Installing new Debian systems with debootstrap

*Posted by **Steve** on Thu 3 Aug 2006 at 09:39*
*Tags: **debian-specific commands**, **debootstrap***

When it comes to installing new installations of Debian GNU/Linux there is one tool which should not be ignored. Whether you're dealing with a real system, or a virtualised one, the `debootstrap` tool is ideal for quickly installing new Debian environments.

Put simply the **debootstrap package** allows you to install a fresh copy of Debian GNU/Linux into a directory. This new installation will have all the basic packages and binaries which you'd expect to be present such as:

- `/bin/ls`
- `/usr/bin/id`

Using `debootstrap` might seem a little bit strange if you're only used to the Debian installer, but it is a very useful tool for performing installations if you've got something like a **LiveCD** which supports your hardware. The process would be go something like this:

- Boot your system with the liveCD
- Verify that you can recognise basic hardware.
  - e.g. SATA drives, etc.
- Partition / format drive(s)
- Use `debootstrap` to install new system into the `/` partition
- Fixup `/etc/fstab`, etc.
- Reboot into your new system.

If you already have a system installed then you can use `debootstrap` to setup a "chroot" installation of Debian for testing purposes.

Chroot?

> A chroot on Unix operating systems is an operation which changes the root directory. It affects only the current process and its children. The term "chroot" itself can refer either to the chroot(2) system call or the chroot(8) wrapper program.
>
> A program that is re-rooted to another directory cannot access files outside that directory. This provides a convenient way to sandbox an untrusted, untested or otherwise dangerous program. It is also a simple kind of "jail" mechanism.

To get started with `debootstrap` you must first install it:

```
root@desktop:~# apt-get install debootstrap
```

Once installed you can start experimenting with it. A simple invocation would look like this:

```
root@desktop:# mkdir /tmp/blah
root@desktop:# debootstrap sarge /tmp/blah/
```

Here we're telling `debootstrap` that we wish to have a fresh installation of Sarge made into the directory `/tmp/blah`. The process will take a while since it will involve downloading packages from the Debian mirrors, as you can see from the following output:

```
root@desktop:/tmp# debootstrap sarge /tmp/blah/
I: Retrieving Release
I: Retrieving Packages
I: Validating Packages
I: Resolving dependencies of required packages...
I: Resolving dependencies of base packages...
I: Found additional required dependencies: libtext-iconv-perl zlib1g
I: Checking component main on http://ftp.debian.org/debian...
I: Retrieving adduser
I: Validating adduser
I: Retrieving apt
I: Validating apt
....
```

Once the system has finished you can use the `chroot` command to "jail" yourself inside it:

```
root@desktop:~# chroot /tmp/blah
root@desktop:/#
root@desktop:/#
root@desktop:/# ls
bin   dev  home    lib    mnt  proc  sbin  sys  usr
boot  etc  initrd  media  opt  root  srv   tmp  var

root@desktop:/#
root@desktop:/# apt-get clean
root@desktop:/#
root@desktop:/# du . --human-readable --total | grep total
111M    total
```

Here you can see that you have a minimal system which occupies just over 100Mb of space. Notice that your prompt still contains the name of your "real" system? Don't be tempted to change that - since the hostname change will take effect on your host system too.

Apart from the hostname issue the new installation is independent from your real system, it just shares the IP address. If you want to use this environment properly you'll need to make a couple of changes. (Exit from the chroot first):

```
root@desktop:~# mount proc /tmp/blah/proc/  -t proc
root@desktop:~# cp /etc/resolv.conf /tmp/blah/etc
root@desktop:~# cp /etc/hosts /tmp/blah/etc
root@desktop:~# chroot /tmp/blah/
root@desktop:/#
```

Since the new system will not be using PPP, it will have access to the network via the host system, you can tidy it up a little:

```
root@desktop:~# chroot /tmp/blah /usr/bin/dpkg --purge ppp pppoe pppconf
root@desktop:~# chroot /tmp/blah /usr/bin/apt-get install deborphan less
root@desktop:~# chroot /tmp/blah
root@desktop:/# deborphan
libpcap0.7
root@destkop:/# dpkg --purge libpcap0.7
```

Now we have a nice minimal installation which can be modified or purged whenever you're done with it. If you mounted `/proc` you should remove it first:

```
root@desktop:~# umount /tmp/blah/proc
root@desktop:~# rm -rf /tmp/blah/
```

Caching

If you're going to use this command often you'll most likely wish to use a local mirror, or even better a cache.

