

# XION ECUADOR

## X I O N G N U L I N U X

TUTORIAL PARA TODO NIVEL DE USUARIO



**Franklin Hermel Coloma Ramírez**

Aporte Personal a la Comunidad de Software Libre



Riobamba, 1 de Abril del 2008

<http://xionecuador.bravehost.com>

Copyright: 2008 Xion Ecuador

**Licencia:** GNU FDL – Licencia de Documentación Libre de GNU es una forma de Copyleft para ser usada en un manual, libro de texto u otro documento que asegure que todo el mundo tiene la libertad de copiarlo y redistribuirlo, con o sin modificaciones, de modo comercial o no comercial.

Para mayor información visite el sitio web de la Free Software Foundation y revise como es el uso de la licencia GNU FDL. <http://www.gnu.org/licenses/fdl.html#SEC4>

**Importante:** No se ofrece ninguna garantía por daños o perjuicios que se den por el uso indebido de la información que se ofrece en el presente documento.

Xion Ecuador – Versión 0.2, Abril 2008

Xion Ecuador – Versión 0.1, Julio 2007

Duchicela 22-45 y Avenida Daniel León Borja

Celular: 59392295787

Riobamba – Ecuador



Este documento es propiedad de Xion Ecuador – Franklin Hermel Coloma Ramírez, se libera bajo los términos de la licencia GNU FDL 1.2.



# XION GNU LINUX

## Introducción

El presente documento es un aporte personal para la Comunidad de Software Libre, constituye una guía fundamental para diferentes niveles de usuario para conocer los aspectos fundamentales de los Sistemas Operativos GNU/Linux.

Surge como fruto de las experiencias adquiridas durante mi vida universitaria en la Escuela Superior Politécnica de Chimborazo ([www.esPOCH.edu.ec](http://www.esPOCH.edu.ec)) y mi pasantía en la empresa ISP de Riobamba Fastnet Cía. Ltda. ([www.fastnet.net.ec](http://www.fastnet.net.ec)).

En el documento podemos encontrar descripciones y ejemplos probados de los principales comandos utilizados en las consolas, los permisos que se otorgan a los usuarios, como gestionar archivos, como configurar los servicios más importantes en un Sistema Operativo GNU/Linux. Se hace referencia de la distribución Red Hat Linux Enterprise Advanced Server 3.0, CentOS 5.0 y Fedora Core 7.0.

Agradezco la colaboración y motivación de los catedráticos de la ESPOCH y de mis amigos para cristalizar este tutorial que ayudara a generaciones futuras a integrarse al maravilloso mundo del Software Libre y utilizar los Sistemas Operativos GNU/Linux.



## Objetivos

- ✚ Otorgar un documento fácil de comprender orientado a todos los niveles de usuarios de Sistemas Operativos GNU/Linux.
- ✚ Motivar a nuevos usuarios a incursionar en el mundo de Software Libre como alternativa al Software Propietario.
- ✚ Mostrar los procedimientos adecuados para la configuración de Servidores WWW, FTP, DNS, DHCP, SAMBA, NFS en sistemas GNU/Linux.
- ✚ Describir lo que es un sistema de archivos y los tipos soportados en Sistemas Operativos GNU/Linux.
- ✚ Gestionar las cuentas de usuarios y diferenciar los tipos de usuarios existentes en los Sistemas Operativos GNU/Linux.

## Shells Usados en GNU/Linux.

Un Shell es un programa interprete especial de los sistemas GNU/Linux, que es el encargado de la interpretación de los comandos digitados por los usuarios y ordenar al computador para que realice una tarea especifica como son: montar particiones, formatear discos, imprimir documentos, listar los archivos del disco duro, mostrar el espacio disponible en el disco, mostrar los procesos que se ejecutan en el sistema, etc.

A continuación se muestra un listado de los Shells soportados por las diferentes Distribuciones GNU/Linux.

TIPO	DESCRIPCIÓN
<b>/bin/sh</b>	Shell Bourne, creado por Steven Bourne.
<b>/bin/ksh</b>	Shell Korn, creado por David Korn añade mejoras con respecto a sh, adoptado como Shell POSIX (1003.2)
<b>/bin/bash</b>	Shell Bourne Again (Shell Bourne de Nuevo), creado por la Free Software Foundation, incorpora las mejores características de ksh y csh, compatible con POSIX y es el shell por defecto en GNU/Linux.
<b>/bin/csh</b>	Shell C, escrito por Bill Joy fundador de Sun Microsystems, usa una sintaxis muy similar al lenguaje C, es mala para la programación de scripts.
<b>/bin/tcsh</b>	Variante de Shell C, admite edición de comandos en línea.

## Comandos más Importantes en GNU/Linux.

Existe una gran colección de comandos que se puede ejecutar en un Sistema Operativo Unix y GNU/Linux.

Un comando es una instrucción que se escribe en el terminal o consola de nuestro sistema operativo anfitrión, cada instrucción le dice a la máquina que ejecute procesos por ejemplo: limpiar el contenido de la pantalla, apagar o reiniciar el ordenador, desplegar el contenido de un archivo, etc.

A continuación se describe las opciones más básicas de los principales comandos usados en GNU/Linux.

**ls.-** El comando "ls" (list) permite listar archivos y directorios.

COMANDO	DESCRIPCIÓN
ls -l	Lista archivos y directorios en forma vertical, muestra el nombre del propietario, permisos, el tamaño, etc.
ls -a	Lista archivos y directorios, además muestra los archivos y directorios ocultos anteceditos por el punto (.) a su nombre.
ls -R	Lista archivos y directorios de forma recursiva mostrando su contenido.

**clear.-** El comando "clear" (clearscreen) permite limpiar el contenido de la pantalla actual debido a la presencia de mucha información como ejemplo la desplegada por el comando "ls".

**pwd.-** El comando "pwd" permite mostrar la ubicación actual dentro del sistema de archivos de GNU/Linux.

**cd.-** El comando "cd" (change directory) sirve para moverse de un directorio a otro.

COMANDO	DESCRIPCIÓN
cd ~	Cambia al directorio personal del usuario actual.
cd /	Cambia al directorio "/" raíz (root) del sistema de archivos, sin importar donde nos encontremos.
cd ..	Abandona un directorio en el que nos encontramos y sube un nivel, se debe tener cuidado de no poner cd.. GNU/Linux emitirá un error debido a que no encuentra un script equivalente con el nombre cd..

**cp.-** El comando "cp" (copy) permite copiar un archivo.

COMANDO	DESCRIPCIÓN
cp x.png x.jpg	Copia un archivo con otro nombre o extensión.
cp x.png x1/x.png	Copiar archivo x.png a otro directorio.
cp -R x1/ x2/	Copia el contenido del directorio x1 al directorio x2 usando la opción -R, caso contrario GNU/Linux no copia directorios.

**mv.-** El comando "mv" (move) permite mover archivos y directorios de un lugar a otros, permite también renombrarlos.

COMANDO	DESCRIPCIÓN
mv xion.jpg /prueba	Mueve el archivo xion.jpg al directorio prueba.
mv xion.jpg xion.png	Cambia el nombre de un archivo y lo guarda en la carpeta actual.

**rm.-** El comando "rm" (remove) permite eliminar archivos o directorios que ocupen espacio en el disco duro. Se debe tener mucho cuidado al usar este comando, puede eliminar archivos críticos del sistema si estamos conectados al ordenador como "root".

COMANDO	DESCRIPCIÓN
<code>rm mxx.mp3</code>	Elimina el archivo mxx.mp3, pide confirmación para hacerlo.
<code>rm -r *.*</code>	Borra de forma recursiva el contenido del directorio actual, se solicita confirmación para hacerlo. Usar con cuidado.

**rmdir.-** El comando "rmdir" (remove directory) permite borrar un directorio que este vacío caso contrario no lo elimina.

COMANDO	DESCRIPCIÓN
<code>rmdir x2</code>	Elimina el directorio x2 que debe estar vacío.

**mkdir.-** El comando "mkdir" (make directory) permite crear un directorio.

COMANDO	DESCRIPCIÓN
<code>mkdir x3</code>	Crea el directorio x3, para ver use el comando "ls" debido a que "mkdir" no presenta ningún mensaje al crear el directorio.

**more.-** El comando "more" permite visualizar un archivo, sin tener la posibilidad de modificar su contenido.

COMANDO	DESCRIPCIÓN
<code>more texto.txt</code>	Visualiza el contenido del archivo texto.txt, si el archivo es extenso more hace una pausa.

**cat.-** El comando "cat" sirve para concatenar uno o varios archivos a la vez, sin embargo, también se usa para visualizar el contenido de un archivo de una sola vez, sin la posibilidad de volver hacia atrás o modificarlo.

COMANDO	DESCRIPCIÓN
cat texto.txt	Visualiza el contenido de texto.txt sin realizar pausas.
cat > texto.txt	Todo lo que digitemos en la consola se añade al archivo texto.txt, para finalizar presione Ctrl+D. Si existía información en el archivo está será reemplazada.
cat >> texto.txt	Añadir texto al final de texto.txt, esto no borra el contenido anterior del archivo.

**ln.-** El comando “ln” (link) establece vínculos que pueden ser simbólicos o duros.

COMANDO	DESCRIPCIÓN
ln texto.txt texto.doc	Crea un enlace del archivo texto.txt al archivo texto.doc.
ln -s x1 x2	Enlace simbólico del directorio X1 hacia el directorio x2.
ln -n texto.txt x2/texto.doc	Crea un enlace blando de texto.txt al directorio x2 con nombre de archivo texto.doc.

**grep.-** El comando “grep” permite buscar en un archivo que contiene una información específica que necesitamos y no sabemos con que nombre lo habíamos guardado.

COMANDO	DESCRIPCIÓN
grep hola /root/*	Busca la palabra “hola” bajo el directorio /root, muestra el nombre del archivo que la contiene así como la línea respectiva.

**du.-** El comando “du” muestra la información sobre el espacio que utiliza un directorio o un rango de directorios dentro del disco duro.

Para mayor información sobre el comando “du” utilice el comando en la Consola de su Distribución GNU/Linux: man du

COMANDO	DESCRIPCIÓN
<code>du -s /root</code>	Muestra la cantidad de espacio ocupada por el directorio /root.
<code>du -s gnu.pdf</code>	Muestra la cantidad de espacio ocupado por el archivo gnu.pdf.

**df.-** El comando “df” presenta la información del espacio utilizado y disponible en un sistema de archivos (vea la sección correspondiente a Sistema de Archivos en este documento, allí podrá encontrar información detallada de los tipos de Sistemas de Archivos soportados por las Distribuciones GNU/Linux) siempre y cuando estos estén montados.

COMANDO	DESCRIPCIÓN
<code>df</code>	Muestra el espacio utilizado por todos los sistemas de archivos.
<code>df /dev/hda1</code>	Muestra el espacio utilizado por el sistema de archivos cargados en la partición hda1.
<code>df /mnt/win</code>	Muestra el espacio utilizado por el sistema de archivo Windows Fat32 montado en el directorio /mnt/win.

**w.-** El comando “w”, muestra quien esta conectado a una consola o logueado al sistema y que procesos esta ejecutando.

**which.-** El comando “which” muestra la localización de un comando.

COMANDO	DESCRIPCIÓN
<code>which ls</code>	Muestra el directorio donde se encuentra el archivo ejecutable del comando “ls”.
<code>which setup</code>	Muestra el directorio donde se encuentra el archivo ejecutable del comando “setup” que es ejecutado por el usuario “root”.

**pstree.-** El comando “pstree” muestra los procesos en forma de estructura de árbol, se puede ver inmediatamente quien es el proceso padre de otro; cuando se quiere

eliminar toda una serie de procesos, y si son todos padres e hijos, es suficiente matar al ancestro común.

COMANDO	DESCRIPCIÓN
<b>pstree -p</b>	Muestra el PID de cada proceso usando la opción “-p”.
<b>pstree -u</b>	Muestra el nombre del usuario que inició el proceso usando la opción “-u”.
<b>pstree -up   less</b>	Esto le da una visión general de toda la estructura de árbol de los procesos.

**top.-** El comando “top” muestra información general y los procesos activos en el sistema de forma dinámica, para salir de la visualización presione la letra “q”.

Cumple las funciones de “ps” y “kill”.

A continuación se muestra las funciones de los subcomandos usados por “top”.

SUBCOMANDO	DESCRIPCIÓN
<b>k</b>	El subcomando k envía una señal a un proceso, “top” le preguntará por el PID del proceso, seguido del número de la señal a enviar (predeterminado es el número 15).
<b>M</b>	Se usa para ordenar el listado de los procesos de acuerdo a la memoria que usan (campo %MEM).
<b>P</b>	Lista procesos de acuerdo al tiempo de CPU que consumen (campo %CPU; este es el método de ordenamiento predeterminado).
<b>u</b>	Se usa para mostrar los procesos de un usuario en particular, “top” le preguntará de cual. Debe ingresar el nombre del usuario, no su UID. Si no ingresa nombre alguno, se mostrarán todos los procesos.
<b>i</b>	Muestra todos los procesos, incluso los que están dormidos; este comando asegura que se muestran sólo los procesos que están en curso de ejecución (los procesos cuyo campo STAT indica R, running, ejecutando) y no los otros.

**ps.-** El comando “ps” lista los procesos activos en el sistema.

COMANDO	DESCRIPCIÓN
ps	Muestra los procesos del usuario actual.
ps a	Muestra los procesos de todos los usuarios conectados al sistema.
ps x	Muestra los procesos que no están conectados con el terminal.
ps u	Presenta información organizada del usuario actual que trabaja en el sistema.
ps aux	Presenta la información de forma personalizada.

**kill.-** El comando “kill” permite matar o detener uno o varios procesos que se encuentran en ejecución. Se debe tener mucho cuidado podríamos matar algún proceso crítico del sistema si estamos conectados como súper usuario “root”. Al utilizar kill, se envía una señal al proceso que la interpreta y espera a que actuara de acuerdo a la señal.

COMANDO	DESCRIPCIÓN
kill -HUP 200	Un proceso con PID 200 se desea reiniciar, se le envía la señal SIGHUP.
kill -TERM 25	Para que el proceso con PID 25 termine su ejecución, se le envía la señal SIGTERM.
kill 25	Es equivalente al comando kill -TERM.
kill -KILL 356	Si un proceso no responde se puede enviar la señal SIGKILL, esta señal no es interceptada por el proceso y es el sistema operativo el que mata el proceso.

**gzip.-** El comando “gzip” es un compresor de archivos en formato gz.

Para obtener más información de las opciones soportadas consulte la ayuda de “gzip”.

COMANDO	DESCRIPCIÓN
<code>gzip -r x1</code>	Comprime el contenido del directorio x1 de forma recursiva.
<code>gzip inst.txt</code>	Comprime el archivo inst.txt en el directorio actual.
<code>gzip --help</code>	Muestra la ayuda de gzip, y una descripción rápida de los comandos internos soportados.

**gunzip.**- El comando “gunzip” es un descompresor de archivos en formato gz.

Es el complemento del comando “gzip”, para mayor información de las opciones soportadas por “gunzip” consulte su ayuda respectiva.

COMANDO	DESCRIPCIÓN
<code>gunzip -r x1</code>	Descomprime el contenido del directorio x1 de forma recursiva.
<code>gunzip inst.txt.gz</code>	Descomprime el archivo inst.txt.gz en el directorio actual.
<code>gunzip --help</code>	Muestra la ayuda de gunzip, una descripción rápida de las opciones de este comando.

**bzip2.**- El comando “bzip2” es un compresor de archivos en formato bz2, ofrece mejor compresión que gzip.

COMANDO	DESCRIPCIÓN
<code>bzip2 -z x.txt</code>	Comprime el archivo x.txt, con la opción “z” se fuerza la compresión.
<code>bzip2 --help</code>	Muestra la ayuda de bzip2.

**bunzip2.**- El comando “bunzip2” es un descompresor de archivos en formato bz2.

COMANDO	DESCRIPCIÓN
bunzip2 x.txt.bz2	Descomprime el archivo x.txt.bz2 en el directorio actual.
bunzip2 --help	Muestra la ayuda de bunzip2.

**tar.-** El comando “tar” “Tape ARchiver (Archivador de Cinta)” es una utilidad de empaquetado de archivos y directorios en un solo archivo, no ofrece ningún tipo de compresión.

Para mayor información de las opciones soportadas por el comando “tar”, consulte su ayuda respectiva.

