones, haga clic en OK para volver a la ventana principal. Pulse Aplicar para guardar los cambios y reiniciar el demonio de impresión.

Dependiendo del tipo de cola escogido, se desplegarán opciones diferentes. Consulte la sección apropiada sobre Añadir impresoras para una descripción de las opciones.

Controlador de Impresoras

La pestaña Controlador de impresoras muestra cúal controlador de impresora está siendo usado actualmente. Si se cambia, haga clic en OK para volver a la pantalla principal. Pulse el botón Aplicar para guardar los cambios y reiniciar el demonio de impresión.

Opciones del Controlador

La pestaña Opciones de controladores muestra las opciones avanzadas del controlador. Las opciones varían para cada controlador de impresoras. Las opciones comunes incluyen:

Envie Form-Feed (FF):

Debería ser seleccionada si la última página del trabajo de impresión no sale de la impresora (por ejemplo, la luz de ‘form feed’ está brillando). Si esto no funciona, intente seleccionando Envie un End-of-Transmission (EOT). Algunas impresoras requieren que ambos Envie Form-Feed (FF) y Envie un Endof-

Transmission (EOT) esten seleccionados para expulsar la página. Esta opción sólo está disponible con el sistema de impresión LPRng.

Envie un End-of-Transmission (EOT) debería ser seleccionado si 'form-feed' no funciona. Consulte Envie Form-Feed (FF) mostrado arriba. Esta opción sólo está disponible con el sistema de impresión LPRng.

Asume que los datos desconocidos son texto:
Debería estar seleccionado si el controlador de impresora no reconoce algunos de los datos enviados a él. Solamente seleccione esta opción si hay problemas imprimiendo. Si esta opción es seleccionada, el controlador de impresión asume que cualquier dato que no pueda reconocer es texto e intenta imprimirlo como texto. Si esta opción es seleccionada junto con Convertir texto a PostScript, el controlador de impresión asume que los datos desconocidos son texto y lo convierte a PostScript. Esta opción está disponible sólo con el sistema de impresión LPRng.

Preparar PostScript:
Debería estar seleccionado si se están enviando carácteres fuera del conjunto básico ASCII a la impresora pero no se están imprimiendo correctamente (tal como caracteres japoneses). Esta opción traduce las fuentes no-estándar PostScript para que se puedan imprimir correctamente.

Si la impresora no soporta las fuentes que usted está tratando de imprimir, inténtelo seleccionando esta opción. Por ejemplo, seleccione esta opción para imprimir fuentes japonesas a una impresora no-japonesa. Se requiere tiempo adicional para realizar esta acción. No la seleccione a menos que tenga problemas imprimiendo las fuentes correctas.

También seleccione esta opción si la impresora no puede manejar PostScript de nivel 3. Esta opción lo convierte a PostScript de nivel 1.

Prefiltrado GhostScript:
Le permite seleccionar Sin prefiltrado, Convertir a PS de nivel 1, o Convertir a PS de nivel 2 en caso de que la impresora no pueda manejar ciertos niveles de PostScript. Esta opción sólo está disponible si el controlador de PostScript es usado con el sistema de impresión CUPS.

Convertir texto a PostScript:
Está seleccionado por defecto. Si la impresora puede imprimir texto plano, intente quitar esta opción cuando esté imprimiendo documentos de texto plano para reducir el tiempo que toma en imprimir. Si está usando el sistema CUPS, esto no es una opción porque el texto siempre se convierte a PostScript.

Tamaño de la página:
Le permite seleccionar el tamaño del papel. Las opciones incluyen Carta US, Legal, A3 y A4.

Localización del filtro efectivo:
Por defecto es C. Si se está imprimiendo caracteres japoneses, seleccione ja_JP. De lo contrario, accepte el valor por defecto de C.

Fuente de medios:
Por defecto está en Impresora predeterminada. Cambie esta opción para usar papel desde una bandeja diferente.

Para modificar las opciones de los controladores, haga click en OK para volver a la pantalla principal. Pulse Aplicar para guardar los cambios y reiniciar el demonio de impresión.

Guardar el Archivo de Configuración

Cuando la configuración de la impresora es guardada usando la Herramienta de configuración de impresoras, la aplicación crea su propio archivo de configuración que es usado para crear los archivos en el directorio /etc/cups (o el archivo /etc/printcap que lpd lee). Puede usar las opciones de línea de comando para guardar o restaurar el archivo de la Herramienta de configuración de impresoras. Si el directorio /etc/cups o el archivo /etc/printcap es guardado y restaurado a las mismas ubicaciones, la configuración de la impresora no es restaurada porque cada vez que el demonio de impresión es iniciado, crea un nuevo archivo /etc/printcap desde el archivo especial de configuración Herramienta de configuración de impresoras. Cuando se esté haciendo un respaldo de los archivos de configuración del sistema, use el método siguiente para guardar los archivos de configuración de la impresora(s). Si el sistema está usando LPRng y se han añadido configuraciones personalizadas en el archivo /etc/printcap.local, debería guardarse como parte del respaldo también.

Para guardar la configuración de su impresora, escriba este comando como root:
/usr/sbin/redhat-config-printer-tui —Xexport > settings.xml
Su configuración es guardada al archivo settings.xml.

Si se guarda este archivo, se puede usar para restaurar las configuraciones de la impresora. Esto es muy útil si la configuración de la impresora es borrada, si RedHat Linux es reinstalado o si se necesita la misma configuración de impresoras en múltiples sistemas. El archivo debería guardarse en un sistema diferente antes de ser reinstalado. Para restaurar la configuración, escriba este comando como root:
/usr/sbin/redhat-config-printer-tui —Ximport < settings.xml

Si ya tiene un archivo de configuración (ya ha configurado una o más impresoras en el sistema) e intenta importar otro archivo de configuración, el archivo de configuración existente será sobreescrito. Si quiere conservar su configuración existente y agregar la configuración en el archivo guardado, puede mezclar los archivos con el comando siguiente (como root):
/usr/sbin/redhat-config-printer-tui —Ximport —merge < settings.xml

Su lista de impresoras consistirá de las impresoras que ha configurado en el sistema así como también las impresoras que importó desde el archivo de configuración guardado. Si el archivo de configuración importado tiene una cola de impresión con el mismo nombre de una cola existente en el sistema, la cola de impresión desde el archivo importado sobre escribirá la impresora existente.

Después de importar el archivo de configuración (con o sin el comando merge), debe reiniciar el demonio de impresión. Si está usando CUPS, escriba el comando:
/sbin/service cups restart

Si está usando LPRng, use el comando:
/sbin/service lpd restart

Configuración de la Impresora desde la Línea de Comandos

Si no tiene instalado el sistema X y no desea usar la versión basada en texto, puede añadir una impresora a través de la línea de comandos. Este método es muy útil si desea añadir una impresora desde un script o en la sección %post de una instalación de arranque rápido (kickstart).

Añadir una Impresora local

Para agregar una impresora:
redhat-config-printer-tui —Xadd-local opciones

Opciones:

--device=nodo	(Requerido) El nodo dispositivo a ser usado. Por ejemplo, /dev/lp0.
--make=make	(Requerido) La cadena de caracteres IEEE 1284 MANUFACTURER o nombre del fabricante de la impresora como en la base de datos foomatic si la cadena de caracteres no está disponible.
--model=modelo	(Requerido) La cadena de caracteres IEEE 1284 MODEL o el modelo de la impresora listada en la base de datos foomatic si la cadena de caracteres no está disponible.
--name=nombre	(Opcional) El nombre dado a la nueva cola. Si alguno no está dado, será usado un nombre basado en el nodo dispositivo (tal como "lp0").
--as-default	(Opcional) Configure esto como la cola predeterminada.

Si está usando CUPS como el sistema de impresión (predeterminado), después de añadir la impresora, use el comando siguiente para iniciar/reiniciar el demonio de impresión:

service cups restart

Si está usando LPRng como el sistema de impresión, después de agregar la impresora, use el comando siguiente para iniciar/reiniciar el demonio de impresión:

service lpd restart

Eliminar una Impresora Local

Una cola de impresión también se puede eliminar a través de la línea de comandos.

Como usuario root, para eliminar la cola de impresión:

redhat-config-printer-tui —Xremove-local opciones

Opciones:

--device=nodo	(Requerido) El nodo dispositivo usado tal como /dev/lp0.
--make=make	(Requerido) La cadena de caracteres de IEEE 1284 MANUFACTURER, o (si no hay ninguna disponible) el nombre del fabricante como aparece en la base de datos foomatic.
--model=modelo	(Requerido) La cadena de caracteres de IEEE 1284 MODEL, o (si no hay ninguna disponible) el modelo de la impresora como aparece listado en la base de datos foomatic.

Si está usando el sistema de impresión CUPS (predeterminado), después de eliminar la impresora de la configuración de la Herramienta de configuración de impresoras, reinicie el demonio de impresión para que los cambios tengan efecto:

service cups restart

Si está usando el sistema de impresión LPRng, después de eliminar la impresora desde la configuración en la Herramienta de configuración de impresoras, reinicie el demonio de impresión para que los cambios tengan efecto:

service lpd restart

Si está usando CUPS, ha removido todas las impresoras y no desea ejecutar el demonio de impresión otra vez, ejecute el comando siguiente:

service cups stop

Si está usando LPRng, ha eliminado todas las impresoras y no desea ejecutar el demonio de impresión otra vez, ejecute el comando:

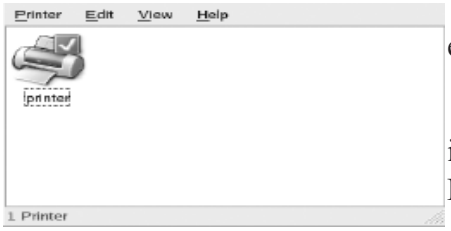
service lpd stop

Administración de Trabajos de Impresión

Cuando usted envía un trabajo de impresión al demonio de impresión, tal como imprimir un archivo de texto desde Emacs o imprimir una imagen desde El GIMP, el trabajo de impresión es añadido al spool de la

cola de impresión. El spool de la cola de impresión es una lista de los trabajos de impresión que han sido enviados a la impresora e información acerca de cada petición de impresión, tal como el estado de la petición, el nombre del usuario de la persona que envió la petición, el nombre de la máquina que lo envió, el número de trabajo, etc.

Si está ejecutando un ambiente gráfico de escritorio, haga clic en el icono Administrador de impresión en el panel para arrancar el Administrador de impresión GNOME.

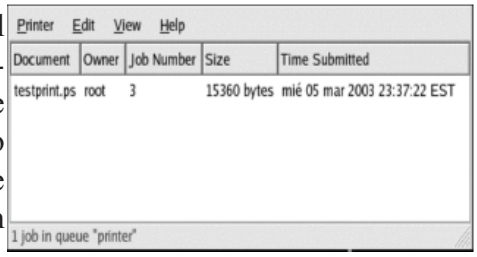


También se puede arrancar seleccionando Botón de menú principal (en el Panel) => Herramientas del sistema => Administrador de impresión.

Para cambiar las configuraciones de la impresora, presione sobre el icono de la impresora con el botón derecho del ratón y seleccione Propiedades. La Herramienta de configuración de impresoras es iniciada.

Haga doble clic sobre una impresora configurada para ver el spool de la cola como se muestra en la Figura. Para cancelar un trabajo específico de impresión listado en el Administrador de impresión GNOME, selecciónelo desde la lista y pulse Modificar => Cancelar documentos desde el menú desplegable.

Si hay trabajos activos de impresión en el spool, aparecerá un icono de notificación de impresión en el Área de notificación del panel del panel del escritorio, como se muestra en la Figura. Debido a que se verifica por trabajos de impresión activos cada cinco segundos, puede que no sea desplegado el icono para trabajos de impresión cortos. Haciendo clic en el icono de notificación de impresión inicia el Administrador de impresión GNOME para mostrar una lista de los trabajos de impresión actuales.



También ubicado en el Panel está un icono Administrador de impresión. Para imprimir un archivo desde Nautilus, navegue hasta la ubicación del archivo y arrastre y suéltelo en el icono de Administrador de impresión en el Panel. Se despliega la ventana mostrada en la Figura 27-16. Haga clic en OK para comenzar a imprimir el archivo.

Para ver una lista de los trabajos de impresión en el spool de impresión desde el intérprete de comandos, escriba el comando lpq. Las últimas pocas líneas de la salida de este comando, serán similares a lo siguiente:

Ejemplo de salida de lpq:

Rank	Owner/ID	Class	Job	Files	Size	Time
active	user@localhost+902	A	902	sample.txt	2050	01:20:46

Si desea cancelar un trabajo de impresión, encuentre el número del trabajo de la petición con el comando lpq y luego use el comando lprm número de trabajo. Por ejemplo, lprm 902 cancelará el trabajo. Debe tener los permisos adecuados para poder cancelar un trabajo de impresión. Usted no puede cancelar trabajos de impresión que fueron iniciados por otros usuarios a menos que usted se haya conectado como root en la máquina a la cual la impresora está conectada.

