Servidor De Raíces NFS Mini-COMO

Ofer Maor, ofer@hadar.co.il
Traducido por Rodolfo Pilas, rodolfo@linux.org.uy
v3, 5 Diciembre 1996

El objetivo de este Mini-COMO es explicar cómo crear directorios en un servidor que los clientes montarán como su directorio raíz por NFS.

Índice General

1 Derechos

Se puede encontrar la última versión de este Mini-COMO (HTML o texto) en
`http://www.hadar.co.il/ofer/nfs-root-client/`

COMO e imponerle restricciones adicionales. Pueden ser concedidas excepciones a estas reglas bajo ciertas condiciones; por favor, contacte con el coordinador de los COMO de Linux en la dirección que se indica más adelante.

En resumen, deseamos promover la diseminación de esta información a través de tantos canales como sea posible. Sin embargo, se desea retener los derechos sobre los documentos COMO, y nos gustaría estar informados de cualquier plan de redistribuir los COMOs.

Si tiene preguntas, por favor, contacte con Ofer Maor `ofer@hadar.co.il`, o Greg Hankins, el coordinador de los Linux COMO, `gregh@sunsite.unc.edu` vía correo electrónico.

Si tiene cualquier cosa para agregar a este Mini-COMO, por favor envíe un email al autor (Ofer Maor `ofer@hadar.co.il`), con la información. Toda información nueva que sea relevante será bienvenida.

1.1 Agradecimientos

Me gustaría expresar mi agradecimiento al autor de *Linux Con Raíz NFS Mini-COMO*, Andreas Kostyrca, `andreas@medman.ag.or.at`. A partir de su Mini-COMO pude dar los primeros pasos para montar mis clientes con directorio raíz por NFS. Este Mini-Como no pretende, de ninguna forma, suplantar dicho trabajo; lo amplía usando mi experiencia en este proceso.

También quiero agradecer a Mark Kushinsky, `mark026@ibm.net` por pulir el inglés y las expresiones de este Como, hasta lograr hacerlo mucho más legible.

2 Prefacio

Este Mini-Como ha sido escrito para ayudar a las personas que desean preparar los directorios de un servidor para ser montados como directorio raíz mediante por clientes NFS.

Por favor, tenga en cuenta que existen muchas maneras de lograr esto, dependiendo de las necesidades y de sus intenciones. Si cada cliente es individual, y cada uno de ellos tiene su propio usuario y administrador, será necesario que una parte significativa de los directorios de cada cliente no sea compartida con otro cliente. Por otro lado, si el cliente es accesible a muchos usuarios y están todos administrados por la misma persona (como en un aula de clases), se buscará que la mayor cantidad posible de archivos estén compartidos a los efectos de hacer la administración más manejable. Este Como se enfoca en esta segunda opción.

2.1 Nota

Cuando se construye el que será el directorio raíz del cliente, y se trata de ocupar un espacio mínimo, se debe atender principalmente a qué archivos pueden ser compartidos o montados desde el servidor. En este Como recomiendo la configuración de un cliente basado en mi experiencia. Pero antes de empezar tenga en cuenta que:

- Este Mini-COMO no explica la importación del directorio raíz por NFS. Refiérase al *Linux-Con-Raíz-NFS Mini-COMO* si necesita mayor información sobre esto.
- Yo baso la mayoría de las configuraciones de mis clientes en montajes y enlaces simbólicos. Muchos de los enlaces simbólicos pueden ser reemplazados por enlaces rígidos (*hardlinks*). Uno puede elegir de acuerdo a sus preferencias personales. Utilizar enlaces rígidos sobre lo montado en vez de enlaces simbólicos tiene sus ventajas, pero puede causar confusión. Un archivo no es borrado definitivamente hasta que todos los enlaces rígidos sean eliminados. Por esto, en previsión del caso en que algún archivo sea puesto al día y el enlace rígido todavía se refiera a la versión vieja, es necesario tener especial cuidado en recordar cada enlace que utilice.

- Cuando se monta la información desde el servidor, pueden seguirse dos métodos. El primero (y más común), es montar la totalidad del directorio raíz del servidor bajo un directorio local, y cambiar simplemente la trayectoria de búsqueda o enlazar los directorios relevantes allí. Personalmente no me gusta montar la partición raíz del servidor en los clientes. Por ello, este Como sugiere un método para montar los directorios relevantes del servidor en los lugares apropiados del sistema.
- Este Como está basado en mi experiencia en la preparación de directorios para clientes basados en la distribución Slackware 3.1. Algunas cosas pueden resultar diferentes (especialmente en los archivos `rc.*`), para usuarios de otras distribuciones, sin embargo los métodos y conceptos no varían.