COMANDO	DESCRIPCIÓN
tar cvf docs.tar x1/	Empaqueta el contenido del directorio x1 como x.tar sin comprimir.
tar cvfz x.tar.gz x1/	Compresión del empaquetado x.tar usando gzip.
tar cvf x.tar x1/ gzip x.tar	El comportamiento es similar al comando anterior.
tar cvfj x.tar.bz2 x1/	Compresión del empaquetado x.tar usando bzip2.
tar xvf docs.tar	Desempaqueta el contenido de docs.tar en el directorio actual.
tar xvzf x.tar.gz	Desempaqueta el archivo comprimido con gzip en el directorio actual.
tar xvfj x.tar.bz2	Desempaqueta el archivo comprimido con bzip2 en el directorio actual.

A continuación se muestran algunos comandos usados con frecuencia en un Servidor de producción GNU/Linux.

**iftop.-** El comando “iftop” permite realizar estadísticas del consumo del ancho de banda dentro de una red, se hace referencia a una interfaz de red Ethernet (en

GNU/Linux cada interfaz de red se identifica de la siguiente manera: eth0, eth1, eth2, etc.).

COMANDO	DESCRIPCIÓN
<code>iftop -i eth1</code>	Muestra la estadística de tráfico en la tarjeta eth0.

**ssh.-** El comando “ssh”, permite establecer un Shell remoto seguro para administrar un servidor, es recomendable tener bien protegido los puertos de nuestro servidor a fin de evitar ataques externos. Al momento de conectarnos a un equipo se solicita la dirección IP, un puerto de acceso, luego de ello se solicita un Login y un password.

COMANDO	DESCRIPCIÓN
<code>ssh -p 22 201.218.45.198</code>	Establece una sesión segura en un servidor remoto, utilizando el puerto 22.

**wget.-** El comando “wget”, permite descargar un archivo que se encuentre en nuestro servidor GNU/Linux desde una máquina remota conectada a través de ssh o telnet.

COMANDO	DESCRIPCIÓN
<code>wget 201.218.45.197/firewall.txt</code>	Descarga el archivo firewall.txt desde el servidor remoto.

**vi.-** El comando “vi”, permite la edición de documentos de texto dentro de la consola de GNU/Linux, existe el “vim” que es una versión moderna de este editor de texto.

COMANDO	DESCRIPCIÓN
<code>vi firewall.txt</code>	El editor “vi” abre el archivo firewall.txt en modo lectura y escritura si existe permisos, caso contrario solo en modo lectura.

**ping.-** El comando “ping”, permite el envío de paquetes de datos a equipos que estén en la red local o en Internet, para comprobar la conexión de red.

COMANDO	DESCRIPCIÓN
ping 192.168.172.100	Envía paquetes de datos a la dirección IP.
ping www.xion.org	Envía paquetes de datos a la dirección Web.
ping 192.168.172.21 -c 100	Envía un paquete de 100 datos, muestra la estadística de recepción y pérdidas.

**who is.-** El comando “who is”, permite consultar a servidores especializados los nombres a partir de una dirección IP. Esto permite averiguar si una IP sospechosa detectada en nuestra red corresponde a máquinas externas que pueden robar información de nuestra empresa.

COMANDO	DESCRIPCIÓN
who is 172.31.200.13	Se consulta el nombre de la dirección IP 172.31.200.13

**service.-** El comando “service”, en sistemas operativos clones de Red Hat Linux permite el arranque, parada, reinicio y estado de un servicio específico. Como ejemplo tenemos el servicio “named”.

Los comandos posibles a ejecutar son los siguientes:

COMANDO	DESCRIPCIÓN
service named start	Arranca el servicio named.
service named status	Muestra el estado del servicio named.
service named restart	Reinicia el servicio named.
service named stop	Detiene el servicio named.

**ntop.-** El comando “ntop”, permite realizar un escaneo de la red para ver el comportamiento de envío y peticiones de paquetes de todos los equipos de la red. Se hace referencia a una interfaz de red.

COMANDO	DESCRIPCIÓN
ntop -i eth1	Escanea la interfaz de red eth0.

**rpm.-** El comando “rpm”, dependiendo de sus parámetros permite la instalación de un nuevo paquete de software o la consulta de la existencia del mismo dentro del sistema operativo GNU/Linux. Este comando es propio de distribuciones basadas en Red Hat Linux.

COMANDO	DESCRIPCIÓN
rpm -Uvh netmrg-0.18.2-1.el4.i386.rpm	Instala un paquete de software.
rpm -q xmms	Consulta en la Base de Datos de programas instalados si existe el nombre del reproductor de audio xmms.

**Nota:** Para mayor información sobre las opciones de los comandos consulte las páginas de manuales de su Distribución GNU/Linux preferida, use el comando: `man Nombre_Comando`.



## Gestión de Usuarios en GNU/Linux.

GNU/Linux es un sistema multiusuario por naturaleza. Una posibilidad de entrada de diferentes usuarios al mismo sistema es por medio de las consolas virtuales.

Hay diferentes tipos de usuario, tenemos tres tipos: los administradores, usuarios comunes y usuarios del sistema.

### Súper Usuario o root.

El súper-usuario o administrador tiene completo control sobre el sistema, él puede guardar, configurar, borrar, o hacer lo que quiera con lo que haya dentro del Sistema Operativo GNU/Linux, se debe usar este usuario solo en casos necesarios.

### Usuario Normal o user.

Un usuario común, sólo puede trabajar con limitados recursos dentro del Sistema Operativo GNU/Linux, esto es porque hay información y cosas que dichos usuarios no deben tener a su alcance por medidas de seguridad, sus cuentas se crean en el directorio /home.

### Usuarios del Sistema o user systems.

Un usuario del sistema no posee un directorio de trabajo como el de un usuario común, este tipo de usuario es usado por programas o servicios en GNU/Linux como son Apache, MySQL, etc.

### Permisos.

Indican la autoridad que se puede ejercer ante un archivo o directorio, o ante otro recurso. Para ver cuales son los permisos de cada archivo o directorio se ejecuta "ls -l", en la Figura XGL 1 se puede observar su ejecución.

```

root@xion:~
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
[root@xion root]# ls -l
total 8624
-rw-r--r--  1 root  root      1247 may 28 14:06 anaconda-ks.cfg
-rw-r--r--  1 root  root        44 jun  3 19:03 eltipodelapatata.wav
-r-xr-xr-x  1 root  root  4599808 jul 17 2006 fer.doc
-rw-r--r--  1 root  root   21580 may 28 14:05 install.log
-rw-r--r--  1 root  root   3974 may 28 14:04 install.log.syslog
-rwxr-xr-x  1 root  root  527341 ene  5 2006 LinuxAvanzado.pdf
-rwxr-xr-x  1 root  root 3586134 feb 19 19:42 maia.mp3
-rwxr-xr-x  1 root  root    44 jun  2 18:33 redhat-sample.wav
drwxr-xr-x  2 root  root   4096 mar 25 16:28 RHEL
    
```

Figura XGL 1 - Ejecución del comando ls -l.

A continuación se describe lo mostrado en la Figura XGL 1.

1. El campo (drwxr-xr-x) corresponde a los permisos del archivo o directorio que estamos mirando. Entendemos esta línea de la siguiente manera: los primeros 4 parámetros corresponden al dueño del archivo; los siguientes 3 parámetros corresponden al grupo o grupos de usuarios al que pertenece el usuario dueño del archivo; y los últimos tres parámetros corresponden a cualquier otro usuario.
  - ✚ La letra "r" (read) quiere decir que el usuario solo tendrá la posibilidad de leer el contenido del archivo, pero no modificarlo ni ejecutarlo.
  - ✚ La letra "w" (write) le da al usuario permiso de escritura, es decir que el usuario tiene la posibilidad de modificar de cualquier manera el archivo o directorio.
  - ✚ La letra "x" (execute) le da al usuario la posibilidad de ejecutar el archivo o directorio al que nos refiramos, con ejecutar un directorio entendemos que podemos acceder a él.

A continuación se muestra unos ejemplos que describen los permisos que pueden tener un archivo o un directorio.

PERMISOS	DESCRIPCIÓN
-rwx-----	El dueño del archivo o directorio podrá leer el contenido (r), modificar (w) y ejecutar (x) el mismo. Los demás usuarios o participantes del mismo grupo no podrán hacer nada con el archivo.
-rwx--x--x	El dueño, el grupo y los demás usuarios pueden ejecutar el archivo al que se refieren. Además el dueño también puede leer y escribir el mismo archivo.

2. El campo (1) corresponde al número de enlace que tiene un archivo (1) o un directorio (2), este número cambiara dependiendo de los enlaces que se añadan al archivo o directorio.
3. El campo (root) corresponde al ID del Propietario de un archivo o directorio, en este caso es el usuario "root".
4. El campo (root) corresponde al ID del Grupo al que pertenece un usuario, en este caso el grupo es "root".
5. Los campo (1247) corresponde al tamaño del archivo o directorio.
6. El campo (may) corresponde al Mes de creación del archivo o directorio.
7. El campo (28) corresponde al Día de creación del archivo o directorio.
8. El campo (14:06) corresponde a la hora de creación del archivo o directorio.
9. El campo (anaconda-ks.cfg) corresponde al nombre del archivo o directorio, en este caso es un archivo.

## Cambio de Permisos.

Notifico que los permisos pueden ser cambiados solamente por el superusuario "root" o por el dueño de un archivo o directorio.

**chmod.-** El comando "chmod" permite cambiar los permisos de un archivo o directorio, usando notación octal o a través de caracteres como se muestra en la siguiente tabla.

TABLA DE PERMISOS DE UN ARCHIVO O DIRECTORIO EN GNU/LINUX.

----/---/---	
U (User)   G (Group)   O (Others)	
<b>U</b>	Corresponde al dueño del archivo, User.
<b>G</b>	Corresponde al grupo, Gruop.
<b>O</b>	Corresponde a los otros usuarios, Other.

Se usa la tabla de permisos anterior combinadas con “chmod”, el signo (+) otorga un permiso, el signo (-) quita un permiso. A continuación se presenta algunos ejemplos:

COMANDO	DESCRIPCIÓN
chmod o-x archivo	Cambiara -rwxr-xr-x a -rwxr-xr--
chmod o+w archivo	Cambiara -rwxr-xr-x a -rwxr-xrwx
chmod g+w archivo	Cambiara -rwxr-xr-x a -rwxrwxr-x
chmod u-x archivo	Cambiara -rwxrwxrwx a -rw-rwxrwx
chmod 700 directorio	Cambiara drwxr-xr-x a drwx-----
chmod 421 directorio	Cambiara drwxr-xr-x a dr---w---x
chmod 625 archivo	Cambiara -rwxr-xr-x a -rw--w-r-x

## Cambio de Dueños.

Para saber quien es el propietario y el grupo de un archivo o directorio debemos observar el segundo y el tercer parámetro del comando “ls -l” respectivamente.

**chown.-** El comando “chown” permite cambiar tanto el dueño como el grupo de cada archivo o directorio. Vea la Figura XGL 2.

```

root@xion:~/xion
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
[root@xion xion]# ls -l
total 28
drwxr-xr-x  2 root    root      4096 jun  4 09:57 docs
drwxr-xr-x  2 root    root      4096 jun  4 09:57 links
-rwxr-xr-x  1 root    root     17546 may  1 17:14 xion.png
[root@xion xion]# chown invitado xion.png
[root@xion xion]# ls -l
total 28
drwxr-xr-x  2 root    root      4096 jun  4 09:57 docs
drwxr-xr-x  2 root    root      4096 jun  4 09:57 links
-rwxr-xr-x  1 invitado root     17546 may  1 17:14 xion.png
[root@xion xion]# █

```

Figura XGL 2 – Ejecución del comando “chown”.

## Cambio de Atributos de un Archivo.

Se usan los comandos “chattr” para cambiar y “lsattr” listar los atributos de un archivo.

**chattr.**- El comando “chattr” permite modificar los atributos de un archivo.

OPCIÓN	DESCRIPCIÓN “CHATTR”
<b>A</b>	No actualiza el archivo “atime”, es útil para limitar las entradas y salidas al disco en un equipo portátil o sobre una conexión NFS.
<b>a</b>	Abre el archivo en modo de actualización, esta opción la establece el usuario “root”.
<b>c</b>	El archivo se comprime en disco automáticamente por el núcleo del sistema.
<b>d</b>	Marca el archivo para que no sea candidato a ser usado por el programa “dump”.
<b>i</b>	El archivo no puede ser modificado, borrado, ni renombrado. Tampoco se podrá crear ningún enlace hacia él, ni se podrá escribir ningún dato en él.
<b>s</b>	Cuando se borre el archivo, sus bloques se rellenarán con ceros y se volverán a grabar en el disco.
<b>S</b>	Cuando se modifique el archivo, los cambios serán grabados instantáneamente en el disco.
<b>u</b>	Cuando se borre el archivo, se salvará su contenido.

**lsattr.**- El comando “lsattr” permite ver los atributos modificados por el comando “chattr”.

A continuación se presenta un ejemplo que combina los comandos “chattr” y “lsattr”.

## COMANDOS A EJECUTAR EN EL TERMINAL Y SUS SALIDAS CORRESPONDIENTES

```
[root@xion xion]# lsattr texto.txt
----- texto.txt
[root@xion xion]# chattr +c texto.txt
[root@xion xion]# chattr +d texto.txt
[root@xion xion]# chattr +s texto.txt
[root@xion xion]# lsattr texto.txt
s-c--d- texto.txt
[root@xion xion]# chattr -d texto.txt
[root@xion xion]# lsattr texto.txt
s-c---- texto.txt
[root@xion xion]#
```

## Creación de Cuentas de Usuarios en GNU/Linux.

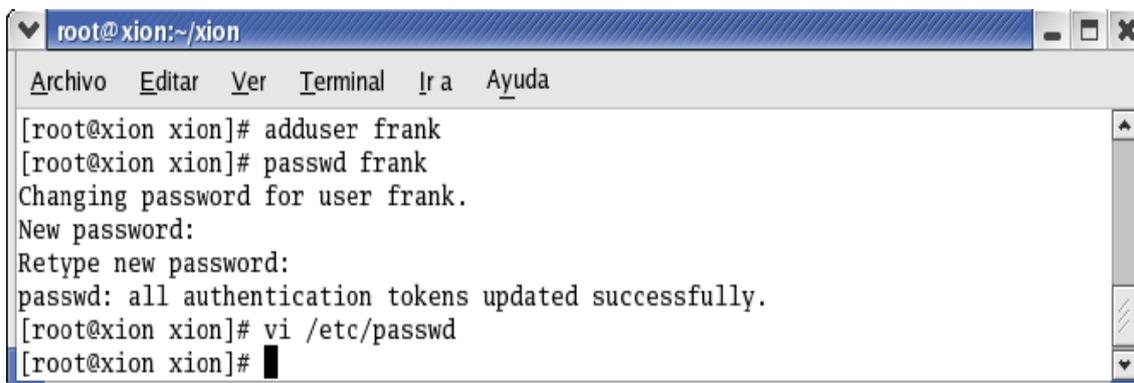
Podemos crear un nuevo usuario en el sistema. Hay dos maneras de crear un nuevo usuario:

1. Usando el comando “useradd” o “adduser”.
2. Usando el administrador de usuarios localizado en el menú **Configuración del Sistema/Usuarios & Grupos**.

Para añadir un usuario nuevo ejecutamos el comando “adduser” seguido de un nombre, luego ejecutamos el comando “passwd” para establecer una clave de acceso y el nombre de usuario nuevo.

La ejecución de los comandos necesarios se muestra en la Figura XGL 3.