También puede imprimir un archivo directamente desde el intérprete de comandos. Por ejemplo, el comando lpr sample.txt imprimirá el archivo de texto sample.txt. El filtro de impresión determina qué tipo de archivos es y lo convierte a un formato de impresión que la impresora pueda entender.

Compartir una Impresora

La habilidad de la Herramienta de configuración de impresoras de compartir las opciones de configuración sólo puede ser usada si está usando el sistema de impresión CUPS. Para configurar impresoras compartidas en un sistema LPRng, consulte la sección dedicada a este tema.

El permitir a otros usuarios en un computador diferente en la red imprimir a una impresora configurada para su sistema se llama compartir la impresora. Por defecto, las impresoras configuradas con la Herramienta de configuración de impresoras no están compartidas.

Para compartir una impresora configurada, arranque la Herramienta de configuración de impresoras y seleccione una impresora desde la lista. Luego seleccione Acción => Compartir desde el menú desplegable.

Nota: Si una impresora no está seleccionada, Acción => Compartir sólo muestra las opciones de compartir en el sistema mostradas normalmente bajo la pestaña General.

En la pestaña Cola, seleccione la opción para hacer la cola disponible a otros usuarios. Después de seleccionar compartir la cola, por defecto, todas las máquinas pueden imprimir a la impresora compartida. Permitir a todos los sistemas en la red imprimir a la cola puede ser un poco peligroso, especialmente si el sistema está directamente conectado a la Internet. Se recomienda que esta opción sea cambiada seleccionando la entrada Todas las máquinas y haciendo clic en el botón Modificar para desplegar la próxima ventana.

Si tiene un cortafuego (firewall) configurado en el servidor de impresión, éste debe ser capaz de enviar y recibir conexiones en el puerto UDP, 631. Si tiene un cortafuego configurado en el cliente (la computadora enviando la petición de impresión), éste debe poder enviar y aceptar conexiones en el puerto 631.

La pestaña General establece configuraciones para todas las impresoras, incluyendo aquellas que no son visualizadas con la Herramienta de configuración de impresoras. Hay dos opciones:

Encuentra automáticamente colas compartidas remotas:
Seleccionada como predeterminada, esta opción activa la navegación IPP, lo cual significa que cuando otras máquinas en la red difunden las colas que tienen, las colas son automáticamente agregadas a la lista de impresoras disponibles en el sistema; no se requiere configuración adicional para una impresora que es encontrada desde la navegación IPP. Esta opción no comparte automáticamente las impresoras configuradas en el sistema local.

Activar el protocolo LPD:
Esta opción permite a la impresora recibir trabajos de impresión de clientes configurados para usar el protocolo LPD usando el servicio cups-lpd, el cual es un servicio xinetd.

Aviso: Si esta opción está activada, todos los trabajos de impresión son aceptados desde todas las máquinas si son recibidos desde un cliente LPD.

Compartir una Impresora con LPRng

Si está ejecutando el sistema de impresión LPRng, compartir debe ser configurado manualmente. Para permitir a los sistemas en la red imprimir a una impresora configurada en un sistema RedHat Linux, siga los pasos siguientes:

1. Cree el archivo /etc/accepthost. En este archivo, añada la dirección IP o el nombre de la máquina al que desea permitir el acceso a la impresión, con una línea por IP o nombre de máquina.

2. Quite los comentarios de la siguiente línea en /etc/lpd.perms:
ACCEPT SERVICE=X REMOTEHOST=</etc/accepthost

3. Reinicie el demonio para que los cambios tengan efecto:
service lpd restart

Intercambiando Sistemas de Impresión

Para cambiar sistemas de impresión, ejecute la aplicación Conmutador del sistema de impresión. Iníciela seleccionando el Botón de menú principal (en el Panel) => Configuración del sistema => Más configuraciones del sistema => Conmutador del sistema de impresión, o escriba el comando redhat-switchprinter en la línea de comandos de la shell (por ejemplo, en un terminal XTerm o GNOME).

El programa detectará automáticamente si el sistema X Window se está ejecutando. Si es así, el programa arrancará en modo gráfico como se muestra en Figura 27-20. Si no se detecta X, arrancará en modo texto. Para forzarlo a que se ejecute en modo texto, use el comando redhat-switch-printer-nox.

Seleccione bien sea LPRng o el sistema de impresión CUPS. En RedHat Linux 9, CUPS es el sistema predeterminado. Si sólo tiene un sistema de impresión instalado, será la única opción mostrada.

Si selecciona OK para cambiar el sistema de impresión, el demonio de impresión seleccionado es activado para iniciarse en el momento de arranque y el demonio de impresión no seleccionado estará desactivado para que no se inicie en el momento de arranque. El demonio de impresión seleccionado es iniciado y el otro demonio es detenido; así los cambios tomarán efecto de inmediato.

Recursos Adicionales

Para saber un poco más sobre la impresión en GNU/Linux, puede consultar los recursos siguientes.

Documentación Instalada

man printcap-	La página del manual para el archivo de configuración /etc/printcap.
map lpr-	La página del manual del comando lpr que le permite imprimir archivos desde la línea de comandos.
man lpd-	La página del manual para el demonio de impresión LPRng.
man lprm-	La página del manual del utilitario para eliminar los trabajos de impresión desde la cola del spool LPRng.
man mpage-	La página del manual para el utilitario de línea de comandos para imprimir múltiples páginas en una hoja.
man cupsd-	La página del manual para el demonio de impresión CUPS.
man cupsd.conf-	La página del manual para el archivo de configuración del demonio de impresión CUPS.
man classes.conf-	La página del manual para el archivo de configuración de clases para CUPS.

Sitios Web Utiles

http://www.linuxprinting.org- Gran cantidad de información sobre la impresión en Linux.
http://www.cups.org/- FAQs y grupos de noticias sobre CUPS.

Ejercicio 8-1: Configurar y Usar Impresoras de Red

Solución se proveen en el Apéndice A.

En este ejercicio configuraremos un printer de red y compartiremos con varios sistemas. Un sistema con una impresora conectada directamente que estableceremos como un servidor de impresión. Los otros sistemas lo llamaremos clientes.

1.- Trabajando en el Printer Server, verifique que puedes enviar data a la impresora y que imprime.

2.- El Servidor de Impresión, configure el Printer, utilizando una utilidad grafica de configuración. Imprima una página de Prueba.

3.- En un cliente, agréguelo con una utilidad gráfica, al network printer que configuro en el dos. Imprima una página de prueba.

Ejercicio 8-2: La Cola/Queue de la Impresora

Solución se proveen en el Apéndice A.

1.- En su sistema deshabilite el printer queue/cola.

2.- Imprima 3 archivos diferente (recuerde el orden). Mire en los trabajos por imprimir con el comando lpc -status y lpq.

3.- Configure que el último trabajo es el próximo en imprimir.

4.- Cancele el trabajo del medio utilizando el comando lprm.

5.- Cancele todos los trabajos utilizando el comando lprm.

RESUMEN

En este capítulo, se introdujo lo siguiente:

- Usted fue introducido a los modelos de impresión de Linux
- El spooler, la Cola/queue utilizado por Linux
- Agrego impresoras gráficamente y bajo línea de comando
- Utilizo la línea de comandos para administrar y manejar impresoras
- Compartió impresoras con Windows, Novel y otros sistemas de impresión de UNiX.
- Trabajo con utilidades graficas de configuración de Impresoras.
- Fue introducido a los comandos lpq, lprm, lpc

PREGUNTAS POST - EXAMEN

Las respuestas a estas preguntas están en el Apéndice A

1.- ¿Cómo eliminaría usted los reportes de impresión desde el queue del (asumiendo que es único trabajo en el queue) administrador de la impresora laser?

2.- ¿Cómo investiga usted el estatus del laser printer admin?

3.- ¿Supongamos que el administrador de impresora laser es la impresora por defecto. Liste dos maneras diferentes de listar los trabajos en estera del queue?

REGISTRO DEL SISTEMA (SYSTEM LOGS)

OBJETIVOS

Al completar este Capítulo, usted podrá realizar las siguientes tareas:

- Configurar y utilizar los archivos del Registro para satisfacer necesidades de Administración y Seguridad.
- Detallar las funciones del estado y registro de mensajes del sistema y analice de funcionamiento.
- Estar completamente familiarizado con el funcionamiento, configuración y registros de los servicios para con la ayuda de estos registros poder gestionar troubleshooting básico.
- Identificar problemas básicos de las redes (networking) y las medidas a tomar para la localización de averías de áreas vulnerables.
- Dictar las medidas de emergencia tomadas cuando situaciones de vulnerabilidad ocurren.
- Describa Cuotas y sus conceptos y su implementación en usuarios y grupos.
- Familiarizarse con el sistema de archivos de Linux.
- Listar los diferentes niveles de almacenaje de RAID.
- Defina el rol del archivo fstab.
- Describa la función del fsck.

PREGUNTAS PRE-EXAMEN

Las repuestas se encuentran en el Apéndice A.

1.- ¿Cuál archivo se utiliza para determinar cual evento se almacena en cual registro

2.- ¿Cuál es el propósito de los archivos de Registro?

3.- ¿Qué se puede hacer con archivos de registro que se tornan muy grande?

4.- ¿Qué criterio se utiliza para determinar cual evento son registrados y dónde

5.- ¿Cuál archivo del registro puede ser considerado como el central que acapara la mayor parte de los mensajes syslog?

INTRODUCCIÓN

Un sistema típico de GNU/Linux esta en todo momento haciendo algo. Talvez actualizando una base de datos, enviando un correo, dirigir (rutear) un paquete, o cualquier otra cosa. Si algo sale mal, ¿Cómo pudiese el administrador tener idea de que sucedió cuando tantas cosas están pasando? ¿Si sabes cual proceso es responsable? ¿Podría haber una manera de conseguir mas información? Los Archivos de Registro son esta fuente de información. Estos archivos proveen información detallada del sistema. Si cualquier cosa significativa le ocurre al sistema, es casi seguro que podríamos encontrar algo referente a esto en los archivos de registro. Este capítulo te encaminara en rendirte una idea de cómo estos archivos de registros trabajan y como usarlos como aliados en tu quehaceres de administrador.

Archivos de Registros Comunes

Los archivos de registro de Linux son idénticos de distro a distro, esto es causa de los factores de tradición y protocolo de UniX, POSIX (Portable Operating System Interface for UniX). Claro existen pequeñas diferencias pero no nada que no se puede aprender o inclusive enfrentar desde el primer día en la nueva distro. La siguiente lista es una lista que debe ser común a todas las distros de Linux:

- /var/log/messages
- /var/log/secure
- dmesg
- lastlog
- Registros Dependientes de Procesos

El Archivo /var/log/messages

En el directorio /var/log, existe un archivo de nombre messages. Dentro de este archivo se encuentra una lista larga de los eventos (en orden cronológica), con cada entrada representando un registro individual. El archivo /var/log/messages es el archivo central del sistema de mensajes y registros del sistema. El almacena los mensajes de kernel con todos los programas que efectúan eventos que generan un registro. La mayor parte de los errores y mensajes del sistema se encuentran aquí en este archivo. Si no encuentras lo que buscas en el directorio /var/log/messages, o necesitas información mas especifica a cerca de un proceso en particular, busque archivos o subdirectorios con el nombre del programa en particular en el directorio /var/log.

El Archivo /var/log/secure

Otro de los archivos del registro de logs importante de GNU/Linux es el /var/log/secure (en otras distros no RedHat también puede estar con el nombre auth.log). El sistema en este archivo registra todos lo referente a usuarios que están accesando el sistema, como el usuario acceso y los posibles sucesos que pongan en peligro la seguridad del sistema. La mayor parte de esta información también esta disponible el archivo /var/log/messages. Por defecto todos los ingresos del root se registran en este archivo. Monitoreando estos archivos es una parte importantísima de la seguridad del sistema.

El Comando dmesg

Los mensajes del inicio del sistema del kernel son registrados en este archivo /var/log/dmesg. Este archivo contiene la información del subsistema del kernel y de las cargas de los modules en tiempo de encendido. A menudo, los drivers despliegan información de diagnostico de cada dispositivo y su modulo que el carga. Si un driver no se puede cargar por una mala configuración o problema del hardware, encontraras información en este archivo que Talvez te pueda orientar en tu búsqueda de fallas. Este archivo puede ser accesado directamente con un editor de texto o a través del comando dmesg.