3 Creación del directorio raíz de los clientes

3.1 Creación del árbol de directorios

Antes que nada, es necesario crear la estructura de directorios en sí. Yo creo todos los clientes bajo `/clients/nombre-maquina` y usaré esto en los ejemplos que cito a continuación. No obstante, se puede cambiar por cualquier otro nombre. La primera etapa por tanto, consiste en crear los directorios relevantes en el que será directorio raíz del cliente. Se deben crear los siguientes directorios:

```
/bin
/dev
/etc
/home
/lib
/mnt
/proc
/sbin
/server
/tmp
/usr
/var
```

y cualquier otro directorio que desee tener en su sistema.

Los directorios locales `/proc` y `/dev` deben ser utilizados separadamente por cada máquina, a diferencia del resto de directorios, que pueden estar compartidos parcial o completamente con el resto de los clientes.

3.2 Creación del sistema de archivos mínimo para arrancar

3.2.1 Creación del directorio `/dev`

Si bien el directorio `/dev` puede ser compartido, es mejor crearlo separado para cada cliente. Puede crear dicho directorio en el cliente con la macro `MAKEDEV`; sin embargo, en muchos casos resulta más simple copiarlo del servidor:

```
cp -a /dev /clients/nombre-maquina
```

Debe recordar que `/dev/mouse`, `/dev/cdrom` y `/dev/modem` son enlaces simbólicos a los dispositivos actuales y por tanto, asegurarse de que queden enlazados correctamente para apuntar al hardware correspondiente del cliente.

3.2.2 Copia de los binarios necesarios

Aunque montemos todo desde el servidor, existe un grupo de binarios que necesitamos copiar para cada cliente. Para empezar, necesitamos "`init`", ya que nuestro sistema no podrá ejecutar nada antes de inicializarse (como su autor indica en los comentarios sobre el programa).

Por tanto, primero deberá copiar `/sbin/init` al directorio `/sbin` del cliente y como ejecutará `rc.S`, necesitará copiar `/bin/sh` al directorio `/bin` del cliente. Para poder montar necesitará copiar `/sbin/mount` al directorio `/sbin` del cliente.

Esto es lo mínimo, asumiendo que la primera línea de su `rc.S` es `mount -av`. No obstante, recomiendo copiar algunos archivos más: `update, ls, rm, cp` y `umount`, y así disponer de las herramientas básicas en caso de que el cliente tenga problemas para montar. Si elige tener el swap habilitado antes de montar, también tendría que copiar el binario `swapon`.

Ya que muchos de estos binarios están por defecto compilados dinámicamente, necesitará copiar una pequeña parte de `/lib`:

```
cp -a /lib/ld.* /lib/libc.* /lib/libcurses.* /client/nombre-maquina/lib
```

También es posible enlazar los binarios a sí mismos, en lugar de copiarlos. Por favor, lea mis comentarios de la sección 2.1 de este documento.

Tenga en cuenta que estamos asumiendo que el núcleo ha recibido todos los parámetros de red mientras se iniciaba. Si planea utilizar `rarp` o `bootp`, posiblemente necesite copiar también los binarios relevantes para esto.

Generalmente, se necesitará tener un mínimo de archivos disponibles para configurar la red y ejecutar `rc.S` hasta el momento en que se monta el resto del sistema de archivos.

3.2.3 El directorio `/var`

El directorio `/var`, en muchos casos, puede ser separado para cada cliente. Sin embargo, gran parte de la información puede estar compartida. Cree bajo el directorio del servidor un directorio llamado `var`. Montaremos el directorio `/var` de allí. Para crear el directorio local `/var`, simplemente escriba:

```
cp -a /var /clients/nombre-maquina
```

Ahora, tiene la posibilidad de elegir lo que desea separar, y lo que desea compartir. Para compartir cualquier directorio/archivo, elimínelo simplemente del directorio `/var` del cliente y enlácelo en al directorio `/server/var`. Tenga en cuenta que debe enlazar tanto para `/server/var` como para `../server/var` pero **NO** para `/clients/nombre-maquina/server/var` ya que esto no funcionará cuando cambie el directorio raíz en cada cliente.

Generalmente recomiendo separar `/var/run`, `/var/lock`, `/var/spool` y `/var/log`.