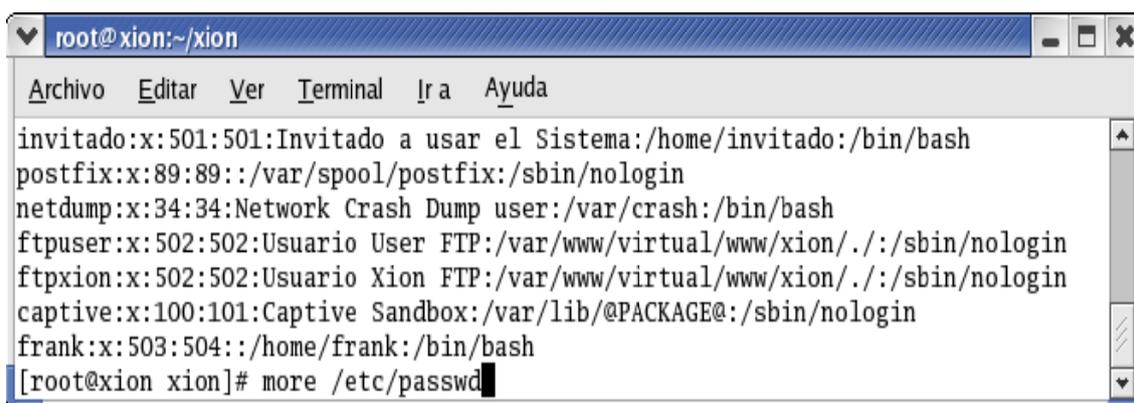
**Nota:** Al momento de escribir la contraseña no se visualiza nada en el terminal por cuestiones de seguridad, tenga en cuenta que GNU/Linux distingue los caracteres en mayúscula (ejemplo: CLAVE) de los caracteres en minúscula (ejemplo: clave).



```
root@xion:~/xion
Archivo  Editar  Ver  Terminal  Ira  Ayuda
[root@xion xion]# adduser frank
[root@xion xion]# passwd frank
Changing password for user frank.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@xion xion]# vi /etc/passwd
[root@xion xion]#
```

Figura XGL 3 – Ejecución del comando adduser y passwd.

En la Figura XGL 4, se puede ver el contenido del archivo /etc/passwd que contiene el nuevo usuario y su password.



```
root@xion:~/xion
Archivo  Editar  Ver  Terminal  Ira  Ayuda
invitado:x:501:501:Invitado a usar el Sistema:/home/invitado:/bin/bash
postfix:x:89:89:./var/spool/postfix:/sbin/nologin
netdump:x:34:34:Network Crash Dump user:/var/crash:/bin/bash
ftpxion:x:502:502:Usuario Xion FTP:/var/www/virtual/www/xion./:/sbin/nologin
captive:x:100:101:Captive Sandbox:/var/lib/@PACKAGE@:/sbin/nologin
frank:x:503:504:./home/frank:/bin/bash
[root@xion xion]# more /etc/passwd
```

Figura XGL 4 – Ejecución del comando more.

Para poder leer o modificar el contenido del archivo /etc/passwd, se debe ingresar al sistema como “root” o alguna cuenta de usuario con privilegios.

De la Figura XGL 4, tomamos la siguiente línea:

```
frank:x:503:504:./home/frank:/bin/bash
```

Esta simple línea da al sistema toda la información que necesite para que el usuario pueda entrar tranquilamente.

Analicemos esa línea:



Línea del archivo `/etc/passwd` para la cuenta de un usuario: 1 2 3 4 5 6 7

<b>1</b>	Nombre del usuario actual (frank).
<b>2</b>	El password generalmente encriptado representado por una X.
<b>3</b>	Número de identificación del usuario UID, (user id=503).
<b>4</b>	Número de identificación del grupo al que pertenece, GID (group id=504)
<b>5</b>	El verdadero nombre del usuario (por el momento no asignado).
<b>6</b>	Directorio home ( <code>/home/frank</code> ) del usuario. Este es el directorio donde aparecerá el usuario apenas sé loguee. Es el único directorio donde los usuarios comunes podrán grabar cosas.
<b>7</b>	Interprete de comandos ( <code>/bin/bash</code> ). Ejecuta las instrucciones que recibe desde es teclado.

## Estructura de Directorios en GNU/Linux.

Dependiendo de las distribuciones pueden variar algunos nombres de directorios o existir algunos nuevos. Como ejemplo se muestra la estructura de directorios en Red Hat Linux Enterprise AS 3.0 en la Figura XGL 5.

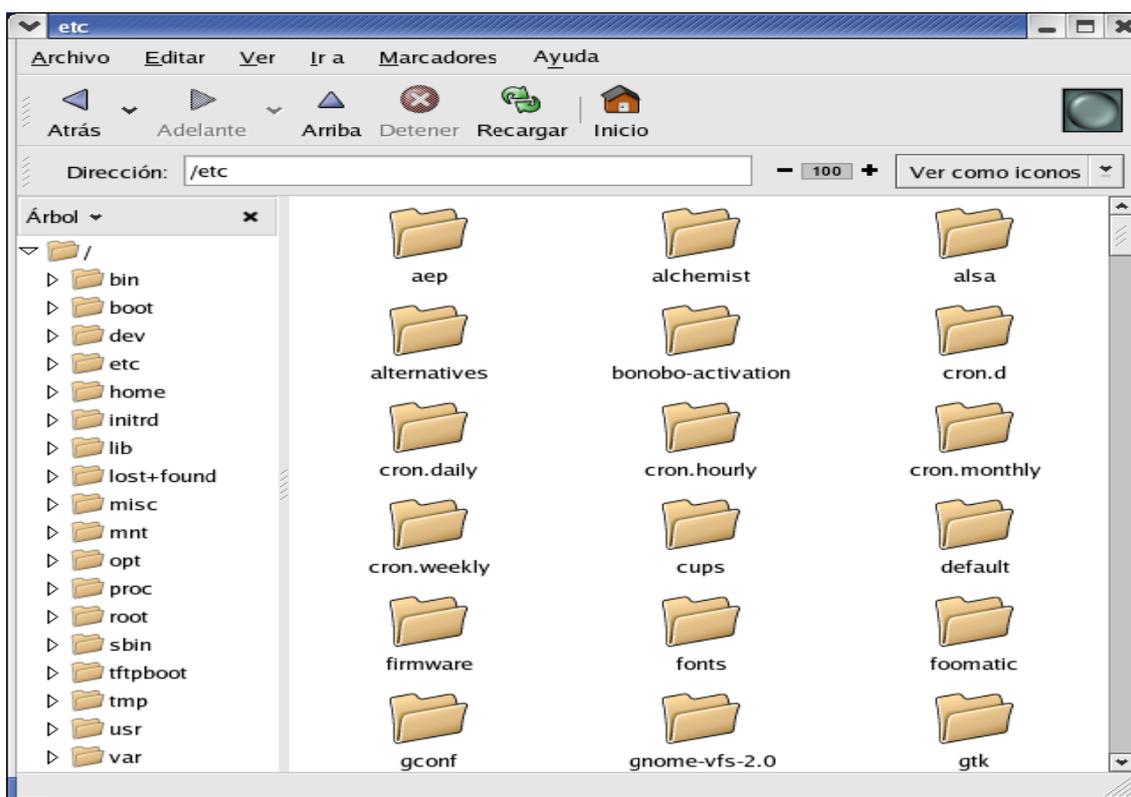


Figura XGL 5 – Estructura de Directorios en RHEL.

### Directorios.

**/bin:** Es la abreviación de “binaries” (ejecutables) se guardan los programas básicos y esenciales para el funcionamiento del sistema como son (ls, cp, clear, more, etc) ejecutados por todos los usuarios de GNU/Linux.

**/dev:** Es la abreviación de “device” (dispositivo) aquí se guardan los controladores de dispositivos, se usan para acceder a los dispositivos del sistema y a sus recursos como discos rígidos, módems, memoria, mouse, etc.



**/etc:** Se guardan muchos de los archivos de configuración del sistema, y de los programas, por ejemplo: `/etc/passwd` guarda información de cada uno de los usuarios como por ejemplo el directorio de inicio del usuario, el intérprete de comandos que usara, su UID, GID, etc.

**/sbin:** Se guardan muchos programas que son sólo usados por el administrador del sistema "root", por ejemplo: `exit`, `halt`, `reboot`, `setup`, etc.

**/home:** Aquí se encuentran generalmente los directorios de inicio (home directory) de los distintos usuarios. Aunque no es obligatorio esto, se usa ya por costumbre para ello.

**/lib:** Acá están las imágenes de las librerías compartidas, estos archivos son usados por muchos programas y se guardan aquí para que cada programa no tenga que tener cada rutina en sus ejecutables. Esto ahorra espacio en disco.

**/proc:** Es un sistema de archivo virtual. Contiene archivos que residen en memoria pero no en el disco rígido. Hace referencia a los programas actuales que se están corriendo en el Sistema Operativo GNU/Linux.

**/tmp:** El directorio `tmp` se guardan archivos de carácter temporal, es decir, si un programa necesita mantener cierta información guardada, pero que no será necesaria en otro momento, guardara esa información en el directorio `tmp` que se vaciara en cuanto el sistema sea reiniciado.

**/usr:** Este directorio es muy importante porque contiene más subdirectorios que contienen a su vez, muchos archivos importantes, programas, y archivos de configuración del sistema. La mayoría de las cosas que hay en el directorio `usr` son opcionales, es decir, el sistema no las necesita para funcionar correctamente.

**/usr/X11:** Contiene el sistema X Windows que es el entorno gráfico de los sistemas UNIX. El X Windows maneja las ventanas y programas gráficos en la modalidad Cliente/Servidor.

**/usr/bin:** Contiene otros programas que no se encuentran en el directorio `/bin`.

**/usr/etc:** Tiene algunos archivos de configuración y programas del sistema. Estos archivos, a diferencia de los del directorio `/etc` no son esenciales para el correcto funcionamiento del sistema, aunque puede que algunos si lo sean.



**/usr/include:** Contiene archivos que utiliza el compilador de C. Son esenciales para compilar casi cualquier programa hecho en C.

**/usr/lib:** Contiene las librerías equivalentes 'stub' y 'static' a los archivos de /lib. Muchos programas, además, guardan archivos de configuración en /usr/lib.

**/usr/local:** Contiene al igual que /usr una gran cantidad de programas. Este directorio difiere mucho entre cada sistema UNIX o GNU/Linux.

**/usr/man:** Aquí se encuentran la mayoría de las páginas del manual de cada programa. Para acceder a los manuales se utiliza el comando man seguido del nombre del programa.

**/usr/src:** Contiene el código fuente (programas por compilar) de muchos programas.

**/usr/src/linux:** Contiene el código fuente del núcleo, que puede ser recompilado nuevamente por el administrador. (No aconsejable para usuarios inexpertos).

**/var:** Aquí están los programas que cambian de vez en cuando o a menudo de tamaño.

**/var/adm:** Contiene archivos que son de incumbencia para el administrador del sistema, más que nada históricos, donde se loguea (graba) todos los errores, entradas, salidas, etc.

**/var/www/html:** Contiene las páginas html que usamos en un servidor Web local o de producción que se ejecuta con el servidor Web Apache.

## Dispositivos.

**hd:** hda1 será el disco rígido IDE (hd), primario (a), y la primera partición (1).

**fd:** Los archivos que empiecen con las letras 'fd' se referirán a los controladores de las disqueteras: fd0 sería la primera disquetera fd1 sería la segunda.



**ttyS - cua:** Los dispositivos “ttyS” y “cua” se usan para acceder a los puertos serie como por ejemplo, ttyS0 seria el puerto com1 en MS-DOS o Microsoft Windows. Los dispositivos “cua” son 'callout' por eso se usan en conjunción con el MODEM.

**sd:** Son los dispositivos SCSI. Su uso es muy parecido al del hd.

**lp:** Son los puertos paralelos, lp0 es equivalente a LPT1 en MS-DOS o Microsoft Windows.

**null:** Este es usado como un agujero negro, ya que todo lo que se dirige allí desaparece.

**tty:** Los tty hacen referencia a cada una de las consolas virtuales, tty1 será la primera consola virtual, tty2 la segunda, etc.

**pty:** Son pseudo terminales, proporcionan una terminal a sesiones remotas, esto quiere decir que si la máquina esta conectada en red cualquiera que permita Telnet para entrar al sistema usara uno de estos dispositivos.

## Sistema de Booteo en GNU/Linux.

El sistema de booteo (inicio) de toda Distribución GNU/Linux cumple un cierto orden donde primero se carga el kernel o núcleo del sistema, luego de esto se carga el programa init que pone en funcionamiento el archivo más importante de inicio inittab<sup>1</sup> después se cargan todos los módulos, dispositivos, demonios, etc., y una vez concluido esto, podemos pasar a digitar nuestra cuenta de usuario y password para acceder al Sistema Operativo.

InitTab distingue multiples run-level, los cuales tienen cada uno sus procesos a inicializar. Hay run-levels validos del 0 al 6. Paso a explicar para que sirvan cada uno de ellos.

<i>RUN-LEVEL</i>	<i>DESCRIPCIÓN</i>
0	Se utiliza para apagar el sistema definitivamente, sin intenciones de reiniciar.
1	Inicia el sistema en modo 'monousuario', no se puede usar las aplicaciones gráficas sólo un limitado conjunto de comandos, podemos recuperar la contraseña o recuperar el sistema GNU/Linux después de una caída.
2	Inicia el sistema en modo multiusuario sin soporte para NFS (Network File System o Sistema de Archivos de Red).
3	Inicia el sistema en modo multiusuario.
4	No utilizado en la Distribución Red Hat Linux y sus clones.
5	Inicia el sistema directamente en el administrador de ventanas X Window. Este run-level es muy práctico para los que acostumbran a trabajar todo el tiempo en X Window y no utilizan la consola casi nunca.
6	Se utiliza al momento de reiniciar el sistema.

<sup>1</sup> **Inittab.**- localizado en el directorio /etc, que describe que otros archivos deben cargarse para poner en marcha los distintos programas en memoria, monten las unidades, etc.

Se muestra algunos ejemplos en la siguiente tabla, algunos comandos necesitaran privilegios del usuario "root" si existen muchos usuarios conectados al equipo.

COMANDO	DESCRIPCIÓN
<b>init 0</b>	Permite apagar el sistema. Su equivalente es el comando "halt".
<b>init 6</b>	Permite reiniciar el sistema. Su equivalente es el comando "reboot".
<b>shutdown</b>	Permite apagar el sistema.
<b>shutdown -r</b>	Equivalente al comando "reboot".
<b>shutdown -h</b>	Equivalente al comando "halt".
<b>shutdown -c</b>	Permite cancelar "cancel" las opciones anteriores.
<b>shutdown -r +t3</b>	Reiniciar el equipo en tres minutos.
<b>shutdown -r now</b>	Reiniciar el equipo ahora "now".
<b>shutdown -h +t5</b> "Mensaje"	Apagar el equipo en cinco minutos, mostrando un mensaje para los usuarios.

## Sistemas de Archivos.

Dependiendo de la Distribución GNU/Linux que utilicemos se soportarán o no los sistemas de archivos, un caso de ellos es el sistema ntfs por lo general no es soportado; se debe actualizar el kernel para que lo soporte.

## Tipos de Sistemas de Archivos.

A continuación se describen algunos tipos de sistemas de archivos más conocidos:

**ext - Extended File System:** Sistema de archivos nativo original de las primeras Distribuciones GNU/Linux.

**ext2 - Second Extended File System:** Es el sistema de archivos común en GNU/Linux, antes se usaba el 'ext' que fue reemplazado por este.

**minix - Minix File System:** Sistema de archivos del Sistema Operativo Minix.

**xiafs - Xia File System:** Parecido al ext2. Casi no se lo utiliza actualmente.

**msdos - MS-DOS File System:** Sistema de archivos de MS-DOS.

**nfs - Network File System:** Sistema de archivos de Red.

**iso9660 - ISO 9660 File System:** Sistema de archivos usado comúnmente para los CD-ROMs.

**xenix - Xenix File System:** Sistema de archivos del Sistema Operativo Linux Xenix de Microsoft, no existe en la actualidad.

**swap:** Sistema de archivos de intercambio.

**hpfs - HPFS File System:** Nos da acceso de solo lectura a las unidades comprimidas con DoubleSpace.

**vfat - FAT File System:** Accede a particiones de FAT16 y FAT32 de Microsoft Windows.

**ntfs – New Technolgy File System:** Sistema de archivos seguro usado por las últimas versiones de Sistemas Operativos Clientes y Servidores Microsoft Windows.

## Todo es un Archivo.

Un disco rígido, una partición en un disco rígido, un puerto paralelo, una conexión a un sitio web, una placa Ethernet, todos estos son archivos. Incluso los directorios son archivos. Note que aquí por tipo de archivo no nos referimos al tipo de contenido de un archivo: para GNU/Linux y cualquier sistema Unix, un archivo, ya sea una imagen GIF, un archivo binario o lo que sea, sólo es un flujo de bytes. Diferenciar a los archivos de acuerdo a su contenido es algo dejado a las aplicaciones.