Comando lastlog

Al invocar el comando lastlog, producirá una lista de los usuarios del sistema y la información de la ultima vez que ingresaron al sistema incluyendo de que maquina y a través de que puerto y la fecha. Si el usuario nunca ingresado al sistema también hará referencia a esto. El comando lastlog recibe su información desde un archivo de nombre /var/log/lastlog. Este archivo no es de formato leíble por humanos.

Registros Dependientes de Procesos

Algunos programas tienen sus propios directorios y pueden almacenar una gran cantidad de información si son configurados correctamente con este objetivo. Daemons, así como httpd, squid y samba, comúnmente mantienen archivos y directorios separados detallando sus eventos específicos a las tareas que ellos desempeñan.

Daemons de Registros

GNU/Linux tiene varios métodos de archivar información en los archivos de registros. Algunos programas simplemente le escriben a sus propios archivo. La mayoría de los programas utilizan una Interfase de Programación de Aplicación (API) la cual es proveída por el kernel llamada syslog. Un daemon llamado syslogd acepta mensajes desde el kernel y decide donde almacenarlo. El kernel mismo tiene su propio mecanismo de registros. El daemon klogd acepta estos mensajes desde el kernel y se los envía al syslogd.

En esta sección, cubriremos los siguientes temas:

- Syslogd
- Klogd

El Estándar Syslogd

En GNU/Linux, un estándar llamado syslog dicta como los programas envían mensajes a través del sistema. Las librerías del sistema que los programas utilizan contienen funciones syslog. Acceso a estas funciones le permite a programas producir los mensajes que pueden ser manejados por el sistema. Un daemon llamado syslogd escucha en el background por estos mensajes, les da formato y entonces los envía a los lugares pre-definidos en los archivos de configuración en /etc/syslog.conf.

Configurar syslog

Syslog se ejecuta por defecto en todas las distribuciones reconocidas de Linux. Sus valores de configuración por defecto deben servir sus necesidades. Por si acaso debes cambiarla o se desconfigura entonces, vamos a conoceré lagunas de sus características.

Existen tres partes en una entrada típica del archivo Syslog con:

- Facilidad
- Prioridad
- Acción

La siguiente sintaxis es utilizada en el comportamiento de syslogd:

mail .* - /var/log/mail

Los primeros dos campos están a las izquierda. Ellos consisten en facilidad y prioridad, separadas por un punto. Ambas son palabras claves utilizadas internamente por el sistema de mensajería. En este ejemplo, la facilidad es correo (mail), con una prioridad siendo representada por un comodín, en este caso el asterisco (*). Este ejemplo ejecuta lo siguiente, envía todos (*) los mensajes del sistema a la localidad (el correo, mail)

/var/log/mail. El campo de Localidad es también conocido como acción.

Además de estas tres partes del syslog.conf discutiremos ingresos remotos.

Facilidad

Las facilidades disponibles son:
cron, daemon, kern, lpr, local, mail, news, priv, syslog, user, uucp

Algunas de estas deben serle familiares ya que ellas reflejan procesos comunes del sistema. El proceso que deseas hacerles registros caerá en unas de estas categorías. El programa que crea el evento y necesita registrarlo decide bajo que categoría hacerlo. Elija una de la facilidad específica o un comodín.

Prioridad

Las prioridades de registros disponibles son (ordenadas por severidad)
debug, info, notice, warning, err, crit, alert, emerg, panic

Ellas determinan la severidad del mensaje. Elija una para registrar prioridades específicas o un comodín.

Acción

El campo de acción determina donde se coloca el registro.

- Una ruta y un nombre de archivo en el campo de acción crea un registro en la máquina local.
- Una @ en el campo acción indica que el registro será enviado a otro sistema a través de la red. El registro será enviado a la dirección que inmediatamente sigue la @. (Ej. @192.168.5.24).

Si hacemos unos pequeños cambios en el archivo Syslog.conf puede direccional los registros otro sistema en vez de enviarlo a un archivo. En la próxima sección daremos un vistazo a como habilitar el sistema de login remotos.

Login Remoto

El syslogd contiene funciones que le permiten pasar los registros por la red entre sistemas que ejecutan syslogd. Si usted es responsable de varios sistemas, las capacidades de login remota pueden ahorrase mucho tiempo. Hay algunos pasos para la creación de registración remota:

1.- Cerciorarse de que el syslog esta definido en /etc/services para todos los sistemas.
2.- Configure syslog en cada máquina a comenzar a habilitar (registración) login remoto.
3.- Edite el archivo /etc/syslog.conf en las máquinas que exportaran sus registros.

Mire en /etc/services. Para que syslogd comience a permitir login remoto, debe haber una entrada parecida a la siguiente:
syslog 514/udp

Después, usted necesitará tener que el syslogd arranque con logging remoto habilitado. Usted necesitará asegurarse que el programa del syslogd se invoque con la opción de -r:
/usr/sbin/syslogd -r

Algunas distribuciones ya se inicia con la opción -r, pero en algunas usted tendrá que agregarla manualmente en los scripts de arranque de algunas distribuciones.

Finalmente, debes modificar el archivo /etc/syslogd.conf. Para que los registros se envíen a otra máqui-

na, se necesita agregar una entrada que le dice a syslog dónde enviarlos. Lo que sigue es un ejemplo de esta entrada en el archivo /etc/syslog.conf de un sistema que se ha configurado para enviar los registros de todos sus eventos a otra máquina:

***.* @una.maquina.com**

Esta entrada transmitirá todos los mensajes del syslog a una.maquina.com. En el sistema de recepción, las entradas del registro se mostrarán en /var/log/messages con la fecha, la hora, y el nombre del sistema que lo envió.

Una de las aplicaciones principales de remote logging es que permite que usted instale un servidor dedicado que reciba todos los registros de todo el sistema de red. Esto proporciona una medida agregada de seguridad porque los intrusos no podrán cubrir sus pistas a menos que puedan comprometer el servidor de registro también. Un método aún más seguro es tener que todos los registros se impriman por una impresora.

klogd

GNU/Linux tiene una utilidad llamada klogd, con la única función de escuchar los mensajes producidos por el kernel. Los mensajes del Kernel son procesados dando los siguientes pasos:

1.- Una parte del Kernel hace una llamada para crear una entrada al registro del sistema.
2.- El klogd daemon recibe el mensaje desde /pro/kmsg donde el kernel lo ha hecho disponible a los programas externos.
3.- El campo de prioridad del mensaje se convierte del formato de mensajes del kernel (un dígito del 0 al 7) para hacerlo compatible con el del syslog.
4.- EL mensaje es enviado a syslogd donde es reconocido como un registro del sistema y procesado de tal manera.

ADMINISTRACION DE LOS ARCHIVOS DE REGISTRO

Los archivos de registro fueron diseñados para almacenar información del sistema en tiempo de ejecución. Estos archivos son generados para proveerle información necesaria. Si ustedes nunca lee esta información entonces no tienen sentido.

Al analizar el registro, es importante saber quién se encuentra en ellos. Es de buena costumbre periódicamente analizar los archivos para así tener presente como lucen los registros cuando el sistema está normal. De esta manera, estarás más preparado cuando quiere localizar un problema.

Los siguientes tópicos son discutidos en esta sección:

• logger • lograte • xconsole

Comando logger

El programa logger ese es una simple herramienta que permite que puedas colocar mensajes estampados con la fecha en tus archivo /var/lo/messages. Es todo puede ser muy cursi y cuando quiere hacer un cambio fijo está seguro a trabajar. Y más adelante si ocurre un problema entonces podrá tener una referencia al porque dejo de funcionar. Utilizar logger es sencillo:

$ logger Instale kbasic obviando dependencias… ha ver que pasa.:)

Después al revisar el log veras el registro de esta manera:

Jul. 27 14:22:45 gnome2 gnome: Instale kbasic obviando dependencias… ha ver que pasa. :)

El logger es muy facil de colocarle estampados de fecha a cambios ejecutados en caso de que después en

el futuro causen problemas.

La Herramienta lograte

Los registros siempre están recogiendo información al menos que el sistema no este en completo reposo sin servicios en uso o activos. El archivo /var/log/messages es donde la mayoria de los mensajes de sistema se registran. Este archivo puede crecer rápidamente. Si se dejase solo, continuaría creciendo de tamaño hasta llegar a ser imposible para manejar. Muchos archivos del registro crecen excesivamente grandes si no se administran, así que es bueno tener herramientas como logrotate. El logrotate hace un trabajo excelente de mantener cosas en orden. El programa logrotate es una herramienta flexible que hace muchas cosas. Si las fechas de los registros caducan, pueden ser suprimidas automáticamente después de permanecer cierto tiempo en el sistema, si se ponen demasiado grande, puedan ser comprimidos. Si el sistema esta en una localización remota, los archivos de registro se pueden enviar a un administrador por e-mail.

Usted puede modificar el comportamiento logrotate para requisitos particulares editando el archivo de logrotate.conf en el directorio /etc. Lo que sigue es una entrada de ejemplo de /etc/logrotate.conf:

```
#Global Settings
compress
size=65k
#settings individual
"/var/Iog/wu-ftpd/access.log" {
mail admin@abiertos.org
size=150k
nocompress
endscript }
```

Las primeras entradas en /etc/logrotate.conf son definiciones de variables globales. Todos los registros las utilizarán, salvo que se especifique lo contrario en una entrada separada. La segunda entrada es para un archivo de registro específico, en este caso, archivo /var/log/wu-tpd/access.log. Las acciones que logrotate tomará son las siguientes:

- size — Logrotate procesara los registros una vez adquieran el tamaño definido.
- compress — Registros son comprimidos durante su proceso.
- nocompress — Esta opción no comprimirá durante el proceso.
- mail — Los archivos de registro serán enviados a este correo una vez procesados.

En este ejemplo del archivo logfile.conf define las acciones específicas para un archivo, Access.log. La definición comienza con la localización del archivo del registro (o del directorio por completo de los archivos del registro) entre las comillas dobles, seguidas por una llave abierta. La definición termina con la palabra clave endscript y entonces una llave de cierre. Toda entre estas llaves es la accion específica a este archivo de registro. Los cambios en las definiciones globales dan efecto par que el archivo Access.log no se le efectue ninguna compresión y los registros se enviarán a admin@codigolibre.org.

Algunos registros son más importantes que otros dependiendo de qué función desempeñan en el sistema, es decir, web server, servidor de archivo, o de acceso multiusos de shell. El poder manejar archivos de registro sobre una base individual es útil cuando hay cosas específicas que usted necesita vigilar.

Salida de mensajes del sistema a la consola generalmente significa que ha ocurrido acontecimientos importantes, a menudo eventos criticos del sistema. Normalmente, esto es adecuado para las operaciones de sistemas, pero si se encuentra trabajando en una seccion en el X, los mensajes pasaran desapercibidos por el usuario. Aquí es adonde la utilidad xconsole entra en juego. El programa xconsole proporciona una ventana a estas salidas. El archivo asociado a la consola es /dev/console. Para que esta consola funcione correctamen-

te en el X, este archivo de dispositivo especial debe ser leible por el usuario actual y el uso del comando chown puede ser necesario. Normalmente, el xdm y otros programas gráficos que manejan la conexión cambian la propiedad de la consola de sistema. Además, un terminal que es capaz de hacer salida de los mensajes de consola puede ser logrado invocando el siguiente comando:

 # xterm -C

La consola de sistema es una parte importante del sistema para mantener su integridad, porque actúa como un portavoz para las emergencias del sistema. Es altamente recomendado que usted utilice el xconsole o una consola virtual dedicada para estos mensajes de emergencia.

Ejercicio 9-1: Encontrar y Accesar Archivos de Registro

Durante este ejercicio, usted localizara algunos de los archivos de registro del sistema y correctamente los analizará. El directorio más importante para observar cuando hace uso los archivos de registro es /var/log. Este directorio es típicamente el destinario para todos los registros del sistema, aunque la utilidad syslogd permite el cambio de dirección a cualquier directorio. Tener acceso a los archivos de registro es muy importante para cualquier administrador. No hay soluciones proporcionadas para este ejercicio.

1. Ingrese a su máquina como root. Si usted esta usando el X, abra una ventana terminal y ejecute el siguiente comando:
   # cd /var/log

   Esto le colocará en el directorio con sus archivos de registro. Ejecute ls -la para listar los archivos en este directorio. Éstos son los registros de los demonios del syslogd y klogd.