3.2.4 El resto de los directorios

- `/etc` se explica en la siguiente sección.
- `/mnt` y `/proc` son para propósitos locales de cada cliente
- `/usr` y `/home` son solamente para montar
- `/tmp` depende de Usted. Se puede crear un directorio `tmp` diferente para cada cliente, o crear un directorio `/clients/tmp` y montarlo para cada cliente en `/tmp`. Yo recomiendo darle a cada cliente un directorio `tmp` separado.

3.3 Preparación del directorio /etc y configuración de los clientes

Tenga en cuenta lo siguiente: esta sección se refiere a la creación del directorio `etc` que está preferentemente compartido entre los clientes. Si sus clientes sin disco tienen administradores de sistema diferentes, lo mejor es configurar un directorio `etc` para cada cliente.

3.3.1 Creación del árbol etc de los clientes

Aunque separemos el directorio `etc` para los clientes, mantendremos compartida una gran parte de los archivos que allí se encuentran. Creo que compartir los archivos con el `/etc` del servidor es mala idea, y recomiendo crear un directorio `/clients/etc` que contenga la información compartida entre los clientes. Para arrancar con esto, simplemente copie el contenido del `/etc` del servidor en el directorio `/clients/etc`.

Se deberá colocar en este directorio todos los archivos de configuración que no sean específicos para una máquina, como pueden ser `motd`, `issue`, etc. y no los que sean específicos del cliente, como `inittab` o `fstab`.

El cambio más importante que deberemos realizar estará bajo el directorio `rc.d`. Primero, deberemos cambiar `rc.inet1` para que refleje la configuración local. Yo paso al núcleo todos los parámetros de red usando LILO/Loadlin, por lo que quito todo lo relacionado del archivo `rc.inet1`. Lo único que dejo allí son los binarios `ifconfig` y `route` del cliente. Si utiliza `rarp` o `bootp`, necesitará crearlo acorde a esto.

En segundo lugar, deberá editar su `rc.S` y eliminar todas las partes responsables de la comprobación por parte de `fsck` que tiene lugar cuando arranca el servidor. Debe encontrar la línea que monta su `fstab`, que se parecerá a:

```
mount -avt nonfs
```

El parámetro `-t nonfs` se debe a que los clientes normales ejecutan primero `rc.S`, usando después `rc.inet1` para configurar la red. Como esto provoca que no sea montada ninguna partition NFS, la línea es ignorada, y por ello, se debe cambiar a `mount -av`. Si necesita ejecutar `rarp/bootp` para configurar la red, hágalo en `rc.S` (o incluya una llamada a alguna macro desde `rc.S`), antes de montar, y asegúrese de que sus directorios `/bin` y `/sbin` tengan disponibles los archivos necesarios.

Una vez se lleva a cabo `mount -av`, tendrá un sistema de archivos funcionando. Cree un `fstab` general, luego puede copiarlo en cada cliente. Su `fstab` debe ser aproximadamente como este:

```
server/nfs                                    default   1 1
server:/bin          /bin           nfs       default   1 1
server:/usr          /usr           nfs       default   1 1
server:/sbin         /sbin          nfs       default   1 1
server:/home         /home          nfs       default   1 1
server:/lib          /lib           nfs       default   1 1
server:/clients/etc  /server/etc    nfs       default   1 1
server:/clients/var  /server/var    nfs       default   1 1
none                 /proc          proc      default   1 1
```

Asegúrese también de que el `/etc/exports` del servidor sea algo como:

```
/clients/<nombre-m\'{a}quina>         maquina.dominio(rw,no_root_squash)
/clients/etc                  maquina.dominio(ro,no_root_squash)
/clients/var                  maquina.dominio(ro,no_root_squash)
/usr                          maquina.dominio(ro,no_root_squash)
/sbin                         maquina.dominio(ro,no_root_squash)
/bin                          maquina.dominio(ro,no_root_squash)
/lib                          maquina.dominio(ro,no_root_squash)
/home                         maquina.dominio(ro,no_root_squash)
```

Con excepción de la primera línea, que debe ser única para cada máquina, el resto de las líneas pueden ser reemplazadas por nombres que engloben a todas las máquinas (por ejemplo `pc*.dominio`). Le sugiero crear la mayoría de los directorios como sólo lectura, aunque esto es decisión suya. La indicación **`no_root_squash`** hará que los usuarios administradores de los clientes tengan también los permisos vigentes de root sobre `nfsd`. Revise la página `man exports(5)`. Si desea que los usuarios puedan ejecutar `passwd` desde los clientes, asegúrese que `/etc` esté con permisos `rw` y no `ro`. Aunque esto no es obligatorio.