## Los Diferentes Tipos de Archivos.

Al hacer un `ls -l`, el carácter antes de los derechos de acceso identifica el tipo de un archivo, los archivos regulares (-) y los directorios (d). También puede encontrarse con estos otros tipos si se desplaza por el árbol de archivos y lista el contenido de los directorios:

### Archivos de Modo Carácter.

Estos archivos son o bien archivos especiales del sistema (tal como `/dev/null`, que ya hemos visto), o bien periféricos (puertos serie o paralelo), que comparten la particularidad de que su contenido (si es que tienen alguno) no se conserva en memoria. Dichos archivos se identifican con la letra 'c'<sup>2</sup>.

### Archivos de Modo Bloque.

Estos archivos son periféricos, y, a diferencia de los archivos de modo carácter, su contenido, está "bufereado"<sup>3</sup>. Los archivos que entran en esta categoría son, por

<sup>2</sup> **Archivos Modo Carácter.**- Se dice de ellos "no-bufereados", que proviene del inglés `unbuffered`.

<sup>3</sup> **Archivos Modo Bloque.**- Conservado en memoria.

ejemplo, los discos rígidos, las particiones de un disco rígido, las unidades de disquete, las unidades de CD-ROM y así sucesivamente.

Los archivos `/dev/hda`, `/dev/sda5` son un ejemplo de archivos de modo bloque. En la salida de `ls -l`, estos están identificados por la letra 'b'.

### Vínculos Simbólicos.

Estos archivos son muy comunes, y se usan ampliamente en el procedimiento de inicio del Sistema Operativo GNU/Linux. Su propósito es vincular archivos de forma simbólica, lo que significa que dichos archivos pueden o no apuntar a un archivo existente. Generalmente se los conoce como "soft links" (en inglés), y están identificados por una 'l'.

### Tuberías Nombradas.

En caso que se lo pregunte, sí, estos son muy similares a las tuberías usadas en los comandos del shell, pero con la particularidad que estas, en realidad, tienen nombre. Sin embargo, son muy raras, y es muy poco probable ver en el árbol de archivos. Se identifica por la letra 'p'.

### Sockets.

Este es el tipo de archivo para todas las conexiones de red. Pero sólo unos pocos tienen nombre. Más aun, hay distintos tipos de sockets y sólo se puede vincular uno. Dichos archivos se identifican con la letra 's'.

### I-nodos.

Los i-nodos son la parte fundamental de cualquier sistema de archivos GNU/Linux. La palabra i-nodo es una abreviación de Information NODE (NODO de Información).

Los i-nodos se almacenan en el disco en una tabla de i-nodos. Existen para todos los tipos de archivos que se pueden almacenar en un sistema de archivos, y esto incluye a los directorios, las tuberías nombradas, los archivos de modo carácter, y así



sucesivamente. Los i-nodos también son la forma en la que un sistema GNU/Linux identifica de forma unívoca a un archivo.

En GNU/Linux a un archivo no se identifica por su nombre, sino por un número de i-nodo<sup>4</sup>. La razón para esto es que un mismo archivo puede tener varios nombres, o incluso ninguno. En GNU/Linux, un nombre de archivo es simplemente una entrada en un i-nodo de directorio. Tal entrada se denomina vínculo.

---

<sup>4</sup> **I-node.**- Los números de i-nodo son únicos para cada sistema de archivos, puede existir un "i-nodo" con el mismo número en otro sistema de archivos. Esto nos lleva a la diferencia entre i-nodos "en disco" e i-nodos "en memoria". Aunque los i-nodos "en disco" pueden tener el mismo número si se encuentran en sistemas de archivo diferentes, los i-nodos "en memoria" tienen un número único a través de todo el sistema. Una solución para obtener la unicidad es, por ejemplo, hacer un hash de el número de i-nodo "en disco" contra el identificador del dispositivo de bloques.

## Servidor Samba SMB (Server Message Block).

Un equipo con GNU/Linux instalado y otros equipos en red en una oficina con alguna versión de Microsoft Windows son conectados con el servidor Samba para compartir archivos o impresoras.

Samba es un conjunto de programas, originalmente creados por Andrew Tridgell y actualmente mantenidos por The Samba Team, bajo la Licencia Publica GNU, y que implementan en sistemas basados sobre Unix el protocolo Server Message Block (Protocolo SMB). Este es a veces referido como Common Internet File System (CIFS), LanManager o protocolo NetBios. Sirve como reemplazo para Microsoft Windows NT, Warp, NFS (Network File System) o servidores Novell Netware.

### Software Requerido.

Necesitara tener instalado los siguientes paquetes: samba, samba-client, samba-common.

Realice una consulta al sistema si se encuentran instalados estos paquetes, utilizando el siguiente comando:

#### Comando a ejecutar en la Consola de GNU/Linux

```
rpm -q samba samba-client samba-common
```

Este comando se probó en Red Hat Enterprise Linux AS 3.0, la salida correspondiente es la siguiente:

#### Resultado de la consulta del comando rpm -q

```
samba-3.0.0-14.3E  
samba-client-3.0.0-14.3E  
samba-common-3.0.0-14.3E
```

Si por alguna razón no se encuentra alguno de estos programas, introduzca el CD de instalación, monte este y ejecute el siguiente comando para instalar el paquete que falte:

#### Comando a ejecutar en la Consola de GNU/Linux

```
rpm -Uvh /mnt/cdrom/RedHat/RPMS/paquete_que_le_falte
```

## Procedimientos.

Definamos primero los parámetros necesarios, sería el NetBios con el que nos vería el grupo de máquinas Windows, el grupo al que pertenecemos y el rango de direcciones IP a las que se permitirá acceder hacia la máquina GNU/Linux. Para fines prácticos el nombre NetBios debe tener un máximo de 11 caracteres. Normalmente tomaremos como en referencia un nombre corto de la máquina o el nombre corto que se asignó como alias a la interfaz de red. Este lo establecemos en el fichero `/etc/samba/lmhosts` en donde encontraremos lo siguiente:

#### Contenido del archivo `/etc/samba/lmhosts`

```
127.0.0.1 localhost
```

Debemos añadir el nombre que hayamos elegido asociado a la dirección IP que se tenga dentro de la red local. Adicionalmente se podrá añadir también el nombre y dirección IP del resto de las máquinas que conformen la red local. Haga la separación de espacios con un tabulador.

#### Modificación del contenido del archivo `/etc/samba/lmhosts`

```
127.0.0.1 localhost  
192.168.1.5 maquinainux  
192.168.1.6 maquinawindows  
192.168.1.7 gerencia
```

Una vez hecho esto se necesitara ahora configurar los parámetros necesarios en `/etc/samba/smb.conf`. Empezaremos por establecer el grupo de trabajo editando la línea `workgroup`, de este modo: `Workgroup MIGRUPO`

Después establecemos de acuerdo al fichero `/etc/samba/lmhosts` en la línea del parámetro `netbios name`:

#### Modificación del contenido del archivo `/etc/samba/smb.conf`

```
netbios name = maquinainux
```

En el fichero `/etc/samba/smb.conf`, encontrará no solo las opciones que requieren editarse, sino también un valioso instructivo que podría consultar mas adelante para hacer algunos ajustes a la configuración. Dentro de este notara que la información que le será de utilidad viene comentada con un símbolo `#` (número) y los ejemplos; (punto y coma), siendo estos últimos los que tomaremos como referencia.

Se establece cierto nivel de seguridad especificando que rango de direcciones IP podrán acceder al servidor Samba, quitando los símbolos de comentarios y editando la línea `hosts allow`. Si nuestra red consiste en máquinas con dirección IP desde `192.168.1.1` hasta `192.168.1.254`, el rango de direcciones IP será `192.168.1.`, y este permitirá el acceso solo a dichas máquinas. Edite esta de manera que quede del siguiente modo: `hosts allow = 192.168.1. 127`.

Si queremos tener que evitar el registro de Windows en todas las máquinas de la red local, debemos quitar los comentarios de las siguientes líneas:

#### Modificación del contenido del archivo `/etc/samba/smb.conf`

```
encrypt password = Yes  
smb passwd file = /etc/samba/smbpasswd
```

Solo nos faltara configurar las impresoras y los directorios compartidos. Para tal fin, debemos añadir solo `public = yes` en la sección de impresoras, de forma quede de la siguiente forma:



### Modificación del contenido del archivo `/etc/samba/smb.conf`

```
[printers]
comment = El comentario que desee.
printable = Yes
browseable = No
public = Yes
```

Para los directorios o volúmenes que se compartirán, en el mismo fichero de configuración encontrara distintos ejemplos para distintas situaciones particulares.

En general puede utilizar el siguiente ejemplo que funcionará para la mayoría de los casos:

### Modificación del contenido del archivo `/etc/samba/smb.conf`

```
[Lo_que_quiera]
comment = Comentario que desee incluir.
path = /cualquier/ruta/que/desee/compartir
guest ok = Yes
```

Hecho todo lo anterior solo necesitara inicializar el demonio correspondiente a fin de que cargue los nuevos parámetros configurados.

Si iniciará Samba por primera vez ejecute siguiente comando:

### Comando a ejecutar en la Consola de GNU/Linux para iniciar el servicio SAMBA

```
/etc/rc.d/init.d/smb start
```

Si va a reiniciar el servicio de Samba, ejecute siguiente comando:

## Comando a ejecutar en la Consola de GNU/Linux para reiniciar el servicio SAMBA

```
/sbin/chkconfig --add smb
```

No olvide sincronizar las cuentas entre el servidor GNU/Linux y las estaciones con Microsoft Windows. Es decir si en una máquina con Windows ingresamos como el usuario "frank" con contraseña "frank2007" en el servidor GNU/Linux debe existir también dicha cuenta con ese mismo login y la misma contraseña.

Añada las cuentas con los comandos adduser y passwd, hágalo también con smbadduser y smbpasswd.

## Comandos a ejecutar en la Consola de GNU/Linux

```
/usr/sbin/adduser usuario_windows  
/usr/bin/passwd usuario_windows  
/usr/bin/smbadduser usuario_windows:usuario_windows
```

Si deseamos restringir las cuentas que se vayan a crear no puedan acceder a servicios distintos de Samba, como serian Telnet, SSH (Secury Shell), etc., es decir, que no se les permita hacer login al sistema, podemos utilizar la siguiente alternativa que solo permitirá acceso a Samba, pero impedirá que el usuario intente acceder al servidor y obtenga un shell:

## Comandos a ejecutar en la Consola de GNU/Linux

```
/usr/sbin/adduser -s /bin/false usuario_windows  
/usr/bin/smbadduser usuario_windows:usuario_windows  
/usr/bin/smbpasswd usuario_windows
```

A continuación se muestra un ejemplo del fichero de configuración del Servidor Samba en Linux Fedora Core 7.0.



## CONTENIDO DEL ARCHIVO DE CONFIGURACIÓN DE SAMBA

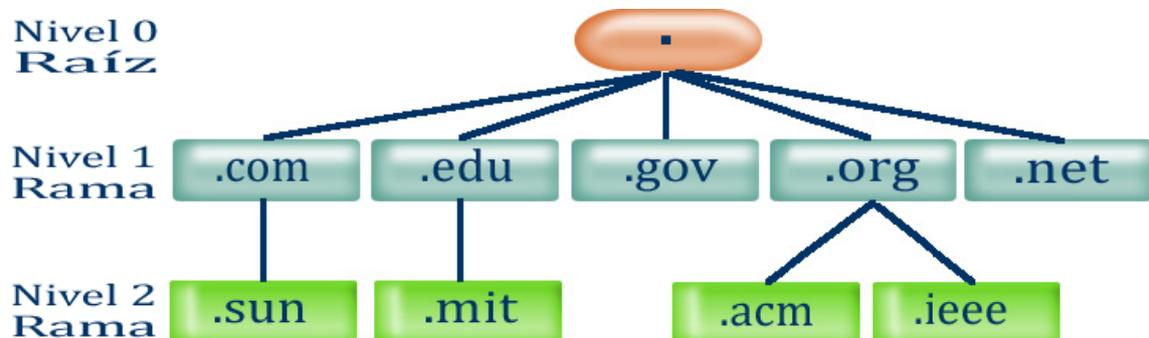
```
#SAMBA config file created using SWAT
#from localhost.localdomain (127.0.0.1)
#Date: 2007/05/21 17:39:05
#Global parameters
[global]
workgroup = XIONGROUP
netbios name = GNULINUX
server string = SAMBA Server
security = SHARE
encrypt password = Yes
smb passwd file = /etc/smbpasswd
username map = /etc/smbusers
log file = /var/log/samba/log.%m
max log size = 50
socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
```

## Servidor DNS (Domain Name Server).

El DNS permite resolver nombres de dominio y direcciones IP, permite ubicar hosts de redes lejanas, es más fácil recordar nombres en vez de cifras. Sobre todo cuando se trata de una cantidad de direcciones tan inmensa como la de Internet.

Las computadoras por otro lado trabajan perfectamente con cifras como la dirección IP. Lo que sucede cuando ingresamos a Internet colocando una dirección como <http://xionecuador.bravehost.com>, el navegador dirige una petición al Servidor de Dominio de su proveedor y este intenta resolver el nombre de dominio con la dirección IP correspondiente.

El DNS define un árbol de direcciones que empieza en una raíz, en este caso es “.”. De esta raíz surgen distintas ramificaciones, ampliamente conocidas por todos, que son: .com, .net, .org, .gov, .edu, .info, .es, .uk, .de, .mil, etc. Cada una puede tener subdominios, esto se aprecia en la Figura XGL 6.



Designed by: Franklin Coloma - Website: <http://xionecuador.bravehost.com>

Figura XGL 6 – Estructura de Árbol de Direcciones.

## Dominio.

Un dominio es el registro de un nombre comercial o marca en Internet, pueden tener varias terminaciones como ejemplo: .com, .edu, .gov, .org, .net, etc.

Existen dos tipos de dominios: dominios directos y dominios indirectos.

**Dominio Directo:** Proporciona para cada nombre una dirección IP.

**Dominio Indirecto o Inverso:** Proporciona para cada dirección IP un nombre. El dominio inverso se conoce también como dominio in-addr.arpa. Los elementos del dominio inverso son las direcciones de red construidas invirtiendo los números que lo componen, y terminando en in-addr.arpa.

**Ejemplo:** La red es 138.117.0.0, el dominio inverso es 117.138.in-addr.arpa.

## Configuración del Servidor DNS.

Archivos que intervienen en la configuración:

ARCHIVO	DESCRIPCIÓN
/resolv.conf	Necesario para la creación de dominios.
/named.conf	Necesario para la creación de zonas.
/named.local	Ejemplo sobre configuraciones de DNS.
/var/log/messages	Bitácoras, logs de ejecución de procesos del sistema.

Para la configuración del Servidor DNS seguiremos los siguientes pasos:

1. **Definir el Dominio.-** Bajo el directorio /etc el archivo resolv.conf y usando un editor de textos como "gedit" o "vi" crear el dominio, para el ejemplo consideramos xion.org y guardamos el archivo.

Contenido del archivo: **/etc/resolv.conf**

```
search xion.org
nameserver 192.168.10.1
nameserver 127.0.0.1
```

2. **Crear Zona.**- Bajo el directorio /etc modificamos el archivo named.conf donde se crearán dos zonas, las mismas que permitirán resolver: De nombre a IP (Resolución Directa) y de IP a nombre (Resolución Inversa).

El archivo root.hints debe incluirse en nuestro Servidor GNU/Linux, con ello mostramos la lista de servidores DNS capaces de cumplir con las peticiones de traducción de nuestro servidor DNS.