2. ¡Búsque el destino de los mensajes del kernel en el archivo de configuración del syslogd ejecutando el siguiente comando:
   # grep kern /etc/syslog.conf

   Esto busca en el archivo por la secuencia de caracteres “kern” y despliega los resultados en pantalla. Usted debe recibir una salida similar a la siguiente:
   # Registre todos los mensajes del kernel a la consola.
   kern.* /var/log/kernmsg

3. Viendo cómo este archivo de registro puede crecer muy grande, usted solo desea analizar las últimas 50 líneas de él. El carácter pipe (|) se utiliza en este ejercicio para pasar la salida de un programa a otro y luego analizar esta salida. Deseamos poder leer las 50 últimas líneas del registro, incluyendo las que normalmente se pasarían del límite de la consola. Esto se puede lograr ejecutando el siguiente comando:
   # tail -50 kernmsg | less

4. Las entradas en un archivo de registro estándar se escriben así:
   date hostname facility: message

   Si por ejemplo usted hallase la siguiente entrada:

   Jul 26 18:52:21 vfs kernel: PCI latency timer (CFLT) is unreasonably low at32. Setting to 64 clock y usted desea darle salida solamente a las cinco ultimas líneas que contienen la secuencia del PCI del registro del mensaje del kernel, usted puede combinar la funcionalidad del grep con la de la tail y ejecutar lo siguiente:
   # grep PCI kernmesg | tail -5

5. Utilice el comando head si usted desea ver las primeras diez líneas del archivo de diario anterior.

   Ejecute el siguiente comando:
   # head -10 kernmsg

6. ¿Cómo puede usted ver un registro mientras se esta actualizado sin tener que volver a ejecutar éstos comandos cada vez? Hay realmente dos maneras de hacer esto. La primera requiere que incorpore una línea como el siguiente:
   ***.* /dev/tty12**

   en el archivo /etc/syslog.conf y ejecutar un killall -HUP syslogd para hacer que el demonio de syslogd vuelva a leer su archivo de configuración. Esto envía toda la salidas del registro a la doceava terminal virtual, así que cualquier momento que desea ver en la pantalla que pasa en los registros, solo tienes que presionar ALT+F12 para cambiar al terminal virtual y hay estará la información.

   La segunda manera de ver como se escribe el registro es utilizar el comando tail con el parámetro especial que te permite seguir su progreso. El comando es:
   # tail -f /var/1og/messages

   no devolverá el prompt después de que haya ejecutado (como es lo normal de otros comandos y el de tail). Continuará actualizándose con cada línea que se le escriba al archivo de registro. Esto es una técnica muy útil para eliminar errores (debuggin) o la administración general del sistema.

RESUMEN

En este capítulo, se introdujo lo siguiente:

- Registros del Sistema (System Logs)
- Archivos Comunes del Registro del Sistema
- Utilidad de logrotate
- Descripción el sistema de mensajes de Linux
- Configurando el Syslog
- Registro Remoto
- Los Programas logger y xconsole

PREGUNTAS POST - EXAMEN

Las respuestas a estas preguntas están en el Apéndice A

1.- ¿Qué tipo de archivos usted investigaría para encontrar ciertos problemas de proceso?

2.- ¿Si un administrador deseara saber quien esta en este momento utilizando el sistema, a que archivo del sistema se debe dirigir el?

3.- ¿Cuál archivo del registro nos da una buena indicación del bien estar de nuestro sistema?

4.- ¿Es buena o mala práctica regularmente revisar los archivos del registro?

209

APENDICE A

ADMINISTRACIÓN DEL SISTEMA OPERATIVO GNU/LINUX

CAPÍTULO 1

PREGUNTAS PRE-EXAMEN

1.- ¿Qué tipo de acceso le permite el comando sudo a los usuarios?

®.- Permite el acceso a cualquier programa (comando) o grupo de programas.

2.- ¿Cuál argumento de opción se le pasa al comando uname para ver el tipo de procesador?

®.- uname -r

EJERCICIO 1-1: Navegar y Usar el Shell del Administrador

No se provee repuesta para este ejercicio.

1.- Ingrese como root y ejecute algunas de las aplicaciones y utilidades gráficas orientadas a la administración de GNU/Linux. Si estas en SuSE (YaST) o una distro de las anteriores, ejemplo RedHat 7.1 o Mandrake 8.1, Debian 2.2 podrías experimentar con Linuxconf. Este tipo de aplicaciones se utilizo mucho en GNU/Linux pero con el tiempo fueron abandonadas ya que están 100% en contra de las filosofías UniX de no concentrar demasiado poder detrás de una sola aplicación. Ahora solo SuSE mantiene el YaST. En RedHat la funcionalidad existe pero las utilidades son individuales.

PREGUNTAS POST - EXAMEN

1.- ¿Por qué no debe usted permanecer en la cuenta root todo el tiempo?

®.- Por razones de seguridad, el administrador del sistema nunca deberia dejar una terminal desantedida logueada como root.

2.- ¿Cómo pueden un usuario comunicarse con otro usuario?

®.- El comando talk permite a 2 usuarios comunicarse en tiempo real.

3.- ¿Cuándo deben las tareas administrativas ser denegada? ¿Quién deben ser delegadas.?

®.- El mantenimiento del hardware deberia ser realizado por un ingeniero o tecnico calificado. Otro tipos de mantenimientos, tales como agregar una impresora o cambiar cartuchos de toner, pueden ser delegadas a otro personal administrativo.

CAPITULO 2

PREGUNTAS PRE-EXAMEN

1.- ¿Cómo puedes determinar si un Kernel es Estable o de Desarrollo?

®.- Dado un numero de kernel A.B.C. Si B es par entonces es un kernel estable; Si B es impar entonces es una version de kernel en desarrollo.

2.- ¿Cuáles son los tres métodos disponibles para configurar el Kernel antes de Compilarlo?

®.– make config, make xconfig, make menuconfig.

3.- ¿Qué es un módulo del Kernel?

®.– Un modulo del kernel es un componente del kernel (tal como un driver de dispositivo) que puede ser agregado al kernel mientras esta corriendo.

4.- ¿Por cuáles razones compilarías un Kernel?

®.– Para agregar nuevas caracteristicas o corregir bugs actualizandolo a una nueva version que no esta todavia lista para distribucion, agregar o configurar dispositvos que no esta en la distribucion estándar, o modificarlo para fijarle un tipo de procesador especifico.

5.- ¿Cuáles son los objetivos del Kernel?

®.– El kernel controla el acceso a los recursos del hardware y los hace disponibles a programas. Este provee las funciones basicas del sistema operativo.

EJERCICIO 2-1. Reconstruir el Kernel

Este ejercicio lo guiara a través del proceso de la compilación e instalación del kernel. Aquí se configurara el kernel para añadirle el soporte de dispositivos Joystick. Este ejercicio requiere un espacio en disco duro de 100mg. y al rededor de 2h. No damos solución para este ejercicio, asegurase que usted cuente con un disquete de boteo, en caso de que su kernel no arranque. Primero asegurase que su dique inicie su máquina, este ejercicio asume que usted ha descargado la version 2.4.2-15 y le va a aplicar un parche para actualizar a la version 2.4.20. Usted puede usar cualquier version que deseé. Si usted desea practicar con el código fuente de su sistema, usted puede obviar la descarga y el desempaquetamiento del kernel. Utilizar el código fuente del kernel de su distribución es una buena idea; ya que usted podrá estar mas seguro de las características con las que la cuenta:

1.- Descargue el kernel , lo puede obtener de http://www.kernel.org

2.- Identifique su kernel actual, cual esta almacenado /boot. Común mente nombrado vmlinuz y tiene la version agregada Ej.: vmlinuz-2.4.2-15

3.- Efectué una copia de resguardo de su kernel. Asumimos que su imagen de kernel es de versión 2.4.2-15, usted utilizaria el siguiente comando:
 # cp vmlinuz vmlinuz-2.4.2-15-bueno

4.- Cree una entrada en su /etc/lilo.conf para que usted pueda iniciar con su kernel ya existente en caso de

que el nuevo no funcione:

image=/boot/vmlinuz-2.4.2-15-bueno
label=Funciona
root=/dev/hda2
read-only

5.- Si ya usted tiene el código fuente del kernel en /usr/src/linux , renombre el directorio

 # mv /usr/src/linux /usr/src/linux-bueno

6.- Desempaquete el código fuente del nuevo kernel

 # cd /usr/src/
 # tar zxvf linux.2.4.19.tar.gz

Si descargo bz2Image, ejecute:

 # tar xvfy o tar xfl

También puede usar el comando:

 # bzcat linux.2.4.-19.tar.bz2 | tar xvf -

7.- Si descargo algún parcho que desea aplicar:

 # zcat patch 2.4.19.gz | patch -pO -N -E -s

Reemplace el zcat con bzcat si descargo el patch.bz2

8.- Cambie al directorio del kernel recién creado:

 # cd linux

9.- Prepare el árbol del código:

 # make mrproper

10.- Configure el kernel:

 # make menuconfig ó make xconfig ó make config

11.- Navegue a través de las categorías seleccionando la categoría INPUT CORE SUPORT y presione ENTER; presione Y para habilitar INPUT CORE SUPORT

12.- Un listado de dispositivos de entrada aparecerá elija Joystick Support y presione Y para habilitarla selecciones EXIT, y EXIT nuevamente.

13.- Navegue hacia abajo y selecciones CHARACTER DEVICES, y luego seleccione Joystick.

14.- Habilite Joystick Support al presionar Y o use la barra espaciadora para selecionar YES/NO/MODULE.

15.- Una vez Joystick Support esta habilitado aparecerán más opciones específicas a dispositivos de Joystick.

Elija Classic PC analog joysticks and gamepads.

16.- Salga de la utilidad menuconfig y guarde su configuración.

17.- Construya el kernel:

 # make dep
 # make clean
 # make bzImage

18.- Instale el kernel en lugar del anterior

 # cp arch/i386/boot/bzImage /boot/vmlinuz

19.- Ejecute lilo para decir al cargador de arranque que el kernel esta en el disco

 # /sbin/lilo

20.- Construya los módulos
 # make modules

21.- Instale los módulos
 # make modules_install

22.- Reinicie el sistema
 # /sbin/shutdown -r now

23.- Si todo ha funcionado bien usted podrá reiniciar el sistema y verificar la version de su nuevo kernel:
 # uname -a
Linux localhost 2.4.19 #1 SMP Mar Jun 15 13:15:55 EST 2003 i386

24.- Al menos que usted no este totalmente confortable con el kernel experimental de este ejercicio, restablezca las copias de seguridad.

EJERCICIO 2-2. Restaurar al Kernel Anterior en Caso de Fracaso

No se proveen soluciones a este ejercicio.

Nota: Este ejercicio no será posible si usted no hizo una copia de seguridad en el ejercicio anterior

1.- Reinicie el sistema e intercéptelo al momento de inicio

Cuando el prompt le presente Lilo: Presione la tecla ctrl. (Control) y luego presione la tecla tab para ver las posibilidades de boteo que usted tiene disponibles. Dependiendo las características puestas en el /etc/lilo.conf usted obtendrá el mensaje de Linux y linux-bueno; entonces usted digitará linux-bueno para accesar a la version anterior del kernel viejo.
lilo: <TAB>
linux linux-bueno
lilo: linux-bueno

2.- Inicie seccion como root y restablezca su kernel anterior y reinicie. Asumiendo que la imagen de su kernel es llamada vmlinuz-2.4.2-15-bueno, usted ejecutara lo siguiente:
 # cp -a /boot/vmlinuz-2.4.2-15-bueno vmlinuz

3.- Ejecuto el comando lilo:
 # /sbin/lilo

4.- Reinicie el equipo.

PREGUNTAS POST-EXAMEN

1.- Liste y describa por lo menos 2 diferentes tipos de código fuente encontrado en el árbol del kernel

®.– Un tipo es el set de archivos core, los cuales deben ser incluidos en cada compilacion del kernel. Otro tipo es el set de archivos dependientes de arquitectura, los cuales incluyen archivos especificos a la arquitectura de la maquina para la cual el kernel esta siendo compilado.

2.- Describa un método de optimizar el kernel

®.– Compilar el kernel con instrucciones especificas del procesador.

3.- ¿Cuando se pueda el caso de que un administrador utilice módulos en vez de compilar el código al kernel?

®.– Utilizar los módulos disminuirá el tamaño del kernel y permitirá que el hardware sea cambiado sin requerir la recompilación del kernel.

4.- ¿Cuál es la diferencia entre un zImage y un bzImage?

®.– zImage debe ser utilizado en arquitecturas diferentes a i386 y tambien limitadas a 1 MB de descompresion. bzImage no tiene limite de tamano de archivo.

5.- ¿Cuando es apropiado recompilar el kernel? ¿Cuales medidas de seguridad deben ser tomadas cuando se reinicia una maquina utilizando un nuevo kernel?

®.- Es apropiado recompilar el kernel para agregar nuevas caracteristicas, eliminar bugs, o agregar soporte para nuevo hardware.

CAPITULO 3

PREGUNTAS PRE-EXAMEN

1.- ¿Qué es un paquete?

