

# Instalación de un sistema Debian chroot

Colin Walters

walters@debian.org

## Tabla de contenidos

1. Introducción .....	3
1.1. Requisitos .....	3
2. Creando el chroot .....	3
3. Estableciendo el acceso o login .....	4
4. Configurando gdm .....	4

## **Instalación de un sistema Debian chroot**

por Colin Walters

Javier

Linares Sánchez

Traducción al castellano

Teófilo

Ruiz Suárez

Traducción al castellano

# 1. Introducción

Con frecuencia, como desarrollador Debian, necesito probar los paquetes antes de subirlos. Sin embargo, si mis paquetes tienen fallos importantes que hicieran cosas equivalentes a **rm -rf /**, no es agradable probarlos en mi instalación principal.

Con la combinación de los programas **chroot** y **debootstrap**, es posible configurar un sistema Debian aislado en el cual poder entrar e instalar el paquete, sin temor alguno a destruir su sistema principal. Actualmente, estoy usando esta configuración para tener GNOME 2 y GNOME 1 instalados a través de paquetes Debian en diferentes chroots, para poder probar scripts para la transición de GNOME 1 a GNOME 2.

## 1.1. Requisitos

- chroot (parte del paquete *coreutils* en Debian *sarge* y *sid*)
- debootstrap
- login
- (*opcional*)gdm

## 2. Creando el chroot

A través de **debootstrap** puede configurar un entorno chroot para *woody*, *sarge* o *sid*. En adelante este documento asume que está creando un chroot para *sid*.

1. Cree un directorio, por ejemplo */sid/*. Si no elige ese nombre, asegúrese de sustituir el que seleccione por */sid/* en el resto de este documento.

```
mkdir /sid/
```

2. Es necesario instalar primero *woody* y, a continuación, actualizar a *sid*. *sid* cambia con suficiente periodicidad como para usar **debootstrap** con esta rama de Debian. Así que ejecutamos **debootstrap**:

```
debootstrap woody /sid/ http://ftp.debian.org/debian/
```

3. Establezca en el archivo */etc/fstab* de *su sistema principal* el montaje del sistema de fichero */proc* del entorno chroot.

```
echo proc-sid /sid/proc proc none 0 0 >> /etc/fstab  
mount /sid/proc
```

4. Configure la red en el chroot.

```
echo 127.0.0.1 localhost $(hostname) > /sid/etc/hosts
```

5. Ahora ya está listo para usar el chroot. Usaremos el programa *chroot* para acceder a él.

```
chroot /sid/ /bin/sh
```

6. Actualizamos la instalación de woody a sid. Si desea ejecutar woody debería saltarse este paso.

```
(echo deb http://ftp.debian.org/debian/ sid main; echo deb-src http://ftp.debian.org/debian/ sid  
apt-get update  
apt-get dist-upgrade
```

7. Llegados a este punto ya debería tener un sistema completamente usable en el que poder jugar sin miedo a interferir con su instalación Debian principal.

### 3. Estableciendo el acceso o login

Ejecutar **chroot /sid/** es sencillo, pero mantiene todo tipo de variables de entorno alrededor que tal vez no desee, y algunas otras cosas más. Una aproximación mucho mejor es ejecutar otro proceso login en un terminal virtual diferente, donde poder entrar en el chroot de forma directa.

1. En su sistema principal, edite el archivo `/etc/inittab`, buscando y modificando las siguientes líneas:

```
# Note que en la mayoría de los sistemas Debian tty7 se usa para el sistema  
# de ventanas X (X Windows System). Si desea, por tanto, añadir más accesos  
# en modo texto (FIXME getty's) siga añadiendo líneas como esta pero  
# sáltese el tty7 si ejecuta X.  
#  
1:2345:respawn:/sbin/getty 38400 tty1  
2:23:respawn:/sbin/getty 38400 tty2  
3:23:respawn:/sbin/getty 38400 tty3  
4:23:respawn:/sbin/getty 38400 tty4  
5:23:respawn:/sbin/getty 38400 tty5  
6:23:respawn:/sbin/getty 38400 tty6
```

2. Ahora, añade una línea similar a la siguiente:

```
8:23:respawn:chroot /sid/ /sbin/getty 38400 tty8
```

El "8" hace referencia a la terminal en la cual se ejecutará el nuevo acceso (login). Puede elegir otra si así lo desea.

3. Reiniciamos init:

```
init q
```

## 4. Configurando gdm

¿Desearía ejecutar X y GNOME en su chroot? ¡Es totalmente posible! El siguiente ejemplo hará que GDM se ejecute en la terminal virtual 8; debería seleccionar otra (como 9) si ya ha configurado login para ejecutarse en vt8 en el ejemplo anterior. A continuación encontrará lo que hay que hacer:

1. (entre en su chroot, ya sea con **chroot /sid/** o con el login configurado previamente)

```
apt-get install gdm gnome x-window-system
```

2. Copie su `/etc/X11/XF86Config-4` desde el sistema principal al entorno chroot, como:

```
cp /etc/X11/XF86Config-4 /sid/etc/X11/XF86Config-4
```

3. A continuación entre en el chroot de nuevo y edite el archivo `/etc/gdm/gdm.conf` dentro de él. Baje hasta la sección `[servers]`. Verá una línea parecida a:

```
0=Standard vt7
```

Cambie esa línea a:

```
0=Standard vt8
```

De nuevo, sustituya 8 por la terminal virtual que desee en lugar de “8”.

4. Y, todavía dentro del chroot, ejecute:

```
/etc/init.d/gdm start
```

Ahora puede conmutar entre el entorno X de su chroot y el entorno X de su sistema principal, simplemente conmutando como lo haría entre terminales virtuales en Linux; por ejemplo, usando **Ctrl+Alt+F7** y **Ctrl+Alt+F8**. ¡Diviértase!