### Contenido del archivo root.hints

```
;  
; Puede haber comentarios aquí si ya tenías el fichero.  
; Si no lo hay, no te preocupes.  
;  
.           6D IN   NS    A.ROOT-SERVERS.NET.  
.           6D IN   NS    B.ROOT-SERVERS.NET.  
.           6D IN   NS    C.ROOT-SERVERS.NET.  
.           6D IN   NS    D.ROOT-SERVERS.NET.  
.           6D IN   NS    E.ROOT-SERVERS.NET.  
.           6D IN   NS    F.ROOT-SERVERS.NET.  
.           6D IN   NS    G.ROOT-SERVERS.NET.  
.           6D IN   NS    H.ROOT-SERVERS.NET.  
.           6D IN   NS    I.ROOT-SERVERS.NET.  
.           6D IN   NS    J.ROOT-SERVERS.NET.  
.           6D IN   NS    K.ROOT-SERVERS.NET.  
.           6D IN   NS    L.ROOT-SERVERS.NET.  
.           6D IN   NS    M.ROOT-SERVERS.NET.  
A.ROOT-SERVERS.NET. 6D IN   A    198.41.0.4  
B.ROOT-SERVERS.NET. 6D IN   A    128.9.0.107  
C.ROOT-SERVERS.NET. 6D IN   A    192.33.4.12  
D.ROOT-SERVERS.NET. 6D IN   A    128.8.10.90  
E.ROOT-SERVERS.NET. 6D IN   A    192.203.230.10  
F.ROOT-SERVERS.NET. 6D IN   A    192.5.5.241
```



G.ROOT-SERVERS.NET.	6D	IN	A	192.112.36.4
H.ROOT-SERVERS.NET.	6D	IN	A	128.63.2.53
I.ROOT-SERVERS.NET.	6D	IN	A	192.36.148.17
J.ROOT-SERVERS.NET.	6D	IN	A	198.41.0.10
K.ROOT-SERVERS.NET.	6D	IN	A	193.0.14.129
L.ROOT-SERVERS.NET.	6D	IN	A	198.32.64.12
M.ROOT-SERVERS.NET.	6D	IN	A	202.12.27.33

Las zonas irán de la siguiente manera:

#### Contenido del archivo: `/etc/named.conf`

```
options {
    directory "/var/named";
    //query-source port 53;
};

//controls {
//    inet 127.0.0.1 allow { localhost; } keys { rndc-key; };
//};

//key "rndc_key" {
//    algorithm hmac-md5;
//    secret
"c3Ryb25nIGVub3VnaCBmb3lgYSBtYW4gYnV0IG1hZGUgZm9yIGEd29tYW4K";
//};

zone "." {
    type hint;
    file "root.hints";
};
```

```
zone "0.0.127.in-addr.arpa" {
    type master;
    file "127.0.0";
};
zone "xion.org" IN {
    type master;
    file "xion.org.zone";
    allow-update {none; };
};
zone "10.168.192.in-addr.arpa" IN {
    type master;
    file "10.168.192.in-addr.arpa.zone";
    allow-update {none; };
};
```

3. **Configurar Diferentes Servicios.**- Se configura los servicios para el caso de nombres a direcciones IP.

Abrimos el archivo named.local que se encuentra bajo el directorio `/var/named/` y lo editamos cambiando la palabra localhost por el nombre de nuestro dominio "xion.org".

En el archivo named.local, existen los siguientes registros:

- **SOA (Start of Authority - Inicio de Autoridad).**- Indica el inicio de los datos para una zona y define parámetros que afectan a todos los registros para la zona.
- **NS (Name Server - Servidor de Nombres).**- Indica el servidor de nombres.
- **A (Address - Dirección).**- Convierte un nombre de Estación en una dirección IP.
- **PTR (Pointer - Puntero).**- Convierte una dirección IP a un nombre de Estación.

A continuación se muestra el contenido del archivo de configuración xion.org.zone, con los diferentes servicios.

Contenido de los archivos: /var/named/xion.org.zone, y /var/named/chroot/var/named/xion.org.zone

```
$TTL 86400
@      IN      SOA    xion.org.    root@xion.org.    (
                                20070633; serial - Serie
                                28800 ; refresh – Tiempo de Refresco
                                14400 ; retry – Tiempo entre Consultas
                                3600000 ; expire – Tiempo de Expiración de la Zona
                                86400 ; ttl – Tiempo de Vida del Dominio
                                )

      NS     xion.org.
      TXT    "Xion Org"
localhost A    127.0.0.1
www       A    192.168.10.1
ftp       A    192.168.10.1
mail      A    192.168.10.1
dns       A    192.168.10.1
dhcp      A    192.168.10.1
;router   A    192.168.10.2
;Workstations
;ws-1     A    192.168.10.20
;ws-2     A    192.168.10.21
```

Luego de analizar los registros y colocando la información correspondiente, escribamos las líneas para la configuración de los diferentes servicios y guardamos creando el archivo xion.org.zone. Los servicios a configurarse son el DNS, WWW, FTP y DHCP, se procede de la siguiente forma:

A continuación se configura los servicios para el caso de nombres a direcciones IP. Bajo /var/named tomamos como base el archivo named.local y configuramos los

mismos servicios anteriores pero en forma inversa, para guardar creamos el archivo `xion.org.zone.inv` de la siguiente forma:

#### Contenido del archivo: `/var/named/xion.org.zone.inv`

```
$TTL 86400
@      IN      SOA    xion.org.      root.xion.org. (
                        20070633; serial - Serie
                        28800 ; refresh - Tiempo de Refresco
                        14400 ; retry - Tiempo entre Consultas
                        3600000 ; expire - Tiempo de Expiración de la zona
                        86400 ; ttl - Tiempo de Vida del Dominio
                        )

      NS     xion.org.

1      PTR   dns.xion.org.
1      PTR   www.xion.org.
1      PTR   dhcp.xion.org.
1      PTR   ftp.xion.org.
1      PTR   mail.xion.org.
;2     PTR   router.xion.org.
;Workstations
;20    PTR   ws-1.xion.org.
;21    PTR   ws-2.xion.org.
```

**Nota:** Es muy importante cuando modifiquemos los archivos `xion.org.zone` y `xion.org.zone.inv`, se debe incrementar en uno el campo Serial para que se actualicen las zonas en el servidor y no genere errores. Se debe evitar dejar espacios en blanco, use el tabulador para separar las cadenas de texto e incluya los puntos al final de los nombres de los servidores.

4. **Levantar el Servicio.**- Para levantar el servicio escribir el siguiente comando:  
service named start
5. **Verificar Errores.**- Para verificar si la configuración del DNS es satisfactoria verificamos los N últimos mensajes para ver si existió errores en dicha información, para lo cual en la consola escribimos la siguiente líneas de comandos:

#### Comando "tail" ejecutado en una Consola de GNU/Linux

```
[root@xion root]# tail -10 /var/log/messages
```

#Ejemplo de salida.

```
Jan 30 16:59:27 server named[2021]: listening on IPv4 interface eth0,  
192.168.10.1#53
```

#... Más mensajes.

```
Ene 30 16:59:27 server named: Iniciación de named succeded
```

```
[root@xion root]#
```

6. **Verificar si Corre el DNS.**- Hacemos ping al servicio WWW, usando los comandos:  
ping www.xion.org, host www.xion.org o host 192.168.10.1, si no existe problemas nos devolverá información de la traducción de los nombres.
7. **Funcionamiento del DNS.**- Para comprobar como realmente funciona la configuración del DNS creamos una página html, que será la página que se presentará en primera instancia, esta se guarda como index.html que se encuentra en /var/www/html en el caso de que no se cree esta página se presentará la pantalla de Bienvenida de Red Hat Linux.
8. **Servidor Web.**- En el directorio /etc/http/conf/ en el archivo httpd.conf buscamos la palabra ServerName y a continuación colocamos el nombre de nuestro dominio www.xion.org.

#### Comando a ejecutar en la Consola de GNU/Linux para iniciar el Servicio Apache

```
service httpd start
```



Por último nos conectamos en red, partiendo de una máquina con Microsoft Windows colocamos en su navegador (Internet Explorer o Mozilla Firefox) el nombre del dominio [www.xion.org](http://www.xion.org) o la dirección IP, y como respuesta obtenemos la página `index.html` que se aprecia en la Figura XGL 7.



Figura XGL 7 – Página Web alojada en el servidor con Red Hat Enterprise Linux 3.0.

## Servidor DHCP (Dinamic Host Configuration Protocol).

DHCP es un protocolo de red para asignar automáticamente información TCP/IP a equipos cliente. Es útil para proporcionar un modo rápido de configuración de la red del cliente. Al configurar el sistema cliente, el administrador puede seleccionar el protocolo DHCP y no especificar una dirección IP, una máscara de red, un Gateway o Servidor DNS. El cliente recupera esta información desde el servidor DHCP.

### Configuración de un Servidor DHCP.

Puede configurarse un servidor DHCP mediante el archivo `/etc/dhcpd.conf`. DHCP también usa el archivo `/var/lib/dhcp/dhcpd.leases` para almacenar la base de datos de arrendamiento de clientes.

### Archivo de Configuración.

Se debe crear el archivo de configuración `dhcpd.conf`, que almacena la información de red de los clientes. Se pueden declarar opciones globales para todos los clientes, o bien opciones para cada sistema cliente. El archivo de configuración puede contener tabulaciones o líneas en blanco adicionales para facilitar el formato. Las palabras clave no distinguen entre mayúsculas y minúsculas, se usa el símbolo `#` (numeral) para denotar una línea de comentarios.

Hay dos tipos de esquemas de actualización implementados actualmente, el modo de actualización del **DNS ad-hoc** y el modo de actualización de Interacción **DHCP-DNS**. Si y cuando estos dos son aceptados como parte del proceso estándar de IETF, habrá un tercer modo el método estándar de actualización DNS. El servidor DHCP tiene que estar configurado para usar uno de estos dos tipos de actualización. Si se desea conservar el mismo comportamiento, añada la siguiente línea al inicio del archivo de configuración:

#### Línea añadida al archivo `dhcpd.conf`

```
file: ddns-update-style ad-hoc;
```

Añada la siguiente línea al inicio del archivo de configuración:

### Línea añadida al archivo dhcpd.conf

```
ddns-update-style interim;
```

El archivo de configuración posee dos tipos de información:

- ✚ **Parámetros.**- Establece cómo se realiza una tarea, si debe llevarse a cabo una tarea o las opciones de configuración de red que se enviarán al cliente.
- ✚ **Declaraciones.**- Describen la topología de red, describen los clientes, proporcionan direcciones para los clientes o aplican un grupo de parámetros a un grupo de declaraciones.

**Importante.**- Si cambia el archivo de configuración, los cambios no se aplicarán hasta reiniciar el demonio DHCP; para que surta efecto use el comando:

### Comando a ejecutar en la Consola de GNU/Linux para reinicio de servicio de DHCP

```
service dhcpd restart.
```

A continuación se muestran varios ejemplos de configuración para DHCP.

### Ejemplo 1 – Configuración de una Subred

```
subnet 192.168.10.0 netmask 255.255.255.0 {  
    option routers      192.168.10.254;  
    option subnet-mask 255.255.255.0;  
    option domain-name "xion.org";  
    option domain-name-servers 192.168.10.1;  
    option time-offset -18000;  
    range dynamic-bootp 192.168.10.10 192.168.10.100;  
};
```

Todas las subredes que comparten la misma red física deben especificarse dentro de una declaración `shared-network` como se muestra en el siguiente ejemplo.

Los parámetros dentro de `shared-network` pero fuera del cerco de las declaraciones `subnet` se consideran parámetros globales.

El nombre de `shared-network` debe ser el título descriptivo de la red, como por ejemplo `test-lab`, para describir todas las subredes en un entorno de laboratorio de pruebas.

### Ejemplo 2 – Configuración de una Red Compartida

```
subnet-network name {  
    option domain-name "test.xion.org";  
    option domain-name-servers ns1.xion.org, ns2.xion.org;  
    option routers 192.168.10.254;  
    ;more parameters for EXAMPLE shared-network  
    subnet 192.168.10.0 netmask 255.255.255.0{  
        ;parameters for subnet  
        range 192.168.10.1 192.168.10.31;  
    };  
    subnet 192.168.10.32 netmask 255.255.255.0{  
        ;parameters for subnet  
        range 192.168.10.33 192.168.10.63;  
    };  
};
```

Como se muestra en el ejemplo 3, la declaración `group` puede utilizarse para aplicar parámetros globales a un grupo de declaraciones. Puede agrupar redes compartidas, subredes, `hosts` u otros grupos.

Para configurar un servidor DHCP que arrienda una dirección IP dinámica a un sistema dentro de una subred, modifique el ejemplo 4 con sus valores.

### Ejemplo 3 – Configuración de un Grupo (Group)

```
group {
option routers    192.168.10.254;
option subnet-mask 255.255.255.0;
option domain-name "xion.org";
option domain-name-servers 192.168.10.1;
option time-offset -18000; #Eastern Standard Time
host gerencia {
    option host-name "gerencia.xion.org";
    hardware ethernet 00:A0:78:8E:9E:AA;
    fixed-address 192.168.10.4;
};
hosts soporte {
    option host-name "soporte.xion.org";
    hardware ethernet 00:A1:DD:74:C3:F2;
    fixed-address 192.168.10.6;
};
};
```

En el ejemplo 4, se declara un tiempo de arrendamiento por defecto, un tiempo de arrendamiento máximo y los valores de configuración de red para los clientes.

En este ejemplo se asigna una dirección IP en el rango 192.168.10.10 y 192.168.10.100 a los sistemas clientes. Para asignar una dirección IP a un cliente según la dirección MAC de la tarjeta de interfaz de red, use el parámetro hardware Ethernet dentro de la declaración del host.

En el ejemplo 5, la declaración "host soporte" especifica que la interfaz de red con una dirección MAC 00:A0:78:8E:9E:AA siempre recibe la dirección IP 192.168.10.4. Tenga en cuenta que puede usar el parámetro opcional host-name para asignar un nombre host al cliente.

#### Ejemplo 4 – Configuración del Parámetro Rango (Range)

```
default-lease-time 600;
max-lease-time 7200;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.10.255;
option routers 192.168.10.254;
option domain-name-servers 192.168.10.1, 192.168.10.2;
option domain-name "xion.org";

subset 192.168.10.0 netmask 255.255.255.0 {
    range 192.168.10.10 192.168.10.100;
};
```

En el ejemplo 5, tenemos el nombre de un servidor soporte.xion.org con IP estática en base a la dirección MAC de su tarjeta de red.

#### Ejemplo 5 – Configuración de Dirección IP Estática con DHCP

```
host soporte {
    option host-name "soporte.xion.org";
    hardware Ethernet 00:A0:78:8E:9E:AA;
    fixed-address 192.168.10.4;
};
```

### Base de Datos de Arrendamiento.

En el servidor DHCP, el archivo `/var/lib/d.C./dhcpd.leases` almacena la base de datos de arrendamiento del cliente DHCP. La información de arrendamiento de DHCP de cada dirección IP asignada recientemente se almacena de modo automático, esta incluye la longitud de arrendamiento, a quién se ha asignado la dirección IP, las fechas iniciales y finales de renta, y la dirección MAC de la tarjeta de interfaz de red utilizada

para recuperar el arrendamiento. Todas las horas de la base de datos de arrendamiento se expresan según formato GMT, no con la hora local.

## Arranque y Parada del Servidor.

**Importante.-** Antes de arrancar por primera vez el servidor DHCP, asegúrese de que existe un archivo `dhcpd.leases` para que no falle el arranque. Use el comando `touch /var/lib/dhcp/dhcpd.leases` para crear el archivo en caso de no exista.

Para arrancar el servicio DHCP:

COMANDO	DESCRIPCIÓN
<code>/sbin/service dhcpd star</code>	Inicia el demonio dhcpd.
<code>/sbin/service dhcpd stop</code>	Detiene el demonio dhcpd.
<code>/sbin/service dhcpd restart</code>	Reinicia el demonio dhcpd.

Si tiene más de una interfaz de red conectada al sistema, pero sólo desea que el servidor DHCP arranque en una de las interfaces, puede configurar el servidor DHCP para que sólo arranque en ese dispositivo. En `/etc/sysconfig/dhcpd`, agregue el nombre de la interfaz a la lista `DHCPDARGS`:

### Modificación del contenido del archivo `/etc/sysconfig/dhcpd`

```
#Command line options here
DHCPDARGS=eth0 #interfaz de red ethernet 0
```

Esto es útil si tiene una máquina firewall con dos tarjetas de red. Se puede configurar una tarjeta de red como cliente DHCP para recuperar una dirección IP en Internet y la otra tarjeta de red puede utilizarse como servidor DHCP para la red interna detrás del firewall. Su sistema será más seguro si especifica la tarjeta conectada a la red interna ya que los usuarios no pueden conectarse al demonio vía Internet.