®.– Un paquete es una coleccion de archivos combinados dentro de un simple archivo. Puede ser instalado sobre un sistema para proporcionar uno o mas programas junto con todos los archivos requeridos para la operacion de esos programas. Los paquetes tipicamente contienen dependencias haciendo referencia a otros paquetes que deben ser instalados en orden para que los programas trabajen.

2.- ¿Nombre algunos de los paquetes más comúnes?

®.– RPM, DEB y archivos tar.gz

3.- ¿Qué es una librería compartida?

®.– Una libreria compartida es un archivo conteniendo codigo de programas utilizados comunmente por varios programas. Este solamente necesita ser cargado en memoria en una ubicacion, y todos los programas utilizaran la misma ubicacion para accesar al codigo.

4.- ¿Si usted descargo un paquete Source como lo instalará?

®.– Descomprime, configura, compila e instala con los siguientes comandos:

```
tar -xfz paquete-x.y.z.tar.gz
cd paquete-x.y.z
./configure
make
make install
```

EJERCICIO 3-1: El Uso Básico de los RPM

En este ejercicio usted se familiarizara con el rpm y su utilidad poderosa para instalar paquetes rpm en un sistema linux. Instalar paquetes rpm es una tarea simple si el sistema soporta el uso de la utilidad rpm, instalaremos un paquete rpm llamado mpg123.4-24.i386.rpm este paquete sera descargado de ftp.redhat.com.

1.- Obtener paquete

®.- El paquete por ejemplo digamos que esta en ftp://ftp.rpm.org/mpg123.4-24.i386.rpm y en http://rpm.org/mpg123.4-24.i386.rpm

2.- Utilizando un web browser
 # Lynx http://rpm.org/mpg123.4-24.i386.rpm

3.- Utilizando un ftp grafico

®.- gftp y colocar el server como ftp.rpm.org y navegar al paquete en el directorio raiz.

4.- Utilizando wget

®.- # wget -c http://rpm.org/mpg123.4-24.i386.rpm

5.- Navegue al directorio donde lo descargo

®.- # cd /ruta/al/directorio/que/lo/descargó

6.- Simplemente invoque el comando rpm e instale el paquete añadiendo las siguientes opciones: explicito (verbose) y que despliegue las barras de progreso.

®.- # rpm -Uvh mpg123.4-24.i386.rpm

EJERCICIO 3-2: Verificar la Instalación de un Paquete

1.- Utilizando rpm, identifique la version de un paquete instalado.

®.- # rpm -q gedit (cualquier otro paquete)

2.- Utilizando rpm, identifique todos los paquetes instalados.

®.- # rpm -qa

3.- Utilizando rpm, despliegue la información de un paquete instalado.

®.- # rpm -qi gedit

4.- Verifique todos los paquetes instalados utilizando la opción verify.

®.- # rpm -Va

EJERCICIO 3-3: Verificar Ubicación de la Base de Datos

1.- Verifique la ubicación de la base de datos RPM del paquete perl utilizando el comando rpm y la combinación verify.

®.- # rpm -Vvv perl

EJERCICIO 3-4: dpkg/dselect

dpkg

1.- Instale el paquete de Debian synaptic-0.7.deb

®.– # dpkg -i synaptic-0.7.deb o # dpkg --install synaptic-0.7.deb

2.- Remueva el paquete synaptic-0.7.deb Asegurase de remover todos los archivos que el paquete instalo incluyendo los archivos de configuración. *** Existe un comando para esta acción.

®.- # dpkg -r synaptic-0.7.deb

3.- Cuestione la base de datos listando los paquetes instalados por nombre

®.– # dpkg –list patron-busqueda

dselect

4.- Abra la utilidad

®.- Como el usuario root corra dselect desde la linea de comandos.

5.- Seleccione la opción que le permite elegir el método de acceso y elija la que le permita utilizar los CDROM

®.- Seleccione el 0. La opcion [A]ccess desde el menu dselect y seleccione el cd como media

6.- Actualice la lista de posibles paquetes

®.- Elige la opcion 1.[U]pdate desde el menu principal dselect. Desde aqui estara presentado con opciones como desde donde actualizar los paquetes disponibles

7.- Instale el paquete disponible

®.- Seleccion la opcion 5.[I]nstall desde el menu principal dselect y la utilidad automaticamente instalara los paquetes seleccionados

PREGUNTAS POST- EXAMEN

1.- ¿Qué hace que un paquete sea diferente de un tarball?

®.- Aunque un paquete podria contener un numero de archivos, tambien posee instalacion, desinstalacion,

configuracion y detalles de dependencias que seran leidas por el administrador de paquetes para realizar operaciones extendidas.

2.- ¿Qué es una dependencia y que pasa si usted continua al instalar un paquete con una que no esta resuelta?

®.- Una dependencia es un paquete que es requerido por otro para funcionar apropiadamente. Si alguien intenta instalar un paquete sin resolver la dependencia, la instalacion terminara y el paquete podria no ser instalado hasta que las dependencias sean resueltas. Si la instalacion es forzada, el paquete podria no funcionar.

3.- ¿Usted ha encontrado un binario no familiar y deseas investigar a que paquete pertenece. ¿Cómo puede investigar esto con dpkg y rpm?

®.- # rpm -qf <Nombre> y # dpkg -S <Nombre> respectivamente.

4.- ¿Qué es un paquete fuente y por que crearías un paquete fuente en vez de un paquete binario?

®.- Un paquete fuente es aquel que contiene el codigo fuente y las instrucciones para pasarlas al administrador de paquetes en orden para ser compiladas e instaladas en forma binaria. Una licencia prohibitiva de distribucion es una razon por la cual deberia ser creado un paquete fuente.

5.- Acabas de compilar e instalar una librería en la fuente y encuentras que las aplicaciones que la requieren aún no funcionan. ¿Cómo verificarías que las opciones la pudieron localizar y como vieras todas las librerías que ellas requieren?

®.- Necesitarias correr el comando ldconfig para reconstruir la base de datos de las librerias compartidas del sistema para verificar que las nuevas librerias pueden ser localizadas por la nueva aplicacion. El comando ldd <filename> te dice que librerias compartidas requiere el archivo binario.

CAPITULO 4

PREGUNTAS PRE-EXAMEN

1.- ¿Cómo se puede prevenir la creación del archivo CORE cuando un programa aborta inadvertidamente?

®.- # ulimit -c 0

2.- ¿Cómo puedes saber cuanto espacio físico y de swap esta en uso en la memoria del sistema?

®.- Utilizando el comando free o ejecutando # cat /proc/meminfo.

3.- ¿Define el término Daemon?

®.- Un demonio es un programa que corre en background y proporciona un servicio al sistema, este espera que algo suceda para poder responder.

4.- ¿Qué es buffer overflow?

®.- Los demonios usualmente corren como root o como usuario privilegiado. Si el programa esta trancado sobre un desbordamiento interno en el buffer, este podria sobreescribir partes del programa, ocasionando que el programa corra erradamente. En algunos casos, el programa puede trancarse en archivos especificos de escritura, y el demonio estará hábil para escribir en algun archivo que el propietario del demonio pueda.

EJERCICIO 4-1: Procesos

Ejecute el comando ps varias veces y note como el PID se va aumentado. Despues corra este comando varias veces:

$ ps aux | grep ps

Note que la mayoria de las veces verá ambos procesos el de ps y el de grep, pero puede pasar dependiendo la velocidad de su maquina que solo vea el de ps porque el de grep no tubo tiempo para entrar en la tabla de los procesos.

No se da solución a este ejercicio.

EJERCICIO 4-2: Modificar Valores en /proc

Para este ejercicio no se dan soluciones.

El sistema de archivos /proc es una herramienta excelente para monitorear el sistema directamente. En este directorio /proc existen directorios de nombre numéricos. Estos nombres se refieren a los PIDs de los procesos en ejecución, y los archivos dentro de estos directorios contienen información correspondiente al proceso. Los archivos con nombres no numéricos corresponden a procesos del sistema y por ende con información pertinente al sistema mismo.

Los archivos representando procesos no tienen permiso de Escritura (Write). Comúnmente solo los archivos que contienen información de configuración pueden ser alterados. Recuerde que esos archivos son utilizados y creados directamente por el kernel y modificarlo puede causar comportamiento impredecible.

El directorio /proa/sys en particular contiene información de la configuración del sistema. Por esto información del sistema puede ser directamente alterada a través de estos archivos. Vamos a dar un pequeño ejemplo de una situación que se puede dar en un servidor de páginas web, que en la compilación de su kernel se dio un número de handles que directamente afecta el número de archivos que puede abrir. Bueno este limite se esta viendo amenazado por el volumen alto de visitantes a las paginas web que esta sirviendo.

Ahora la pregunta es ¿Qué podemos hacer? La respuesta es que podemos modificar el archivo

/proc/sys/fs/file-max el cual nos dice por ejemplo:

```
[root@www /]# cat /proc/sys/fs/file-max
4096
[root@www /]# echo Nuevo _ valor /proc/sys/fs/file-max
```

1.- Determine en su equipo cual es el valor actual y triplíquelo. Luego desvuélvalo a su estado original.

Ejercicio 4-3: Encendido y Apagado del Sistema

En este ejercicio nosotros podríamos reconfigurar el mecanismo de encendido y apagado de linux.

1.- ¿Cuál es el runlevel default de sistemas y como lo determinas?

®.– # grep inittdefault /etc/inittab; # /sbin/runlevel.

®.- No hay una lista simple de servicios parados e iniciados cuando un sistema es reiniciado; ellos cambiaran de sistema a sistema.

2.- Cree un simple script de inicio entrando los siguientes comandos:

```
# cat >/etc/rc.d/init.d/tele
case "$1" in
start) echo Starting telepathic services;;
stop) echo Stopping telepathic services;;
esac
^D
```

Luego ejecutar estos comandos:

```
# ln -s /etc/rc.d/init.d/tele /et/rc.d/rc2.d/S01tele
# ln -s /etc/rc.d/init.d/tele /et/rc.d/rc2.d/K01tele
# ln -s /etc/rc.d/init.d/tele /et/rc.d/rc0.d/K01tele
```

Ahora pruebe sus cambios reiniciando su computador y observando los mensajes en tu pantalla.

Ejercicio 4-4: Cambiando de runlevel (nivel de ejecución)

1.- Cambie su default runlevel al nivel 4. Habilita los niveles 2 y 3 dentro del 4 y agrega los requerimientos para configurarla /etc/rc.d/rc4.d directorio.

®.– Copie el contenido de /etc/rc.d/rc2.d y rc3.d a /etc/rc.d/rc4.d

```
# cp /etc/rc.d/rc2.d/* /etc/rc.d/rc3.d/* /etc/rc.d/rc4.d/
```

®.- Edite /etc/inittab para configurar la liena initdefault a runlevel 4 y reinicie.

PREGUNTAS POST- EXAMEN

Si usted tiene tres programas debe cerrar. Al usar el comando ps ux muestra lo siguiente:

```
[root@www /root]# ps ux
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.2	1424	548	?	S	09:21	0:04	init [7]
root	2	0.0	0.0	0	0	?	T	09:21	0:00	[kflushd]
root	3	0.0	0.0	0	0	?	S	09:21	0:00	[kupdate]
root	4	0.0	0.0	0	0	?	Z	09:21	0:00	[kswapd]

¿Cuál es el PID de cada uno de estos procesos?

®.- Los pids son del 1 al 4

¿Cuál es el estado de cada uno de estos procesos?

®.- El estado de los procesos son las letras debajo de STAT en este caso es S -es Sleep o Durmiendo, T - es Stopped Detenido, Z -es Zombie

2.- Usted ha hecho un cambio en el archivo de configuración de un daemon llamado xyzd. Sin embargo, el daemon parece no reconocer los cambios. ¿Cómo podemos hacer que el daemon relea el archivo config?

®.- # killall -hup xyzd (podrias correr tambien ps para encontrar el PID del demonio xyzd y hacerle killall -hup a PID.).

3.- Usted esta corriendo un comando make que da inicio y ejecuta un comando hijo make, y este desestabilizando el CPU (debido a que la compilación tiende hacer uso intensivo del CPU). ¿Cómo puede detenerse este proceso?

®.- # killall -STOP make

4.- ¿Cómo puede usted mostrar cuales programas han sido ejecutados más veces desde la última vez que se reinicio el sistema?

®.- # sa -n (El primer comando listado sera el unico que corra mas,

5.- Ivelis empezó un proceso con el comando nice y le dió una prioridad inicial menor que la del sistema. Después ella se da cuenta que el proceso tiene más urgencia que lo que pensó. ¿Cómo puede Ivelis alterar la prioridad del proceso?

®.- # renice 0 <pid>

CAPITULO 5

PREGUNTAS PRE-EXAMEN

1.- ¿Liste algunos de los sistemas de archivos disponibles en Linux?

®.- ext2fs, ext3, reiserfs, vfat, iso9660, proc, smbfs, nfs (puede utilizar cat /proc/filesystems para ver los tipos que su kernel soporta. Mire la pagina del manual de mount para una larga lista de sistemas de archivos).

2.- ¿Qué comando se usa para particionar un disco duro?

®.- fdisk (Otros comandos disponibles son cfdisk, sfdisk y GNU Parted).

3.- ¿Cuál es los propósitos de la Cuotas?

®.- Limitar el monto de espacio que el usuario puede utilizar.

4.- ¿Qué archivo se puede utilizar para especificar montar volúmenes automáticamente?

®.- /etc/fstab

Ejercicio 5-1: Permisos de Archivos

A continuación es una sesión:

```
[alex@www ABIERTOS]$ id -a
uid=5027(alex) gid=5027(alex) groups=5027(alex)
[alex@abiertos]$ ls -ld . arch* /etc/passwd /etc
```

```
drwxrwxrwx 2   root  root    4096     Jul 19 19:29     .
-rw-rw-r--   1   alex  alex    0        Jul 19 19:29     arch1
-rw-r-----   1   alex  alex    0        Jul 19 19:29     arch2
-rw-rw-rw-   1   alex  alex    0        Jul 19 19:29     arch3
drwxr-xr-x  37 root  root    4096     Jul 19 09:27     /etc
-rw-r--r--   1   root  root    4435     Apr 4 20:24      /etc/passwd
```

1.- Basado en la información arriba presentada, ¿cuáles de las siguientes operaciones son permitidas?