Por favor, tenga en cuenta otro detalle en el archivo `rc.S`. En Slackware, por defecto, se crea un nuevo `/etc/issue` y `/etc/motd` cada vez que se ejecuta. Esta función **DEBE** ser deshabilitada si dichos archivos son montados como sólo lectura desde el servidor, y recomiendo que estén deshabilitados en todos los casos.

Finalmente, si desea mantener la misma base de usuarios tanto en el servidor como en los clientes, deberá elegir entre:

1. usar NIS (Páginas Amarillas - revise el *NIS-Como*), cada cliente deberá disponer, por separado, de un `/etc/passwd` y un `/etc/group` tal y como lo recibe del servidor NIS.
2. En la mayoría de los casos, será suficiente con un simple enlace simbólico. Por lo tanto, necesitará o bien un enlace rígido (*hardlink*) de `/clients/etc/passwd` a `/etc/passwd`, o si prefiere, un enlace simbólico, enlazando `/etc/passwd` a `/clients/etc/passwd` (y no de la otra forma, ya que los clientes no montarán el directorio `/etc` del servidor). Haga lo mismo para `/etc/group`.

3.3.2 Creación del directorio `/etc` del cliente

Generalmente, la mayoría de los archivos en el `/etc` del cliente estarán enlazados simbólicamente con el directorio `/server/etc`. Sin embargo, algunos archivos son diferentes para cada máquina, y otros solamente tienen que estar allí cuando el núcleo los carga. Lo mínimo que se necesita en el directorio `/etc` es lo siguiente:

```
resolv.conf
hosts
inittab
rc.d/rc.S
fstab
```

Ya que estos cinco archivos pueden ser idénticos para todos los clientes, los puede simplemente enlazar rígidamente o copiarlos una y otra vez. No obstante, con los archivos `rc.S` y `fstab` debe tener presente que se necesita una copia separada para cada cliente. También necesitará separar el `/etc/HOSTNAME` de cada cliente. Yo personalmente recomiendo tener todos los archivos de `rc.d` separados para cada cliente, ya que la configuración de hardware puede variar de uno a otro.

Para cada cliente, agregue en el `fstab` una línea apropiada para el archivo de intercambio:

```
/dev/particion_swap          swap    swap    default 1 1
```

Con el resto de los archivos del `/etc` del cliente, puede o bien enlazarlos rígidamente a los archivos `/clients/etc/*` o bien simbólicamente con `/server/etc` (ya que es el punto de montaje de `/clients/etc/`)

Asegúrese de que su máquina puede resolver apropiadamente, tanto a través de `named` como de `etc/hosts`. No es mala idea mantener el número IP del servidor en `/etc/hosts`, en lugar de contar con resolverlo. Si confía resolverlo con `named` solamente, cualquier problema con `named` puede impedir que sus clientes arranquen.

3.3.3 Arrancando

Ahora, todo lo que necesita hacer es iniciar su máquina, cruzar los dedos y esperar que todo funcione como debe :-).

4 Creación de más clientes

Si ha seguido mis instrucciones, lo cual debería haberle resultado simple, cambie al directorio `/clients/` y escriba:

```
cp -a hostname1 hostname2
```

y asegúrese de revisar estos puntos:

los archivos `rc.d/*` coinciden con el hardware y la configuración de software de `/etc/HOSTNAME` es correcta, la línea del archivo de swap en `fstab` es correcta, y los enlaces simbólicos a `dev/mouse`, `dev/modem` y `dev/cdrom` son correctos.

Mucha suerte ...

5 Anexo: El INSFLUG

El *INSFLUG* forma parte del grupo internacional *Linux Documentation Project*, encargándose de las traducciones al castellano de los Howtos (Comos), así como la producción de documentos originales en aquellos casos en los que no existe análogo en inglés.

En el **INSFLUG** se orienta preferentemente a la traducción de documentos breves, como los *COMOs* y *PUFs* (**P**reguntas de **U**so **F**recuente, las *FAQs*. `:)`), etc.

Diríjase a la sede del INSFLUG para más información al respecto.

En la sede del INSFLUG encontrará siempre las **últimas** versiones de las traducciones: `www.insflug.org`. Asegúrese de comprobar cuál es la última versión disponible en el Insflug antes de bajar un documento de un servidor réplica.

Se proporciona también una lista de los servidores réplica (*mirror*) del Insflug más cercanos a Vd., e información relativa a otros recursos en castellano.

Francisco José Montilla, `pacopepe@insflug.org`.