Otras opciones de línea de comando que pueden ser especificadas en el archivo `/etc/sysconfig/dhcpd` incluyen:

- ✚ **-p <portnum>.-** Especifique el número de puerto UDP en el cual debería escuchar `dhcpd`. Está predeterminado el puerto 67. El servidor DHCP transmite las respuestas al cliente a un puerto con un número superior que el puerto UDP especificado. Por ejemplo si acepta el puerto predeterminado 67, el servidor escucha en el puerto 67 y responde en el puerto 68. Si especifica un puerto en este momento y usa el agente de transmisión, debería especificar el mismo puerto en el que el agente debería escuchar.
- ✚ **-f.-** Ejecutar el demonio como un proceso de primer plano. Casi siempre se usa para la depuración.
- ✚ **-d.-** Registrar el dominio del servidor DHCP en el descriptor de errores estándar. Casi siempre se usa para el depurado. Si no está especificado, el registro será escrito en `/var/log/messages`.
- ✚ **-cf filename.-** Especifica la localización del archivo de configuración. La configuración por defecto es `/etc/dhcpd.conf`.
- ✚ **-lf filename.-** Especifica la ubicación de la base de datos de arrendamiento. Si ya existe el archivo de la base de datos de arrendamiento, es muy importante que el mismo archivo sea usado cada vez que el servidor DHCP se inicia. Se le recomienda que use esta opción sólo para propósitos de depuración en máquinas que no estén en producción. La ubicación por defecto es `/var/lib/dhcpd.leases`.
- ✚ **-q.-** No imprima el mensaje de copyright entero cuando inicie el demonio.

## Agente de Transmisión DHCP.

El agente de transmisión DHCP (`dhrelay`) le permite transmitir las peticiones DHCP y BOOTP desde la subred sin un servidor DHCP para uno o más servidores en otras subredes. Cuando un cliente DHCP pide información, el agente de transmisión DHCP reenvía la petición a la lista de servidores DHCP específica cuando se inicia el agente de transmisión DHCP. Cuando un servidor DHCP devuelve una respuesta, la respuesta puede ser broadcast o unicast en la red que ha enviado la petición original.

El agente de transmisión escucha las peticiones DHCP en todas las interfaces a menos que las interfaces estén especificadas en `/etc/sysconfig/dhcrelay` con la directiva `INTREFACES`. Para iniciar el agente de transmisión DHCP, use el comando `service dhcrelay start`.

## Configuración del Cliente DHCP.

Antes de configurar un cliente DHCP es de asegurarse que el kernel reconoce la tarjeta de la interfaz de red. La mayoría de las tarjetas se reconocen durante el proceso de instalación y el sistema se configura para utilizar el módulo de kernel correcto para la tarjeta. Si instala una tarjeta después de la instalación, la aplicación **Kudzu** debería reconocerla y solicitarle que configure el módulo del kernel correspondiente para ésta tarjeta.

Para configurar un cliente DHCP manualmente, debe modificar el archivo `/etc/sysconfig/network` para habilitar el uso del archivo de configuración y de red en los dispositivos del directorio `/etc/sysconfig/network-scripts`. En este directorio, cada dispositivo debería tener un archivo de configuración llamado `ifcfg-eth0` donde `eth0` es el nombre del dispositivo de red.

El archivo `/etc/sysconfig/network` debería contener la línea: `NETWORKING=yes`, puede disponer de más información en este archivo.

El archivo de configuración `/etc/sysconfig/network-scripts/ifcfg-eth0` debería tener las siguientes líneas:

### Contenido del archivo `/etc/sysconfig/network-scripts/ifcfg-eth0`

```
DEVICE=eth0
BOOTPROTO=dhcp
ONBOOT=yes
```

**Nota:** Necesita un archivo de configuración para cada dispositivo que desee configurar para el uso de DHCP como ejemplo: `eth0` (Ethernet 0), `eth1` (Ethernet 1), `eth2` (Ethernet 2), etc.

## Compartición de Archivos con el Servidor vsftpd.

### Protocolo FTP (File Transfer Protocol).

El Protocolo de Transferencia de Archivos es idóneo para la transferencia de grandes bloques de datos a través de redes que soporten TCP/IP. El servicio utiliza los puertos 20 y 21, exclusivamente sobre TCP.

El puerto 20 es utilizado para el flujo de datos entre cliente y servidor. El puerto 21 es utilizado para el envío de órdenes del cliente hacia el servidor. Prácticamente todos los Sistemas Operativos y plataformas incluyen soporte para FTP, lo que permite que cualquier computadora conectada a una red basada sobre TCP/IP pueda hacer uso de este servicio a través de un cliente FTP.

### Servidor vsftpd (Very Secure FTP Daemon).

Es un equipamiento lógico utilizado para implementar servidores de archivos a través del protocolo **FTP**. Se distingue principalmente porque sus valores predeterminados son muy seguros y por su sencillez en la configuración, comparado con otras alternativas como ProFTPD y Wu-ftp. Puede encontrar más información en el sitio web: <http://vsftpd.beasts.org/>

### Instalación de vsftpd Utilizando Yum y Up2Date.

COMANDO	DESCRIPCIÓN
<code>yum -y install vsftpd</code>	Instalación o actualización del paquete vsftpd, en CentOS Linux 5 y White Box Enterprise Linux 4.
<code>up2date -i vsftpd</code>	Instalación o actualización en Red Hat Enterprise Linux 4 y Fedora Core Linux 7.

### Ficheros de Configuración.

Se utilizan los siguientes archivos de configuración: /etc/vsftpd.user\_list define la lista usuarios que pueden acceder o no al servidor, dependiendo de la configuración. /etc/vsftpd/vsftpd.conf es el archivo de configuración principal.

Utilice un editor de texto y modifique el contenido del archivo /etc/vsftpd/vsftpd.conf. A continuación se describe el significado de cada parámetro:

- ✚ **Parámetro anonymous\_enable.**- Se utiliza para definir si se permitirán los accesos anónimos al servidor. Establezca como valor **YES** o **NO** de acuerdo a lo que se requiera.

#### Parámetro anonymous\_enable – Parte del archivo: /etc/vsftpd/vsftpd.conf

```
anonymous_enable=YES
```

- ✚ **Parámetro local\_enable.**- Es particularmente interesante si se combina con la función de jaula (**chroot**). Establece si se van a permitir los accesos autenticados de los usuarios locales del sistema. Establezca como valor **YES** o **NO** de acuerdo a lo que se requiera.

#### Parámetro local\_enable – Parte del archivo: /etc/vsftpd/vsftpd.conf

```
local_enable=YES
```

- ✚ **Parámetro write\_enable.**- Establece si se permite el mandato **write** (escritura) en el servidor. Establezca como valor **YES** o **NO** de acuerdo a lo que se requiera.

#### Parámetro write\_enable – Parte del archivo: /etc/vsftpd/vsftpd.conf

```
write_enable=YES
```

- ✚ **Parámetro ftpd\_banner.**- Este parámetro sirve para establecer el banderín de bienvenida que será mostrado cada vez que un usuario acceda al servidor. Puede establecerse cualquier frase breve que considere conveniente.

#### Parámetro ftpd\_banner – Parte del archivo: /etc/vsftpd/vsftpd.conf

```
ftpd_banner=Bienvenido al servidor FTP de la empresa Xion Ecuador
```

Estableciendo restricciones de acceso para los usuarios: parámetros `chroot_local_user` y `chroot_list_file`.

De modo predeterminado los usuarios del sistema que se autentiquen tendrán acceso a otros directorios del sistema fuera de su directorio personal. Para restringir el acceso de los usuarios a los directorios que no les pertenecen use el parámetro `chroot_local_user` que habilitará la función de `chroot()` y los parámetros `chroot_list_enable` y `chroot_list_file` para establecer el archivo con la lista de usuarios que quedarán excluidos de la función `chroot()`.

#### Restricciones de Acceso de Usuarios – Parte del archivo: `/etc/vsftpd/vsftpd.conf`

```
chroot_local_user=YES
chroot_list_enable=YES
chroot_list_file=/etc/vsftpd/vsftpd.chroot_list
```

**Nota:** No olvide crear el fichero `/etc/vsftpd/vsftpd.chroot_list`, ya que de otro modo no arrancará el servicio `vsftpd`.

#### Comando a ejecutar en la Consola de GNU/Linux para crear el Archivo de Listas

```
touch /etc/vsftpd/vsftpd.chroot_list
```

#### Control del Ancho de Banda.

✚ **Parámetro `anon_max_rate`.**- Se utiliza para limitar la tasa de transferencia en bytes por segundo (Bps) para los usuarios anónimos, algo sumamente útil en servidores FTP de acceso público. En el siguiente ejemplo se limita la tasa de transferencia a 5 Kbps para los usuarios anónimos:

#### Parámetro `anon_max_rate` – Parte del archivo: `/etc/vsftpd/vsftpd.conf`

```
anon_max_rate=5120
```

- ✚ **Parámetro local\_max\_rate.**- Hace lo mismo que **anon\_max\_rate**, pero aplica para usuarios locales del servidor FTP. En el siguiente ejemplo se limita la tasa de transferencia a 5 Kbps para los usuarios locales:

Parámetro local\_max\_rate – Parte del archivo: /etc/vsftpd/vsftpd.conf

```
local_max_rate=5120
```

- ✚ **Parámetro max\_clients.**- Establece el número máximo de clientes que podrán acceder simultáneamente hacia el servidor FTP. En el siguiente ejemplo se limitará el acceso a 5 clientes simultáneos.

Parámetro max\_clients – Parte del archivo: /etc/vsftpd/vsftpd.conf

```
max_clients=5
```

- ✚ **Parámetro max\_per\_ip.**- Establece el número máximo de conexiones que se pueden realizar desde una misma dirección IP. Tome en cuenta que algunas redes acceden a través de un servidor Proxy o puerta de enlace y debido a esto podrían quedar bloqueados innecesariamente algunos accesos. El siguiente ejemplo se limita el número de conexiones por IP simultáneas a 5.

Parámetro max\_per\_ip – Parte del archivo: /etc/vsftpd/vsftpd.conf

```
max_per_ip=5
```

## Inicio, Detención y Reinicio del Servicio vsftpd.

El servicio vsftpd incluido en distribuciones como CentOS Linux 4, Red Hat™ Enterprise Linux 4 y White Box Enterprise Linux 4 puede iniciar, detenerse o reiniciar a través de un guión similar a los del resto del sistema y no depende del servicio xinetd.

COMANDO

DESCRIPCIÓN



service vsftpd start	Inicia por primera vez el servicio.
service vsftpd restart	Reinicia el servicio, en casos de modificación de la configuración del Servidor vsftpd.
service vsftpd stop	Detiene el servicio.

Para hacer que el servicio de vsftpd esté activo con el siguiente inicio del sistema, en todos los niveles de arranque (2, 3, 4, y 5), se utiliza lo siguiente:

#### Comando a ejecutar en la Consola de GNU/Linux para arrancar el servicio vsftpd

```
chkconfig vsftpd on
```

Si se utilizan cortafuegos con políticas estrictas, como por ejemplo Shorewall, es necesario abrir los puertos 20 (FTP-DATA) y 21 (FTP) por TCP.

#### Contenido del archivo de Configuración de Shorewall: /etc/shorewall/rules

```
#ACTION SOURCE DEST PROTO DEST SOURCE # PORT PORT(S)1 ACCEPT net fw tcp  
20,21  
#LAST LINE -- ADD YOUR ENTRIES BEFORE THIS ONE -- DO NOT REMOVE
```

## Compartición de Archivos usando el Servidor NFS (Network File System).

El sistema NFS “Network File System” permite montar una partición perteneciente a una máquina remota como si fuera una partición local. Consiste en un método rápido y eficaz de compartición de archivos y espacio de disco entre distintos computadores de una red que soporten este sistema. Necesitamos tener instalado en nuestro equipo “portmap” y el paquete “nfs-utils”. El software “portmap” permite realizar conexiones RPC al servidor y es el encargado de permitir o no el acceso de otros equipos.

COMANDO	DESCRIPCIÓN
<code>ps aux   grep portmap</code>	Determino si nfs esta activo.
<code>rpcinfo -p</code>	Determino si “portmap” soporta nfs, caso contrario se debe instalar los paquetes “nfs-utils”.

## Compartición del Disco Duro.

Se determina que directorios deseo compartir de mi disco duro para la red. Como ejemplo tendremos:

### Compartición del Directorio RedXion

```
/home/redxion
```

Se debe editar el archivo `/etc/exports`, añadiendo la siguiente línea:

### Modificación del archivo `/etc/exports`

```
/home/redxion 192.168.10.6(rw, sync)
```



## Seguridad en NFS.

Se debe editar los archivos `/etc/hosts.allow` y `/etc/host/hosts.deny` para indicar que computadores de mi red pueden acceder al servidor de disco NFS. Vea los siguientes ejemplos:

### Ejemplo 1 - Contenido del archivo `/etc/hosts.allow`

```
#
# hosts.allow This file describes the names of the hosts which are allowed to use the
# local INET services, as decided by the '/usr/sbin/tcpd' server.
#
portmap:192.168.10.6
lockd:192.168.10.6
mount:192.168.10.6
rquotad:192.168.10.6
statd:192.168.10.6
```

### Ejemplo 2 - Contenido del archivo `/etc/hosts.deny`

```
#
# hosts.deny This file describes the names of the hosts # which are *not* allowed to
# use the local INET services, # as decided by the '/usr/sbin/tcpd' server.
# The portmap line is redundant, but it is left to remind # you that the new secure
# portmap uses hosts.deny and # hosts.allow. In particular you should know that NFS
# uses # portmap!
#
portmap:ALL
lockd:ALL
mountd:ALL
rquotad:ALL
statd:ALL
```

## Iniciar la Compartición de Archivos.

Si hemos modificado el archivo `/etc/exports` después de haber iniciado el demonio `nfsd`, debo reiniciarlo usando el comando `exportfs -ra` que permite actualizar los cambios del servicio NFS o a su vez usar el comando `/sbin/service nfs restart`.

## Configuración del Cliente NFS.

En el equipo cliente como “root” ejecute el comando `showmount -e 192.168.10.1`, que permite consultar los volúmenes exportados a través del Servidor NFS.

Montamos en la máquina cliente (192.168.10.6) el directorio compartido en el servidor (192.168.10.1).

### Comando a ejecutar en la Consola de GNU/Linux para levantar un Cliente NFS

```
mount 192.168.10.1:/home/redxion /mnt/nfs
```

Y podremos acceder a todos los archivos y directorios compartidos en el Servidor GNU/Linux.

## Instalación y Configuración de Emulador Wine

El emulador Wine es un programa Open Source, permite la ejecución de programas de Microsoft Windows (incluyendo ejecutables de MS-DOS, Windows 3.x y Win32) sobre Sistemas Operativos Unix y GNU/Linux. Consiste en un programa cargador, que carga y ejecuta un programa binario de Microsoft Windows, y una librería (llamada Winelib) que implementa las llamadas a la API<sup>5</sup> de Windows usando sus equivalentes Unix o X11. La librería puede también utilizarse para portar código Win32 a ejecutables Unix nativos.

Wine corre en sistemas Unix x86 incluidos GNU/Linux versión 2.0.36 o superior, FreeBSD 5.3 o superior, Mac OS X y Solaris x86 2.5 o superior.

Wine es software libre, publicado bajo los términos de la licencia GNU LGPL (Lesser General Public License).

La versión Wine 0.9.32 fue lanzada el 2 de Marzo del 2007, y cumple con las expectativas que los usuarios necesitan, estas son:

- Correcciones en Direct3D y mejoras en el desempeño.
- Correcciones en la compilación de cmd.exe.
- Corrección y mejoras de ayuda en formato HTML.
- Corrección general de errores.

El código fuente está disponible para descarga en los siguientes enlaces:

- <http://ibiblio.org/pub/linux/system/emulators/wine/wine-0.9.32.tar.bz2>
- <http://www.winehq.org/site/download>

### Información del Núcleo de GNU/Linux.

Se hace referencia solo a distribuciones GNU/Linux como son: Debian Linux, Red Hat Enterprise Linux y sus clones CentOS Linux, Fedora Linux, etc.

<sup>5</sup> **API.**- Application Program Interface. Es una interfaz para programas de aplicación usadas en sistemas operativos Windows, que encapsulan todas las rutinas necesarias para el control del Hardware del ordenador.

Es recomendable tener un núcleo del sistema operativo GNU/Linux mínimo de 2.4.x, en la actualidad hay el núcleo 2.6.x soportado por la mayor parte de distribuciones GNU/Linux.

## Sistemas de Ficheros Soportados

Wine debería ejecutarse en la mayoría de los sistemas de ficheros. Sin embargo, Wine no conseguirá iniciarse si un MS-DOS es utilizado para el directorio /tmp. Unos cuantos problemas de compatibilidad se han reportado también al utilizar ficheros accedidos a través del servicio de Samba. Además, como de momento NTFS sólo puede ser utilizado con seguridad con acceso de sólo lectura, se recomienda no utilizar NTFS, ya que los programas Microsoft Windows necesitan acceso de escritura en casi cualquier sitio. En el caso de ficheros NTFS, cópielos a una localización escribible.