```
$ vi arch1
$ more arch2
$ ls -l > arch1
$ less /etc/passwd
$ rm arch2
$ rm arch3
$ cp arch1 arch4
$ rm /etc/passwd
```

®.- – Estas opciones son permitidas :

```
$ vi arch2
$ ls -l > arch1
$ more /etc/passwd
$ rm arch3
$ cp arch1 arch4
```

Ejercicio 5-2: Trabajar con Utilidades de quota

1.- Modifique esta línea del /etc/fstab para permitir cuotas para los usuarios y grupos:

```
/dev/hda1 /home ext3 defaults 1 2
```

®.– /dev/hda1 /home ext2 defaults,usrquota,grquota 1 2

2.- Examine esta salida del comando

```
# edquota -u ivelis:
Quotas for user ivelis:
/dev/hda1: blocks in use: 7502, limita (soft = 9600, hard = 12500)
inodes in use: 735, limits (soft = 2605, hard = 3600)
```

Escriba el comando para duplicar esta salida de cuota para todos los usuarios del grupo Operadores.

®.– # edquota -p tempuser -g

3.- ¿Cuál de los comandos el suite de cuotas produce listado para todas las cuotas en el sistema?

®.– # repquota -a

4.- ¿Cuál daemon permite obtener información de cuota de directorios montados remotamente?

®.– Este servicio es proporcionado por el demonio rquotad.

Ejercicio 5-3: Sistema de Archivos

Escriba los comandos para efectúar las siguientes tareas:

1.- Crear un sistema de archivos tipo ext3 en un disco de 200-MB hda3

®.- # mkfs -t ext3 /dev/hda3 400000

2.- Monte este sistema de archivos recién creado en /mnt/hda3. Cree el directorio si no existe ya.

®.– # mount -t ext3 /dev/hda3 /usr

3.- Crear un sistema de archivos tipo ext2 en un disco de 150-MB hda5

®.- # mkfs -t ext2 /dev/hda6 300000

4.- Monte este sistema de archivos en /mnt/hda5. Cree el directorio si no existe ya.

®.- # mkdir /usr/lib

mount -t ext2 /dev/hda5 /usr/lib

5.- Crear un sistema de archivos tipo Minix en un disco de 100-MB hda6 y montélo en /mnt/hda5/Minix.

Cree el directorio si no existe ya.

®.- # mkfs -t minix /dev/hda6 200000

mount -t minix /dev/hda6 /home

6.- Desmonte los tres sistemas de archivos; Note como debe desmontar /mnt/hda5 antes de /mnt/hda5/Minix.

®.- # umount /home/

umount /usr/lib

umount /usr

Ejercicio 5-4: Identificar los Archivos Recuperados

Dado la siguiente información del directorio /lost+found:

 # cd /lost+found
 # file *
000973: ASCII text
000979: comands text
001256: iAPX 386 executable not stripped
001385: C source code
001576: data

¿Cuales comandos usarías para identificar el contenido de cada archivo?

1.- El archivo 00973

®.- Lectura : less, more, tail, vi, cat

2.- El archivo 000979

®.- Lectura : less, more, tail, vi, cat

3.- El archivo 001256

®.- No se ejecuta; utilice el comando strings

4.- El archivo 001385

®.- Lectura : less, more, tail, vi, cat

5.- El archivo 001576

®.- No se ejecuta; utilice el comando strings

Ejercicio 5-5: Examinar y Revisar Sistema de Archivos

Escriba los comandos para ejecutar lo siguiente:

1.- Colocar el número máximo de veces encendido en /dev/hda1 antes de ejecutar fsck a 20.

®.- # tune2fs -c 20 /dev/hda1

2.- Diga dos comandos para revisar los bloques defectuosos (bad blocks) en /dev/hda2 (con 65,535 blocks)

®.- # fsck.ext3 /dev/hda2 o badblocks /dev/hda2 65535

3.- Muestre el progreso (sin la ejecución actual) para una revisión del sistema de todos los sistemas de archivos listados en el archivo /etc/fstab

®.- # fsck -NA

4.- Ejecute una revisión de sistema de archivos en /dev/ hda3 con barra de progreso, especificando que el sistema de archivos es VFAT.

®.- # fsck -t vfat -Cvr /dev/hda3

Ejercicio 5-6: Uso del Comando mount con NFS

Escriba los comandos para ejecutar lo siguiente:

1.- Con que comando montaria usted el directorio /usr/share desde el equipo abiertos12 en el punto de montaje local /mnt/disco con la opcion de background y timeout?

®.– # mount -o bg,soft, int abiertos12:/usr/share /usr/share

2.- Complete las opciones de mount para los siguientes dos archivos /etc/fstab:

®.- # grep home /etc/fstab

ivelis:/home/ivelis /home/ivelis nfs,soft,bg,int 0 0

rsh ivelis grep home /etc/fstab

fcld12:/home/ivelis /home/ivelis nfs,soft,bg,int 0 0

PREGUNTAS POST- EXAMEN

1.- ¿Qué debes hacer para usar disco para almacenar archivos en Linux?

®.- Usted debe crear un file system en el disco.

2.- ¿Cuál es el proceso de borrar archivos con el comando rm?

®.- Cuando corres el comando rm, estas solamente removiendo un link hacia el archivo. Una vez el kernel ve que no hay links en el archivo, lo borra. El usuario nunca borra archivos.

3.- ¿Cuál es la diferencia entre quotas HARD y SOFT?

®.- Soft quota permite que el usuario exceda quotas por un periodo de tiempo especificado. Hard quota no permite a un usuario exceder su limite.

4.- ¿Para qué se usa SAMBA?

®.- Samba es utilizado para integrar sistemas GNU/Linux con redes existentes basadas en sistemas operativos que utilizan el protocolo SMB.

5.- ¿Para qué se utiliza el cache del KERNEL?

®.- El proposito del cache es mejore el rendimiento reemplazando acceso de data al disco como el kernel frecuentemente encuentra la data que quiere en cache.

CAPITULO 6

PREGUNTAS PRE-EXAMEN

1.- ¿Para manualmente crear una cuenta de usuario desde la línea de comandos, que archivos hay que editar?

®.- Edite los archivos /etc/passwd y /etc/groups. Usted deberia tambien crear el directorio home del usuarioy asignarle los permisos y las propiedades del usuario. Si tiene habilitado shadow passwords, necesitaras editar el archivo /etc/shadow tambien. (Como sea deberias siempre utilizar las herramientas disponibles en lugar de hacer cosas manuales. Si algunos pasos son hechos utilizando las herramientas y otras manuales, podria llegar a ser inconsistente.

2.- ¿Cómo puede un usuario agregar su teléfono al archivo /etc/passwd?

®.- Utilice el comando chfn

3.- Cuando un usuario nuevo es creado, algunos archivos son creados en su directorio home. ¿De dónde son estos archivos copiados?

®.- Se copian desde /etc/skel.

4.- ¿Nombre los archivos de configuración que se pueden ejecutar en el momento de login de shell bash?

®.- /etc/profile, todo en el directorio /etc/profile.d, /etc/bashrc, ~bash_profile y ~/bashrc.

Ejercicio 6-1: Agregar y Modificar Usuario

1.- Agregue un usuario llamado miguel

®.- # useradd -m miguel

2.- Agregue un usuario llamado ivellise y especifique el shell sh.

®.- # useradd -m -s /bin/sh ivellise

3.- Agregue un usuario llamado alex, usando /home/alex, como su home directorio

®.- # useradd -m -d /home/alex alex

4.- Agregue un usuario llamado peque especificando su UID de 400 en el grupo de staff

®.- # useradd -u 400 -g staff peque

5.- Modifique el usuario alex para que pueda usar el shell bash

®.- # usermod -s /bin/bash alex

6.- Modifique el usuario alex para conseguir un nuevo UID de 401

®.- # usermod -u 401 alex

Ejercicio 6-2: Seguridad de Cuentas de Usuarios

1.- Agregue un passwd a miguel

®.- # passwd miguel

2.- Exija a miguel que cambie su contraseñas, la próxima vez que se haga un login

®.- # passwd -f miguel

3.- Habilite característica de vencimiento al passwd de alex (min. 21, max 31, advertencia 7)

®.- # chage -m 21 -M 31 -w 7 alex

4.- La fecha de expiración de ivellise a la fecha 24 de abril del 2003-07-24

®.- # usermod -e 04/24/2003 ivellise

5.- Cierre (bloquee) la cuenta de ivellise.

®.- # chage -E 0 ivellise

6.- Vuelva a abrirla (desbloquearla).

®.- # chage -E 01/08/05 ivellise

Ejercicio 6-3: Administrar los Usuarios

1.- Utilice el comando useradd para agregar el usuario juan (nombre completo Miguel Antonio) con el UID de 355. No olvides utilizar la opción –m para crearle su directorio home.

®.- # useradd -u 355 -c “Miguel Antonio” -m juan

2.- Corregirle el nombre a miguel a Miguel Antonio y agrégale su shell de /bin/bash.

®.- # usermod -s /bin/bash -c “miguel” miguel

3.- Como root, utiliza el comando chage -l para mostrar el estatus de la protección del password de juan, y cambiárselo a lo siguiente (Max numero de días = 7, Min número de días = 2, Advertencia numero = 7).

Luego ingresa como miguel y trata de cambiar el password. Si no puedes, su a root y arregla el problema.

Salga del shell de root y cambia el password de miguel.

®.- # chage -m 2 -M 7 -W 7 juan

4.- Cree un nuevo grupo y llámelo “estudiantes”. Modifique a juan que sea un miembro del grupo estudiante; pero no modifique es grupo default de juan. En un comando, agregue un nuevo usuario ivelis con su nombre completo de Ivelis A González, UID 359, y miembro suplementario del grupo estudiante (no cambies el default de su grupo). Agréguele el password a luis y salga del shell. Ingrese como ivelis, y como el, entre los siguientes comandos:

$batch
date
^ D

Esto se asegurara que exista un mensaje de correo para ivelis. No lea este mensaje aun; queremos que permanezca el la caja de correo.

®.- # groupadd estudiantes
usermod -G estudiantes juan
useradd -c “Ivelis A Gonzalez” -u 359 -G estudiantes
passwd luis

5.- Remueva la cuenta ivelis, incluyendo su directorio home. Ahora busque si ivelis es propietario de algunos archivos en el sistema.

Utilice el comando useradd para crear de nuevo el usuario ivelis con los mismos parámetros de antes.

Usted no podrá porque el sistema no le permitirá reusar el UID de ivelis (digamos que el 421) por otros 20 días.

¿Qué opción de comando debieras especificar para asegurarte que reuse este UID de 421?

®.- #userdel -r ivelis
find / -user 319 -print
userdel -d 0 -r ivelis

6.- Liste todos los grupos de los cuales root es miembro (no olvide el grupo defecto).

®.- # grep root /etc/group

®.- # grep root /etc/passwd

Ejercicio 6-4: Administrar home de Usuarios y Directorios

1.- Cree un directorio de nombre /home/esqueleto, el cual contiene todos los archivos de /etc/skel (con todos los archivos incluyendo los ocultos). Cree dos directorios vacíos llamados libros y mascotas en /home/esqueleto. Cree un nueva cuenta para mike (nombre completo michael foster, UID 511) con su archivo skel en /home/esqueleto. Verifique que los archivos correctos existen en /home/mike.

®.– # mkdir /home/esqueleto
cd /etc/skel
cp -r ./* /home/esqueleto
touch libros mascotas
useradd -d /home/esqueleto -c “michale foster” -u 511 mike
cd /home/esqueleto

2.- Agregue una cuenta nueva para jazzy (nombre Maria Francisca Arias, UID 575) pero no le cree el directorio home. Asígnele una contraseña y verifíquela:

 # su - jazzy
 # pwd

¿Cuál es su directorio actual mostrado por el ultimo comando? Porque no es /home/jazzy? Salga de esta cuenta y intente ingresar como jazzy al sistema. Usted no podra, ya que el directorio home de jazzy no existe.

®.– # mkdir /home/jazzy
cd /home/esqueleto
cp -r *.* /home/jazzy
chown -R jazzy /home/jazzy
chgrp -R staff /home/jazzy

Ingrese como root al sistema y manualmente cree el directorio home/jazzy utilizando una copia desde /etc/skel.

Ingrese como jazzy y verifique que puede guardar un archivo en /home/jazzy/tarea.

3.- Cambie el UID de juan de 355. Y ingrese el siguiente comando:

 # ls -la /home/juan

¿Cómo se ve todo, bien? ¿Qué se le olvido?

®.– # usermod -u 355 juan

Usted no cambio las propiedades de todos los archivos de juan

 # find / -user 320 -print | xargs chown juan
o
 # find / -user 320 -exec chown juan {} \;

Ejercicio 6-5: Ejemplo de Variables de Ambiente

¿Cuál de estos archivos es mantenidos por el administrador?

/etc/profile
/etc/bashrc
$HOME/.bashrc
$HOME/.bash_profile

®.– /etc/profile y /etc/bashrc

Ejercicio 6-6: Variables de Ambiente del Usuario

1.- Agregue un nuevo usuario llamado apagar con el comando useradd, cual llama el programa /sbin/shutdown como su shell de login. Recuerde que shutdown debe ser ejecutado desde el directorio raíz (/), y debe ser ejecutado por root (UID=0). Recuerde que debes poder apagar el sistema con simplemente ingresar con el usuario apagar.

** Necesitara utilizar la opción –o del comando useradd para especificar el UID de "0".**

®.- # useradd -u 0 -o -d / -c "Apagar" -s /sbin/shutdown apagar

2.- Agregue un nuevo usuario llamado fecha con el comando useradd, cual llama el programa /bin/date como su shell de login. Asigne un password en blanco y pruébelo con su.

®.- # useradd -d /tmp -s /bin/date fecha

3.- Ingrese al sistema como miguel e investigue el valor de su mascara de default. ¿Dónde se establece este valor?

®.– La mascara puede ser agregada en uno de 3 lugares, mostrados abajo en orden de prioridad donde la prioridad mas alta es la primera.

$HOME/.bash_profile
/etc/profile
/etc/default/login

Luego ejecute los siguientes comandos:

```
$ mkdir prueba
$ cd prueba
$ umask 0
$ touch um000
$ umask 022
$ touch um022
$ umask 077
$ touch um077
$ umask 770
$ touch um770
$ ls -l
```

Analice los diferentes permisos de archivos. ¿Cuales permisos le gustaría utilizar de default para crear archivos?

Asigne un umask correctamente para que en su próximo login este sea su umask default. Haga este cambio y salga del sistema e ingrese de nuevo para verificar sus cambios.

¿Qué hay de extraño en los permisos del archivo um770?

Ejercicio 6-7: Ambiente Restringido del Usuario

1.- Modifique los archivos de ambiente de miguel, para que el no pueda cambiar o borrar su perfil de usuario.

®.- # chown root /home/miguel/.bash_profile

2.- Pruebe sus cambios ingresando como miguel y verifique que el no puede editar su archivo .bash_profile o borrarlo.

Ejercicio 6-8: Trabajando con Tipos de TERM

Para este ejercicio no se dan soluciones.

1.- ¿Qué despliega el prompt del usuario después que ejecute la siguiente secuencia de comandos del shell bash?

```
$ tput rev
$ tput rmso
$ REV=$(tput rev)
$ NRM=$(tput rmso)
$ PS1=$REV$(uname -n)$NRM: ‘echo $PWD:’
```

Ejercicio 6-9: Login y Terminales

1.- Cambia los default del login global para que el valor de la umask sea 077.

®.- Agrega una linea UMASK=077 a /etc/default/login y borra algunas lineas en /etc/profile

2.- Ingrese con alex y escriba la configuración de las teclas especiales descrita por stty?

erase?
intr.?
kill?
eof?

Cambia su tecla de “intr” Que sea CTRL+A y que tu tecla de kill sea CTRL+B. Digite un comando a medias y presione CTRL+C. Describa lo que paso.

®.- # stty intr ^A kill ^B

Presione CTRL+A y Describa lo que paso ahora.

Digite un comando a medias y presión CTRL+B

3.- Ingrese como Alex y determine cual es su carácter de quit es probablemente CTRL+\, borrar el siguiente comando que debe tomar un buen rato y presiona las teclas de quit. $ ls -lR /

®.- para determinarlo es con el comando: - $ stty -a

Busque el archivo core que se genero.

®.- El archivo debe ser bastante grande (alrededor 150,000 bytes)

Entre los siguientes comandos:

```
$ rm core*
$ ulimet -c 0
```

Trate la primera parte de la pregunta y fíjese la diferencia en el tamaño del core.

®.– Esta vez el archivo core tuvo un tamaño

4.- ¿Cuál es su tipo de terminar en uso?

Entre los siguientes comandos:

```
$ OTERM=$TERM
$ TERM=wyse50
$ vi
:q! #salga del vi puesto que todo salio mal.
$ TERM=$OTERM
```

¿Porque no pudo vi dibujar en la pantalla?

®.– El tipo de terminal fue incorrecto y vi utilizo la secuencia de escape equivocada para limpiar la pantalla, mover el cursor, etc

5.- Digite los siguientes comandos (NOTE que los primeros 3 usan paréntesis y los ultimos llaves):