## Requisitos Básicos

Necesita tener los ficheros de inclusión (archivos cabecera) de desarrollo de X11 instalados (llamados xlib6g-dev en Debian y XFree86-devel en Red Hat Enterprise).

## Requisitos de Herramienta de Construcción

- ✚ Sobre sistemas basados en x86 se requiere que el compilador sea gcc >= 2.7.2.
- ✚ Se necesita el programa "make" (preferiblemente GNU make).
- ✚ Se necesita tener instalado los paquetes flex versión 2.5 o superior y bison.

## Compilación de Ficheros de Wine

Ejecute los siguientes comandos para construir el ejecutable de Wine:

### Comandos a ejecutar en la Consola de GNU/Linux para instalar Wine

```
./configure  
make depend  
make
```

Para actualizar a nueva versión usando un fichero de parches, primero ejecute el comando "cd" para ir al directorio superior de la versión (el que contiene este fichero README).

### Ejecución de Comandos para insertar un parche en una instalación de Wine

```
make clean  
gunzip -c fichero-parche | patch -p1
```

Donde "fichero-parche" es el nombre del fichero de parches (algo como Wine-aammdd.diff.gz).

## Configuración de Wine

Una vez que Wine ha sido construido correctamente, puede ejecutar el comando "make install"; esto instalará el ejecutable de wine, la página de manual de Wine, y otros cuantos ficheros necesarios.

COMANDO	DESCRIPCIÓN
<code>dpkg -r wine</code>	Estos comandos permiten desinstalar una versión anterior conflictiva de Wine.
<code>rpm -e wine</code>	
<code>make uninstall</code>	

En el caso de que tenga problemas de carga de librerías (por ejemplo "Error while loading shared libraries: libntdll.so"), asegúrese de añadir la ruta de las librerías a /etc/ld.so.conf y ejecutar el comando ldconfig como Administrador "root".

## Ejecución de Programas

Cuando invoque al programa Wine, puede especificar la ruta completa al ejecutable, o sólo el nombre del fichero.

Por ejemplo: Para ejecutar el programa servidor de Cyber Control<sup>6</sup>:

COMANDO	DESCRIPCIÓN
wine servidor.exe	Usa la ruta de búsqueda indicada en el fichero de configuración para encontrar el fichero.
wine c:\\servidor.exe	Usa la sintaxis de nombre de fichero de MS-DOS.
wine /usr/servidor.exe	Usa la sintaxis de nombre de fichero de Unix.
wine servidor.exe /parametro1 - parametro2 parametro3	Llama al programa con parámetros, si el programa lo permite.

## Instalación de Wine desde Paquetes RPM

En distribuciones derivadas de Red Hat Enterprise Linux existen los paquetes RPM (Red Hat Packed Managed) que permiten la instalación de nuevos programas sin necesidad de compilar ningún complemento, lo importante es que tengamos instalado todas las dependencias necesarias en nuestro ordenador.

Como alternativa a los pasos anteriores de instalación de wine, que pueden resultar largos y complicados se pone a consideración el siguiente comando:

### Ejecución del Comando RPM para la instalación rápida de Wine

```
rpm -Uvh --nodeps wine*.*
```

Con el comando anterior desde Internet descargamos cualquier versión de wine desde cualquier Repositorio de Paquetes, sin importar las dependencias.

<sup>6</sup> **Cyber Control**: Programa Argentino Cliente/Servidor gratuito para el Control de Cybers

## Configuración del Router Quagga en CentOS Linux 5.0

El Router Quagga es un software derivado del Router Zebra, que es un clon del IOS del Router Cisco, se encuentra disponible bajo la Licencia GNU.

### Configuración Básica de Router Quagga.

Antes de proceder a configurar cualquier servicio, debemos consultar al sistema si esta instalado el paquete de Quagga ejecutando como root el siguiente comando:

#### Comando a ejecutar en la Consola de GNU/Linux

```
[root@xion /etc]# rpm -q quagga
```

Si no está instalado el paquete de quagga, ejecute el siguiente comando:

#### Comando a ejecutar en la Consola de GNU/Linux

```
[root@xion /etc]# yum install quagga
```

Después de haber instalado Quagga en el sistema procedemos a levantar los demonios correspondientes:

#### Comandos a ejecutar en la Consola de GNU/Linux

```
[root@xion /etc]# service zebra start  
[root@xion /etc]# service ripd start  
[root@xion /etc]# service ripngd start  
[root@xion /etc]# service ospfd start  
[root@xion /etc]# service ospf6d start  
[root@xion /etc]# service gbpdd start
```

## Ejemplo de Configuración del Router xion - Red 192.168.1.0

A continuación se muestra un ejemplo básico de configuración del Router Quagga para lo cual debe realizar lo siguiente:

### Comandos a ejecutar en la Consola de GNU/Linux – Consola de Router Quagga

```
[root@xion /etc]# vtysh
xion# configure terminal
xion(config)# interface eth0
xion(config-if)# ip address 192.168.1.2/24
xion(config-if)# no shutdown
xion(config-if)# exit
xion(config)# end
xion# show interface eth0
xion# exit
[root@xion /etc]#
```

## Configuración de Router Quagga en GNU/Linux CentOS 5.0.

La siguiente información fue probada en mi computador con GNU/Linux CentOS 5.0 y en el Laboratorio Intranets de la Escuela de Ingeniería Electrónica y Tecnología en Computación de la Escuela Superior Politécnica de Chimborazo con GNU/Linux CentOS 4.4 y CentOS 5.0. Para configurar de forma adecuada el Router Quagga realice los siguientes procedimientos:

Determinación de soporte del kernel de IPv6<sup>7</sup>:

<sup>7</sup> **IPv6.**- Internet Protocol versión 6, es el sucesor de Internet Protocol versión 4 (IPv4). Incluye un mayor rango de direcciones IP es utilizado por empresas transnacionales. El Hardware y Software debe ser capaz de soportar este protocolo.

### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# test -f /proc/net/if_inet6 && echo "existe soporte IPv6 en el kernel"
existe soporte IPv6 en el kernel
[root@xion ~]#
```

Si existe error en el paso anterior, cargue el modulo IPv6 correspondiente:

### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# modprobe ipv6
```

Si la carga fue exitosa tendremos:

### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# lsmod | grep -w 'ipv6' && echo "modulo IPv6 cargado exitosamente"
ipv6 251136 19 ip6t_REJECT
modulo IPv6 cargado exitosamente
[root@xion ~]#
```

Configuración de interfaces (Desactivar):

### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# /sbin/ifconfig eth0 down
```

Configuración de interfaces (Activar):

### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
root@xion ~]# /sbin/ifconfig eth0 up
```

Configuración de direcciones IPv6 (Añadir):

#### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ifconfig eth0 inet6 add 2001:db8:fedc:bcde::4/64
```

Configuración de direcciones IPv6 (Eliminar):

#### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ifconfig eth0 inet6 del 2001:db8:fedc:bcde::4/64
```

Mostrar los enlaces existentes - Se muestra varias interfaces debido a que se encontraba instalado la Máquina Virtual Xen en el Servidor CentOS Linux 5.0:

#### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ip link list
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: peth0: <NO-CARRIER,BROADCAST,NOARP,UP> mtu 1500 qdisc pfifo_fast qlen 1000
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
3: sit0: <NOARP> mtu 1480 qdisc noop
link/sit 0.0.0.0 brd 0.0.0.0
4: vif0.0: <BROADCAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc noqueue
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
5: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
link/ether 00:d0:09:9c:52:e7 brd ff:ff:ff:ff:ff:ff
6: vif0.1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
7: veth1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
8: vif0.2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
9: veth2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
10: vif0.3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
11: veth3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
12: xenbr0: <BROADCAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc noqueue
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
[root@xion ~]#
```

Mostrar nuestras direcciones IP: Se muestran varias direcciones virtuales debido a que en el servidor GNU/Linux CentOS 5.0 se tenían instaladas las máquinas Virtual VmWare y Xen.

### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ip address show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
inet6 ::1/128 scope host
valid_lft forever preferred_lft forever
2: peth0: <NO-CARRIER,BROADCAST,NOARP,UP> mtu 1500 qdisc pfifo_fast qlen 1000
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
3: sit0: <NOARP> mtu 1480 qdisc noop
link/sit 0.0.0.0 brd 0.0.0.0
4: vif0.0: <BROADCAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc noqueue
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
inet6 fe80::fcff:ffff:feff:ffff/64 scope link
valid_lft forever preferred_lft forever
5: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
link/ether 00:d0:09:9c:52:e7 brd ff:ff:ff:ff:ff:ff
inet 192.168.1.1/24 brd 192.168.1.255 scope global eth0
inet6 2001:db8:fedc:edef::4/64 scope global
valid_lft forever preferred_lft forever
inet6 fe80::2d0:9ff:fe9c:52e7/64 scope link
valid_lft forever preferred_lft forever
6: vif0.1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
7: veth1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
8: vif0.2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
9: veth2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
10: vif0.3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
11: veth3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
12: xenbr0: <BROADCAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc noqueue
link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
[root@xion ~]#
```

Mostrar nuestras rutas: Se muestra la dirección IPv4 192.168.1.0/24, asociada a la tarjeta de red eth0.



### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ip route show
192.168.1.0/24 dev eth0 proto kernel scope link src 192.168.1.1
[root@xion ~]#
```

Mostrar las tablas de ruta: Se muestra el Default Gateway del servidor.

### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
[root@xion ~]#
```

Reglas por defecto (Base de datos de normas de enrutado):

### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ip rule list
0: from all lookup 255
32766: from all lookup main
32767: from all lookup default
[root@xion ~]#
```

Tabla local (Normas de encaminamiento): Observamos las direcciones IPv4 192.168.1.0/24 y 127.0.0.0/8.

### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ip route list table local
local 192.168.1.1 dev eth0 proto kernel scope host src 192.168.1.1
broadcast 192.168.1.0 dev eth0 proto kernel scope link src 192.168.1.1
broadcast 127.255.255.255 dev lo proto kernel scope link src 127.0.0.1
broadcast 192.168.1.255 dev eth0 proto kernel scope link src 192.168.1.1
broadcast 127.0.0.0 dev lo proto kernel scope link src 127.0.0.1
local 127.0.0.1 dev lo proto kernel scope host src 127.0.0.1
local 127.0.0.0/8 dev lo proto kernel scope host src 127.0.0.1
[root@xion ~]#
```

Tabla main (Normas de encaminamiento):

#### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ip route list table main
192.168.1.0/24 dev eth0 proto kernel scope link src 192.168.1.1
[root@xion ~]#
```

Generación de una nueva regla: Para añadir una regla en la configuración del servidor, ejecute los siguientes comandos.

#### Comandos a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# echo 200 Xion >> /etc/iproute2/rt_tables
[root@xion ~]# ip rule add from 192.168.1.10 table Xion
[root@xion ~]# ip rule ls
0: from all lookup 255
32765: from 192.168.1.10 lookup Xion
32766: from all lookup main
32767: from all lookup default
[root@xion ~]#
```

Generación de la tabla Xion y refresco de la cache de rutas:

#### Comandos a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ip route add default via 192.168.1.1 dev eth0 table Xion
[root@xion ~]# ip route flush cache
[root@xion ~]#
```

A continuación se muestra los pasos de configuración necesarios al utilizar la consola Vtysh del Router Quagga, los comandos a ejecutar son similares a los usados por el IOS<sup>8</sup> del Router Cisco.

<sup>8</sup> **IOS.**- Sistema Operativo interprete utilizado por los Routers Cisco.

### Comandos a ejecutar en la Consola de Vtysh del Router Quagga

```
xion# configure terminal
xion(config)# interface eth0
xion(config-if)# ip address 10.0.1.0/24
xion(config-if)# no shutdown
xion(config-if)# exit
xion(config)# interface eth1
xion(config-if)# ip address 10.0.2.0/24
xion(config-if)# no shutdown
xion(config-if)# exit
xion(config)# exit
xion# copy running-config startup-config
xion#
```

Creación de Túneles IPv6, considero una Red 3ffe:406:5:1:5:a:2:1/96, con IPv4 Local 172.16.17.18 y la IPv4 del Router es 172.22.23.24:

### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ip tunnel add sixbone mode sit remote 172.22.23.24 local
172.16.17.18 ttl 255
[root@xion ~]# ip link set sixbone up
[root@xion ~]# ip addr add 3ffe:406:5:1:5:a:2:1/96 dev sixbone
[root@xion ~]# ip route add 3ffe::/15 dev sixbone
[root@xion ~]#
```

Utilizamos ifconfig para ver todas las interfaces:

### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ifconfig
eth0 Link encap:Ethernet HWaddr 00:D0:09:9C:52:E7
inet addr:192.168.1.1 Bcast:192.168.1.255 Mask:255.255.255.0
inet6 addr: 2001:db8:fedc:edef::4/64 Scope:Global
inet6 addr: fe80::2d0:9ff:fe9c:52e7/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
```



```
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:92 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:0 (0.0 b) TX bytes:18461 (18.0 KiB)

lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:866 errors:0 dropped:0 overruns:0 frame:0
TX packets:866 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:1278006 (1.2 MiB) TX bytes:1278006 (1.2 MiB)

peth0 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
UP BROADCAST NOARP MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
Interrupt:11 Base address:0xde00

sixbone Link encap:IPv6-in-IPv4
inet6 addr: fe80::ac10:1112/128 Scope:Link
inet6 addr: 3ffe:406:5:1:5:a:2:1/96 Scope:Global
UP POINTOPOINT RUNNING NOARP MTU:1480 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

vif0.0 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
inet6 addr: fe80::fcff:ffff:feff:ffff/64 Scope:Link
UP BROADCAST RUNNING NOARP MTU:1500 Metric:1
RX packets:135 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:28260 (27.5 KiB) TX bytes:0 (0.0 b)

xenbr0 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
UP BROADCAST RUNNING NOARP MTU:1500 Metric:1
RX packets:135 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:26370 (25.7 KiB) TX bytes:0 (0.0 b)

[root@xion ~]#
```

Ejecutamos el comando ping6, para comprobar si el túnel esta activo:

#### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# ping6 3ffe:406:5:1:5:a:2:1
PING 3ffe:406:5:1:5:a:2:1(3ffe:406:5:1:5:a:2:1) 56 data bytes
64 bytes from 3ffe:406:5:1:5:a:2:1: icmp_seq=0 ttl=64 time=0.386 ms
64 bytes from 3ffe:406:5:1:5:a:2:1: icmp_seq=1 ttl=64 time=0.271 ms
64 bytes from 3ffe:406:5:1:5:a:2:1: icmp_seq=2 ttl=64 time=0.243 ms
64 bytes from 3ffe:406:5:1:5:a:2:1: icmp_seq=3 ttl=64 time=0.246 ms

[3]+ Stopped ping6 3ffe:406:5:1:5:a:2:1
[root@xion ~]#
```

Configuración de IPv6, si esta activo el demonio radvd en el Sistema Operativo GNU/Linux:

#### Comando a ejecutar en la Consola de GNU/Linux para Configurar Router Quagga

```
[root@xion ~]# /sbin/ip -f inet6 addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436
inet6 ::1/128 scope host
valid_lft forever preferred_lft forever
4: vif0.0: <BROADCAST,NOARP,UP,LOWER_UP> mtu 1500
inet6 fe80::fcff:ffff:feff:ffff/64 scope link
valid_lft forever preferred_lft forever
5: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
inet6 2001:db8:fedc:cdef::4/64 scope global
valid_lft forever preferred_lft forever
inet6 fe80::2d0:9ff:fe9c:52e7/64 scope link
valid_lft forever preferred_lft forever
13: sixbone@NONE: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1480
inet6 3ffe:406:5:1:5:a:2:1/96 scope global
valid_lft forever preferred_lft forever
inet6 fe80::ac10:1112/128 scope link
valid_lft forever preferred_lft forever
[root@xion ~]#
```

## Compatibilidad de la Tarjeta de Red Eth0 con IPv6.

Utilizando la herramienta grafica “Control de Dispositivos de Red”, habilite la opción de “Compatibilidad con IPv6”, lo cual permite que la tarjeta eth0 pueda interpretar las tramas en formato IPv6.

El siguiente comando lista todos los servicios complementarios para soporte de IPv6.

### Comando a ejecutar en la Consola de GNU/Linux para Compatibilidad de IPv6

```
[root@xion /etc]# sysctl -a | egrep *net.ipv6
```

El siguiente comando visualiza la configuración inicial del archivo de configuración de la red.

### Comando a ejecutar en la Consola de GNU/Linux para Compatibilidad de IPv6