```
$ BOLD=$(tput bold)
$ BLINK=$(tput blink)
$ NORM=$(tput sgr0)
$ PS1='${BOLD}Yes ${BLINK}Master${NORM}? '
$ tput clear
```

6.- Asegúrese a colocar todo como estaba antes de comenzar este ejercicio.

PREGUNTAS POST- EXAMEN

1.- ¿Qué es un terminar virtual? ¿Cómo puedes distinguir en cual terminar estas?

®.– Es un metodo que permite multitarea que ofrece varias consolas (prompts) sobre un dispositivo fisico (monitor). El comando tty regresara una cadena diferente por cada terminal virtual.

2.- ¿Describa como corregir un terminar que no responde?

®.– Cambiar a un diferente terminal virtual y utilice el comando sane o reset o mata el login shell para la terminal colgada dependiendo de las circunstancias.

3.- ¿Describa el rol de mingetty en los Login de los usuarios?

®.– Este configura las caracteristicas de la terminal, el cual permite hacer login llama un prompt y solicita nombre y clave al usuario.

4.- ¿Cómo puede un usuario redefinir las opciones del terminar?

®.- – Utilizando el comando stty

5.- ¿Describa como y porque un administrador de sistema debe configurar un directorio con el bit SGID?

®.- El bit SGID podria establecer por configuracion los permisos del sistema asi: chmod 4770 nombre

CAPITULO 7

PREGUNTAS PRE-EXAMEN

1.- Nombre los cinco campos del archivo crontab

®.– minuto, hora, dia del mes, mes y dia de la semana

2.- ¿Cómo edita usted una entrada de crontab?

®.– # crontab -e

3.- ¿Cómo puede usted programar para que un programa se ejecute una sola vez ?

®.– Utilice el comando at.

4.- ¿Qué programa debe estar ejecutandose para que una entrada de crontab se ejecute ?

®.– crond (El demonio cron es nombrado crond sobre algunos sistemas y cron en otros.)

5.- ¿Qué debe usted hacer despues que ha efectuado el backup de un archivo ?

®.– Verificar que este puede ser restaurado.

6.- Cuáles son las ventajas y desventajas de los backups sobre la red ?

®.– Ventajas: Solamente necesitan realizar copias del hardware en una maquina, copia centralizada para administracion.

®.– Desventajas: Carga sobre los recursos de la red

7.- ¿Cuáles son las diferentes herramientas disponibles de GNU/Linux para hacer backups de sus sistemas ?

®.– cpio, tar y dd.

8.- Liste 2 diferentes maneras de desempaquetar archivos tar con gzip. Ej. archivo.tar.gz

®.– $ tar -xvzf xyz.tar.gz
$ zcat xyz.tar.gz | tar xf -
$ gzip -dc xyz.tar.gz |tar xf -
$ gunzip -c xyz.tar.gz |tar xf -

Ejercicio 7-1: Usos de at y cron

1. Usted debe fijar las variables de entorno VISUAL o EDITOR antes de usar crontab ya que puede ser que le habrá un editor de textos que usted no sepa utilizar. Los editores de textos recomendados son pico, jed, emacs, y vi. (Note que no todas las distribuciones tendrán todos éstos instalado)

®. – $ export VISUAL=pico
$ export EDITOR=pico

2. Edite su tabla del cron usando el comando crontab

®.- $ crontab -e

3. Cree un trabajo del cron para enviarse un saludo por E-mail en su cumpleaños. Crear otro trabajo del cron de enviarse un E-mail 10 minutos de ahora. Cerciorarse de que usted reciba ese mensaje.

®.– Para enviarse un correo usted mismo, simplemente utilice el comando echo porque toda la se enviar por correo a usted. Necesitara colocar tiempo arbitrario del dia para enviar el mensaje porque si permanecen las horas y minutos con asterisco deberia correr el trabajo cada minuto. Asi que asumiento que a las 9am y su cumpleanos es Julio 26. La entrada en el crontab seria como:
$ 0 9 26 07 * echo “Feliz Cumpleaños”

4. Utilice at para enviarse un mensaje por E-mail 5 minutos de ahora. Cerciorarse de que usted lo reciba.

Pruebe con varios formatos de tiempo.

®.- La manera mas facil para crear un trabajo cada 5 minutos deberia ser:
$ at now + 5 minutes

Despues de presionar ENTER, Deberias introducir los comandos. De nuevo, echo es una buena eleccion para enviar la salida a tu cuenta. Despues que entre el comando y presione ENTER, presione CONTROLD al final de la entrada de comandos.
at> echo “It is now 5 minutes later than before”
at> ^D

5. Liste la cola de at y quite algunas entradas

®.– Para ver los trabajos de at en espera, utilice el comando atq:
$ atq

Para eliminar el numero de trabajo que quieres borrar. Si seleccionas eliminar el numero 10, seria de esta manera
$ atrm 10

Ejercicio 7-2: Uso de cpio

Explique que sucede al ejecutar los siguientes comandos.

1.- # find . -print \ afio -ovb > /dev/ftape

®.- Todos los archivos y directorios el actual directorioson archivados en un dispositivo de cinta.

2.- # afio -tb < /dev/ftape

®.- Una lista de todos los archivos almacenados en una cinta son desplegados.

3.- # cd /tmp;

afio -ivab < /dev/ftape

®.- El archivo almacenado en la cinta es restaurado bajo /tmp

4.- # cd /;
 # find etc home var -print | afio -ovb > /dev/ftape

®.- Todos los archivos bajo /etc, /home y /var son almacenados a cinta.

5.- # cd /tmp;
 # afio -ivaby etc/init.d /dev/ftape

®.- El archivo /etc/init.d es recuperado desde el archivo almacenado en la cinta.

Ejercicio 7-3: Copiar un Disco

1.- ¿Cómo copiarías un disquete preformateado?

®.– Inserte el disco a ser copiado.
$ dd if=/dev/fd0 og=/tmp/fd.image

Cambia el disquete (Inserte uno en blanco)
 # dd if=/tmp/fd.image of=/dev/fd0

Ejercicio 7-4: Usar tar, gzip, y compress

Este ejercicio se ocupará del uso de tar, gzip, y de la utilidad de comprimir compress para archivar y de comprimir varios archivos para crear un backup.

1.- ¿Cuál es la manera más simple de colocar el contenido del directorio /bin en un solo archivo tar y nombrado bin.tar?

®.- $ tar -cf bin.tar /bin

2.- ¿Usando el gzip, cómo se puede comprimir un archivo para ahorrar el mayor espacio?

®.- $ gzip -9 archivo o gzip archivo --best

3.- ¿Cómo se puede comprimir el archivo usando compress?

®.- $ gzip bin.tar (este creará el archivo bin.tar.Z)

4.- ¿Cómo se pueden combinar estos pasos de una manera más simple? ¿Si es así, cómo?

®.- $ tar -cfz bin.tar.gz

5.- ¿Cómo se puede descomprimir el archivo gzippped? ¿ Cómo se puede descomprimir el archivo compress?

®.- $ gunzip bin.tar.gz; uncompress bin.tar.gz

6.- ¿Qué comando expandirá el archivo comprimido a un archivo sin comprimir?

®.- $ tar -xf bin.tar

7.- ¿Cómo se pueden ejecutar los comandos uncompress y untar al mismo tiempo?

®.- $ tar -xfz bin.tar.gz, tar -xfz bin.tar.Z

Ejercicio 7-5: Backups y Restaurar

1.- Abra una sesión como root y ejecute el siguiente comando:
 # touch /home/backup

2.- Cambie al directorio /usr y hágale un backup al directorio dict a disquete usando tar. El dispositivo del disquete se encuentra en /dev/fd0.

®.- # tar -xvf /dev/fd0

¿Por qué piensa usted que se le pidió cambiar al directorio y hacer el backup a los discos de backup en dos acciones separadas? ¿Cómo puede usted verificar que el backup trabajara?

®.- Ahi que probarlos....respuesta incorrecta...!!!!!

Restaure el backup en el directorio /tmp/dict.tar.

®.– # mkdir /tmp/ditc.tar
cd /tmp/dict.tar
tar -xvf /dev/fd0

3.- Repita el backup anterior y restaurarla con el cpio (restaure en /tmp/dict.cpio).

®.– # cd /usr
find dict -print | cpio -ocv > /dev/fd0
cpio -itvc < /dev/fd0
mkdir /tmp/dict.cpio
cd /tmp/dict.cpio
cpio -ivcdum < /dev/rfd0

4.- ¿Si usted no supiera qué formato de datos fue escrito al disquete, cómo usted lo descubriría?

®.– La manera mas rapida seria :
 # dd if=/dev/fd0 of=/tmp/unknwn
 # file /tmp/unknown

Ejercicio 7-6: Backups Programados

1. Repita las operaciones de backup de los pasos 2 y 3 en el ejercicio 7-5, primero tar use y entonces cpio, pero mida el tiempo de la operación usando el comando time como en los siguientes ejemplos:
 # cd /usr
 # time tar cvf del /dev/fd0 dict
 # time (find dict -print | cpio -ocv > /dev/fd0)

¿Por qué usted nos piensa utilizó paréntesis en el segundo ejemplo?

®.– Para agrupar los comandos find y cpio juntos; De otra manera, el tiempo deberia estar solamente aplicado al comando find

2. Repita los dos backups otra vez, pero esta vez, de salida a los archivos a /tmp/tar y /tmp/cpio más bien que a disquete.

Repita otra vez, pero esta vez haga que tar y cpio escriban a la salida estándar, pase la salida por tuberí-

as al comando compress, y redireccione todo a los archivos /tmp/tar.Z y /tmp/cpio.Z. Los ejemplos siguientes demuestran lo qué se requiere:

 # cd /dict
 # tar cvf - dict | compress > /tmp/tar.Z
 # find dict -print | cpio -ocv \ compress > /tmp/cpio.Z

Ahora vamos a comparar los tamaños de los cuatro ficheros archivados.

®.- # cd /dict
tar -cvf /tmp/tar dict
find dict -print | cpio -ocv > /tmp/cpio

3. ¿Cómo haría usted un backup de todos los archivos que se han modificado desde que /home/backup fue creado?

Ejercicio 7-7: Tecnicas de Backup

1. Ponga el disquete que contiene un backup en la disquetera y ejecute los siguientes comandos:

 # dd if=/dev/fd0 count=1 of=/tmp/floppy
 # file /tmp/floppy

¿Qué piensan usted que acaba de hacer?

2. De formato a un disquete con un sistema de archivos DOS.

Copie los archivos hosts a este disquete. Ejecute un ls para probar que esta hay. Ahora copie este archivo de nuevo a /tmp y llámelo DOS.hosts. Analícelo para asegurarse que es valido este archivo.

Copy el disquete entero a un archivo y llámelo /tmp/dos.fd. Ejecute el siguiente comando para corromper el disquete del DOS:

 # dd if=/usr/bin/date of=/dev/fd0

¿Puede usted conseguir un listado del directorio del DOS disquete?

Ahora copie la imagen de su /tmp/dos.fd al disquete.

¿Puede usted ahora conseguir un listado DOS del disquete?

PREGUNTAS POST- EXAMEN

1.- ¿Cuál del siguiente es entradas válidas del archivo del crontab?

A.- * * * * echo “Hola” > /tmp/hola.out
B.- # Mes Dia Hora Minuto Comando
C.- 0 2/4 * * * echo “Hola”
D.- * * 0 * echo “Hola”
E.- 11111 echo “Hola”
F.- 13 13 13 13 * echo “Hola”
G.- */4 * * * * echo “Hola”

2.- ¿Cuál de los siguientes es una válida representaciones de time/date?

A.- 1
B.- next friday

C.- last tuesday
D.- 07/26/1961
E.- 7/26/2003 9am
F.- 1 tomorrow
G.- now
H.- 4:11pm+ 2 days
I.- 10:00 am July 31
J.- noon + 2 hours
K.- midnight +1 week

3.- Muestre una entrada de crontab que ejecute uptime a la 1:00 de la mañana cualquier martes y cualquier día del mes que termine en un cero.

4.- ¿Cómo programarías para que un programa se ejecute a la 1:23 a.m. cada lunes y a las 2:34 a.m. cada martes?

5.- Ejecutas un backup completo el domingo. Ejecutas backups incrementales en las noches de lunes y martes.

Usted necesita restaurar un archivo la mañana del miércoles. ¿Cuántas cintas usted necesita utilizar para restaurar el archivo si el archivo se modifica todos los días?

6.- ¿Qué utilidad de backup puede usted utilizar para encontrar archivos específicos para hacer su backup?

De un ejemplo del comando que usted utilizaría.

7.- ¿Cuáles son algunos de los directorios importantes en un sistema Linux que cambia con frecuencia y tienen la necesidad de ser backup regularmente?

CAPITULO 8

PREGUNTAS PRE-EXAMEN

1.- ¿Dónde se van los trabajos en lo que esperan su turno a la impresora?

2.- ¿Qué utilidad grafica se puede utilizar para configurar la impresora GNOME/KDE?

3.- ¿Qué necesita estar instalado para imprimir por impresoras conectadas a una maquina de Windows?

4.- ¿Cuál es el comando para ver la cola de impresión?

5.- Enviaste tres trabajos a imprimir por Laser1; los archivos se encuentran en el queue (sus números son 1,2 y 3), ¿Cómo puedes hacer que el 3 imprima primero que todos?

Ejercicio 8-1: Configurar y Usar Impresoras de Red

En este ejercicio configuraremos un printer de red y compartiremos con varios sistemas. Un sistema con una impresora conectada directamente que estableceremos como un servidor de impresión. Los otros sistemas lo llamaremos clientes.

1.- Trabajando en el Printer Server, verifique que puedes enviar data a la impresora y que imprime.

2.- El Servidor de Impresión, configure el Printer, utilizando una utilidad grafica de configuración. Imprima una página de Prueba.

3.- En un cliente, agréguelo con una utilidad gráfica, al network printer que configuro en el dos. Imprima una página de prueba.

Ejercicio 8-2: La Cola/Queue de la Impresora

1.- En su sistema deshabilite el printer queue/cola.

2.- Imprima 3 archivos diferente (recuerde el orden). Mire en los trabajos por imprimir con el comando: **lpc -status y lpq**

3.- Configure que el último trabajo es el próximo en imprimir.

4.- Cancele el trabajo del medio utilizando el comando lprm.

5.- Cancele todos los trabajos utilizando el comando lprm.

PREGUNTAS POST- EXAMEN

1.- ¿Cómo eliminaría usted los reportes de impresión desde el queue del (asumiendo que es único trabajo en el queue) administrador de la impresora laser?

2.- ¿Cómo investiga usted el estatus del laser printer admin?

3.- ¿Supongamos que el administrador de impresora laser es la impresora por defecto. Liste dos maneras diferentes de listar los trabajos en estera del queue?

CAPITULO 9

PREGUNTAS PRE-EXAMEN

1.- ¿Cuál archivo se utiliza para determinar cual evento se almacena en cual registro

2.- ¿Cuál es el propósito de los archivos de Registro?

3.- ¿Qué se puede hacer con archivos de registro que se tornan muy grande?

4.- ¿Qué criterio se utiliza para determinar cual evento son registrados y dónde

5.- ¿Cuál archivo del registro puede ser considerado como el central que acapara la mayor parte de los mensajes syslog?

Ejercicio 9-1: Encontrar y Accesar Archivos de Registro

Durante este ejercicio, usted localizara algunos de los archivos de registro del sistema y correctamente los analizará. El directorio más importante para observar cuando hace uso los archivos de registro es /var/log. Este directorio es típicamente el destinario para todos los registros del sistema, aunque la utilidad syslogd permite el cambio de dirección a cualquier directorio. Tener acceso a los archivos de registro es muy importante para cualquier administrador. No hay soluciones proporcionadas para este ejercicio.

1. Ingrese a su máquina como root. Si usted esta usando el X, abra una ventana terminal y ejecute el siguiente comando:
 # cd /var/log

Esto le colocará en el directorio con sus archivos de registro. Ejecute ls -la para listar los archivos en este directorio. Éstos son los registros de los demonios del syslogd y klogd.

2. ¡Búsque el destino de los mensajes del kernel en el archivo de configuración del syslogd ejecutando el siguiente comando:
 # grep kern /etc/syslog.conf

Esto busca en el archivo por la secuencia de caracteres “kern” y despliega los resultados en pantalla. Usted debe recibir una salida similar a la siguiente:
 # Registre todos los mensajes del kernel a la consola.
kern.* /var/log/kernmsg

3. Viendo cómo este archivo de registro puede crecer muy grande, usted solo desea analizar las últimas 50 líneas de él. El carácter pipe (|) se utiliza en este ejercicio para pasar la salida de un programa a otro, y luego analizar esta salida. Deseamos poder leer las 50 últimas líneas del registro, incluyendo las que normalmente se pasarían del límite de la consola. Esto se puede lograr ejecutando el siguiente comando:
 # tail -50 kernmsg | less

4. Las entradas en un archivo de registro estándar se escriben así:
date hostname facility: message

Si por ejemplo usted hallase la siguiente entrada:

Jul 26 18:52:21 vfs kernel: PCI latency timer (CFLT) is unreasonably low at32. Setting to 64 clock

Y usted desea darle salida solamente a las cinco ultimas líneas que contienen la secuencia del PCI del

registro del mensaje del kernel, usted puede combinar la funcionalidad del grep con la de la tail y ejecutar lo siguiente:
 # grep PCI kernmesg | tail -5

5. Utilice el comando head si usted desea ver las primeras diez líneas del archivo de diario anterior. Ejecute el siguiente comando:
 # head -10 kernmsg

6. ¿Cómo puede usted ver un registro mientras se esta actualizado sin tener que volver a ejecutar éstos comandos cada vez? Hay realmente dos maneras de hacer esto. La primera requiere que incorpore una línea como el siguiente:
***.* /dev/tty12**

en el archivo /etc/syslog.conf y ejecutar un killall -HUP syslogd para hacer que el demonio de syslogd vuelva a leer su archivo de configuración. Esto envía toda la salidas del registro a la doceava terminal virtual, así que cualquier momento que desea ver en la pantalla que pasa en los registros, solo tienes que presionar ALT+F12 para cambiar al terminal virtual y hay estará la información.

La segunda manera de ver como se escribe el registro es utilizar el comando tail con el parámetro especial que te permite seguir su progreso. El comando es:
 # tail -f /var/1og/messages

no devolverá el prompt después de que haya ejecutado (como es lo normal de otros comandos y el de tail).

Continuará actualizándose con cada línea que se le escriba al archivo de registro. Esto es una técnica muy útil para eliminar errores (debuggin) o la administración general del sistema.

PREGUNTAS POST- EXAMEN

1.- ¿Qué tipo de archivos usted investigaría para encontrar ciertos problemas de proceso?

2.- ¿Si un administrador deseara saber quien esta en este momento utilizando el sistema, a que archivo del sistema se debe dirigir el?

3.- ¿Cuál archivo del registro nos da una buena indicación del bien estar de nuestro sistema?

4.- ¿Es buena o mala práctica regularmente revisar los archivos del registro?