```
[root@xion /etc]# more /etc/sysconfig/network  
NETWORKING=yes  
NETWORKING_IPV6=yes  
HOSTNAME=xion  
[root@xion /etc]#
```

Modificamos la configuración de la red, de la siguiente manera:

### Comando a ejecutar en la Consola de GNU/Linux para Compatibilidad de IPv6

```
[root@xion /etc]# gedit /etc/sysconfig/network  
[root@xion /etc]# more /etc/sysconfig/network  
NETWORKING=yes  
NETWORKING_IPV6=yes
```

```
IPV6FORWARDING=yes
```

```
IPV6_ROUTER=yes
```

```
HOSTNAME=xion
```

```
[root@xion /etc]#
```

El siguiente comando muestra la configuración inicial de la tarjeta eth0.

#### Comando a ejecutar en la Consola de GNU/Linux para Compatibilidad de IPv6

```
[root@xion /etc]# more /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
# Silicon Integrated Systems [SiS] SiS900 PCI Fast Ethernet
```

```
DEVICE=eth0
```

```
HWADDR=00:d0:09:9c:52:e7
```

```
BOOTPROTO=none
```

```
ONBOOT=yes
```

```
DHCP_HOSTNAME=xion
```

```
IPADDR=192.168.1.1
```

```
NETMASK=255.255.255.0
```

```
GATEWAY=192.168.1.1
```

```
[root@xion /etc]#
```

Modificamos el archivo de configuración de eth0, añadiendo las siguientes líneas:

#### Comando a ejecutar en la Consola de GNU/Linux para Compatibilidad de IPv6

```
[root@xion /etc]# gedit /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
[root@xion /etc]# more /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
# Silicon Integrated Systems [SiS] SiS900 PCI Fast Ethernet
```

```
DEVICE=eth0
```

```
HWADDR=00:d0:09:9c:52:e7
BOOTPROTO=none
ONBOOT=yes
DHCP_HOSTNAME=xion
IPADDR=192.168.1.1
NETMASK=255.255.255.0
GATEWAY=192.168.1.1
IPV6ADDR=2001:db8:fedc:cd::1
IPV6ADDR_SECONDARIES="2001:db8:fedc:abcd::1"
TYPE=Ethernet
USERCTL=no
IPV6INIT=yes
PEERDNS=yes
[root@xion /etc]#
```

Habilitamos la configuración anterior ejecutando el siguiente comando:

#### Comando a ejecutar en la Consola de GNU/Linux para Compatibilidad de IPv6

```
[root@xion /etc]# ifup eth0
```

Reiniciamos la red ejecutando el siguiente comando:

#### Comando a ejecutar en la Consola de GNU/Linux para Compatibilidad de IPv6

```
[root@xion /etc]# service network restart
```

Hacemos ping6 a la dirección IPv6 de la tarjeta eth0, para comprobar el funcionamiento:

### Comando a ejecutar en la Consola de GNU/Linux para Compatibilidad de IPv6

```
[root@xion /etc]# ping6 2001:db8:fedc:cdef::1
```

Podemos activar o desactivar ip6table:

✚ Con el siguiente comando desactivamos ip6table.

### Comando a ejecutar en la Consola de GNU/Linux para Compatibilidad de IPv6

```
[root@xion /etc]# chkconfig ip6tables off
```

✚ Con el siguiente comando activamos ip6table.

### Comando a ejecutar en la Consola de GNU/Linux para Compatibilidad de IPv6

```
[root@xion /etc]# chkconfig ip6tables on
```

## Configuración del Túnel IPv4.

A continuación se muestran las direcciones IPv4 para la creación de un Túnel.

### Configuración Red 192.168.1.0 en el Punto A – Soporte IPv4

```
network 192.168.1.0  
netmask 255.255.255.0  
router 192.168.1.1
```

### Configuración Red 122.16.17.18 en el Punto C (Intermedio) – Soporte IPv4

El Router tiene la IP: 172.16.17.18

### Configuración Red 192.168.2.0 en el Punto B – Soporte IPv4

```
network 192.168.2.0
netmask 255.255.255.0
router 192.168.2.1
```

### Configuración Red 172.19.20.21 en el Punto C (Intermedio) – Soporte IPv4

El Router tiene la IP: 172.19.20.21

En la interfaz del router del punto A, haremos lo siguiente:

### Configuración en la Interfaz del Router en el Punto A

```
ip tunnel add netb mode gre remote 172.19.20.21 local 172.16.17.18 ttl 255
ip link set netb up
ip addr add 192.168.1.1 dev netb
ip route add 192.168.2.0/24 dev netb
```

En la interfaz del router del punto B, haremos lo siguiente:

### Configuración en la Interfaz del Router en el Punto B

```
ip tunnel add neta mode gre remote 172.16.17.18 local 172.19.20.21 ttl 255
ip link set neta up
ip addr add 192.168.2.1 dev neta
ip route add 192.168.1.0/24 dev neta
```

Cuando se desee eliminar el túnel del punto A, ejecute lo siguiente:

### Eliminación del Túnel en el Punto A

```
ip link set netb down  
ip tunnel del netb
```

Cuando se desee eliminar el túnel del punto B, ejecute lo siguiente:

### Eliminación del Túnel en el Punto B

```
ip link set neta down  
ip tunnel del neta
```

## Configuración del Túnel IPv6.

Se tiene la red IPv6 3ffe:406:5:1:5:a:2:1/96 (IPv4 172.16.17.18 y del router 172.22.23.24), ejecute los siguientes comandos.

### Ejecute los siguientes comandos para Configurar un Túnel IPv6

```
ip tunnel add sixbone mode sit remote 172.22.23.24 local 172.16.17.18 ttl 255  
ip link set sixbone up  
ip addr add 3ffe:406:5:1:5:a:2:1/96 dev sixbone  
ip route add 3ffe::/15 dev sixbone
```

## Creación de Túnel en IPv6IPv4 en Windows.

Para la creación del túnel IPv6IPv4 se debe tener en cuenta el Service Pack de Microsoft Windows, de preferencia debe ser Service Pack 2. La configuración que se presenta a continuación fue probada en un computador con Microsoft Windows XP con Service Pack 2.

Utilice la consola de Microsoft Windows para configurar el túnel. Creación de túnel automático - Interfaz 2.

#### Ejecute los siguientes comandos para Configurar un IPv6 en Windows

```
c:\> ipv6.exe install
c:\> ipv6.exe rtu ::/0 2/::192.168.2.3
c:\> ipv6.exe adu 2/2001:db8:fedc:cdea::6
c:\> ipv6 if
```

Asignación de una dirección IPv6 a la tarjeta local. Revisar que la interfaz esta conectada a la red.

#### Ejecute los siguientes comandos para Configurar un IPv6 en Windows

```
c:\> netsh
netsh> interface ipv6 add address interface=5 address=2001:db8:fedc:cdea::5
netsh> interface ipv6 add route ::/0 5 2001:db8:fedc:cdea::1 publish=yes
netsh> interface ipv6 show address
netsh> interface ipv6 del address interface=5 address=2001:db8:fedc:cdea::5
netsh> exit
c:\> ping6 2001:db8:fedc:cdea::5
```

Especificación para el Router.

#### Ejecute los siguientes comandos para Configurar un IPv6 en Windows

```
c:\> ipv6.exe rtu ::/0 2/::192.168.1.2
c:\> ipv6.exe adu 2/2001:db8:fedc:cdef::1
c:\> ipv6.exe rtu 2001:db8:fedc:cdef::1/64 4 pub life 86400
c:\> ipv6.exe ifc 2 forw
c:\> ipv6.exe ifc 3 forw
c:\> ipv6.exe ifc 4 forw adv
```

## Ejemplos de Archivos de Configuración de Router Quagga.

A continuación se muestra el archivo de configuración del servicio Zebra.

### Archivo de Configuración zebra.conf

```
!  
! Zebra configuration saved from vty  
! 2007/07/19 19:09:10  
!  
hostname xion  
!  
interface eth0  
link-detect  
ip address 192.168.1.0/24  
ipv6 address 2001:db8:fedc:cdef::1/64  
ipv6 nd suppress-ra  
!  
interface eth1  
link-detect  
ip address 192.168.2.0/24  
ipv6 address 2001:db8:fedc:cdef::2/64  
ipv6 nd suppress-ra  
!  
interface lo  
!  
interface sit0  
ipv6 nd suppress-ra  
!  
interface sit1  
ipv6 nd suppress-ra  
!  
interface sit2  
ipv6 nd suppress-ra  
!  
ip route 192.168.1.0/24 192.168.1.2  
ip route 192.168.2.0/24 192.168.2.2  
!  
!  
line vty  
!
```



A continuación se muestra el archivo de configuración del servicio Riptd, parte del software Quagga.

#### Archivo de Configuración ripd.conf

```
!  
! Zebra configuration saved from vty  
! 2007/07/19 19:09:10  
!  
hostname xion  
password zebra  
log stdout  
!  
router rip  
version 2  
network 192.168.1.0/24  
network 192.168.2.0/24  
passive-interface default  
no passive-interface eth1  
no passive-interface eth0  
route 192.168.1.0/24  
route 192.168.2.0/24  
!  
line vty  
!
```



## Bibliografía

Se ha considerado como referencia para la elaboración del presente documento la información de las siguientes fuentes bibliográficas:

- ✚ Tutoriales Website de Red Hat Linux Enterprise Europa.  
<http://www.europe.redhat.com/software/rhel/as/>
- ✚ Tutorial Introducción Básica a GNU/Linux  
Jorge Gajón. – [gajon@gajon.org](mailto:gajon@gajon.org)
- ✚ Libro Hackers en Linux - Secretos y Soluciones para la Seguridad de Linux.  
Brian Hactch, James Lee y George Kurtz.
- ✚ Website de Difusión de Información de Software Libre - Xion Ecuador.  
<http://xionecuador.bravehost.com>  
Franklin Hermel Coloma Ramírez – [foxnet\\_xtrem@yahoo.es](mailto:foxnet_xtrem@yahoo.es)
- ✚ Informes de Prácticas de Laboratorio de Intranets realizadas en la Escuela de Ingeniería Electrónica y Tecnología en Computación - Escuela Superior Politécnica de Chimborazo.  
<http://fie.esPOCH.edu.ec>, <http://www.esPOCH.edu.ec>
- ✚ Informe de Prácticas Pre-profesionales realizadas en la empresa ISP Fastnet Cía. Ltda., de la ciudad de Riobamba.  
<http://www.fastnet.net.ec>
- ✚ Documentos elaborados para dictar clases teórico- prácticas en la Colaboración de Asistente de Cátedra de Sistemas Operativos en el año 2003, en la Escuela de Ingeniería Electrónica y Tecnología en Computación – Escuela Superior Politécnica de Chimborazo.

## Biografía

### INFORMACIÓN PERSONAL

<b>NOMBRES Y APELLIDOS:</b>	Franklin Hermel Coloma Ramírez	
<b>ESTADO CIVIL:</b>	Soltero	
<b>EDAD:</b>	27 años	
<b>LUGAR DE NACIMIENTO:</b>	DE San Miguel, Provincia de Bolívar - Ecuador	
<b>FECHA DE NACIMIENTO:</b>	DE 12 de Septiembre de 1980	
<b>LUGAR DE RESIDENCIA:</b>	DE Riobamba, Provincia de Chimborazo - Ecuador	
<b>DIRECCIÓN:</b>	Duchicela 22-45 y Daniel león Borja	
<b>CELULAR:</b>	59392295787	
<b>E-MAIL:</b>	<a href="mailto:foxnet_xtrem@yahoo.es">foxnet_xtrem@yahoo.es</a>	
<b>WEBSITE:</b>	<a href="http://xionecuador.bravehost.com">http://xionecuador.bravehost.com</a>	
<b>TÍTULOS SECUNDARIOS:</b>	Técnico en Manejo de Equipos de Computo Técnico Industrial en Electrónica	
<b>TÍTULO UNIVERSITARIO:</b>	Egresado de la Escuela de Ingeniería Electrónica - Escuela Superior Politécnica de Chimborazo – Riobamba Provincia de Chimborazo - Ecuador	

### Miembro de:

Ecuallug.org - Comunidad Ecuatoriana de Software Libre - <http://www.ecuallug.org>

Safecreative.org - Organización de Protección de propiedad Intelectual

Usuario SafeCreative: <http://www.safecreative.org/user/0710150019004>

Obra Registrada Xion GNU Linux: <http://www.safecreative.org/work/0801100372895>

Com-SL.org - Comunidad de Software Libre de Cuba - <http://www.com-sl.org/>

Voip-info.org - Wiki sobre VoIP Asterisk - <http://www.voip-info.org/wiki/view/Asterisk>

Asterix.org – Distribución GNU/Linux para IPBX - <http://www.asterisk.org>

Elastix.org – Distribución GNU/Linux Ecuatoriana para Voz sobre IP - <http://www.elastix.org>

Desktoptwo.com - GNU/Linux Virtual - Acceso Web a servidor GNU/Linux Virtual - <https://desktoptwo.com/>

Xing.com – Grupos de Profesionales - <https://www.xing.com/>



## Índice

Introducción . . . . .	2
Objetivos . . . . .	3
Shells Usados en GNU/Linux. . . . .	4
Comandos más Importantes en GNU/Linux. . . . .	5
Gestión de Usuarios en GNU/Linux. . . . .	17
Súper Usuario o root. ....	17
Usuario Normal o user. ....	17
Usuarios del Sistema o user systems. ....	17
Permisos. ....	17
Cambio de Permisos. ....	19
Cambio de Dueños. ....	20
Cambio de Atributos de un Archivo. ....	21
Creación de Cuentas de Usuarios en GNU/Linux. ....	22
Estructura de Directorios en GNU/Linux. . . . .	25
Directorios. ....	25
Dispositivos. ....	27
Sistema de Booteo en GNU/Linux. . . . .	29
Sistemas de Archivos. . . . .	31
Tipos de Sistemas de Archivos. ....	31
Todo es un Archivo. ....	32
Los Diferentes Tipos de Archivos. ....	32
I-nodos . . . . .	33
Servidor Samba SMB (Server Message Block). . . . .	35
Software Requerido. ....	35
Procedimientos. ....	36
Servidor DNS (Domain Name Server). . . . .	41
Dominio. ....	41
Configuración del Servidor DNS. ....	42
Servidor DHCP (Dinamic Host Configuration Protocol). . . . .	50
Configuración de un Servidor DHCP. ....	50
Archivo de Configuración. ....	50
Base de Datos de Arrendamiento. ....	54
Arranque y Parada del Servidor. ....	55
Agente de Transmisión DHCP. ....	56
Configuración del Cliente DHCP. ....	57
Compartición de Archivos con el Servidor vsftpd. . . . .	58
Protocolo FTP (File Transfer Protocol). ....	58
Servidor vsftpd (Very Secure FTP Daemon). ....	58
Instalación de vsftpd Utilizando Yum y Up2Date. ....	58
Ficheros de Configuración. ....	58
Inicio, Detención y Reinicio del Servicio vsftpd. ....	61



Compartición de Archivos usando el Servidor NFS (Network File System) . . . . .	63
Compartición del Disco Duro.....	63
Seguridad en NFS. ....	64
Iniciar la Compartición de Archivos. ....	65
Configuración del Cliente NFS.....	65
Instalación y Configuración de Emulador Wine . . . . .	66
Información del Núcleo de GNU/Linux.....	66
Sistemas de Ficheros Soportados.....	67
Requisitos Básicos .....	67
Requisitos de Herramienta de Construcción .....	67
Compilación de Ficheros de Wine .....	67
Configuración de Wine .....	68
Ejecución de Programas.....	68
Instalación de Wine desde Paquetes RPM.....	69
Configuración del Router Quagga en CentOS Linux 5.0 . . . . .	70
Configuración Básica de Router Quagga.....	70
Ejemplo de Configuración del Router xion - Red 192.168.1.0.....	71
Configuración de Router Quagga en GNU/Linux CentOS 5.0.....	71
Compatibilidad de la Tarjeta de Red Eth0 con IPv6. ....	80
Configuración del Túnel IPv4.....	83
Configuración del Túnel IPv6.....	85
Creación de Túnel en IPv6IPv4 en Windows.....	85
Ejemplos de Archivos de Configuración de Router Quagga.....	87
Bibliografía . . . . .	89
Biografía . . . . .	90
Índice . . . . .	91