For example if you have the machine `itchy` running **an installation of `apt-proxy`** you can use that :

```
root@desktop:~# debootstrap sarge /tmp/blah/ http://itchy:9999/
```

This will download all the packages via your proxy, and the second time you execute it will give you a significant speedup (since the Debian packages will be coming from the cache).

So what can you use the combination of `chroot` and `debootstrap` for? Well the possibilities are endless! If you remember the previous article on **testing packages with pbuilder** you might see a connection. `pbuilder` allows you to setup pristine "environments" for testing Debian packages, and it does that by invoking `debootstrap` to create fresh installations of Debian into a directory.

So using debootstrap, as we've shown, you can test package installations and dependencies. You could even setup a new environment for each package build by hand if you were masochistic.

If you're using Xen or UML you'll find debootstrap is *the* way to install new instances of Debian. Make the installation then add in a /etc/fstab file and networking details in /etc/network/interfaces and you're done.

There are a lot of useful options which debootstrap has which we've not covered, to setup an arch, etc. See the manpage for details:

```
man debootstrap
```

Share / Save

Re: Installing new Debian systems with debootstrap

*Posted by **Anonymous** (212.219.xx.xx) on Thu 3 Aug 2006 at 11:10*

I would recommend apt-cacher instead of apt-proxy. As apt-proxy is currently broken in etch (**http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=375677**)

Re: Installing new Debian systems with debootstrap

*Posted by **Steve** (62.30.xx.xx) on Thu 3 Aug 2006 at 11:13*
*[**Send Message** | **View Steve's Scratchpad** | **View Weblogs**]*

True enough. The last time I installed my system I noticed this and installed **approx** instead.

(I chose that only because it is 99% compatible with apt-proxy and listens on the same port by default. This meant I didn't need to update any of my client machines.)

Steve

Re: Installing new Debian systems with debootstrap

*Posted by **mverwijs** (131.211.xx.xx) on Mon 7 Aug 2006 at 08:58*
*[**Send Message**]*

Well, I've just given up on the whole proxy thing. I just mirror the entire archive with rsync. It's what: 8.8GB?

This is a one-time download and a nightly rsync to keep up to date. The initial installation can also be done from a Debian DVD, to save on bandwidth.

Believe me: this is 100% accurate. Never a glitch.

Kind regards,

Maarten

Re: Installing new Debian systems with debootstrap

*Posted by **Anonymous** (203.206.xx.xx) on Thu 3 Aug 2006 at 22:08*

Yeah, I have noticed a few issues with apt-proxy. I must admit that apt-cacher is a bit slower for me and doesn't allow the control I would like but is a far cry from apt-proxy at the moment.

Re: Installing new Debian systems with debootstrap

*Posted by **Anonymous** (83.67.xx.xx) on Sat 3 Nov 2007 at 22:20*

If you use apt-cacher then use a command line like:

debootstrap sarge /scratch/xen/tst **http://apt-cacher-host:3142/ftp.uk.debian.org/debian/**

Re: Installing new Debian systems with debootstrap

*Posted by **Anonymous** (62.235.xx.xx) on Fri 4 Aug 2006 at 00:27*

Would be nice to know if there's a mechanism to "finish" the debootstrap install in the same way as a normal debian-installer install.

I often use debootstrap to install debian on machines that are not yet supported by the debian-installer (just yesterday I installed it on a system using VIA8237A SATA which isn't supported in any released kernel yet - as opposed to the normal VIA8237 which works with sata_via by default)

However, little niggles like timezones, locales, root password, user+groups, hosts, hostname, fstab, etc are left for you to do manually. I'm not always sure if I forgot anything or not.

Re: Installing new Debian systems with debootstrap

*Posted by **Eliteforce** (193.171.xx.xx) on Fri 4 Aug 2006 at 23:14*
*[**Send Message**]*

see **http://www.debian.org/releases/stable/i386/apcs04.html.en** for further details (a simple link instead of writing the whole article would have been sufficient i think :P)

Re: Installing new Debian systems with debootstrap

*Posted by **Steve** (62.30.xx.xx) on Sat 5 Aug 2006 at 02:28*
*[**Send Message** | **View Steve's Scratchpad** | **View Weblogs**]*

But `debootstrap` is useful for more than just the installer - and it might be new to people so I figure a summery is a good thing.

Besides this will be a nice introduction to the idea of installing into a directory for next weeks piece on `rpmstrap` ;)

Steve

Re: Installing new Debian systems with debootstrap

*Posted by **dkg** (216.254.xx.xx) on Sat 5 Aug 2006 at 19:06*
*[**Send Message** | **View dkg's Scratchpad** | **View Weblogs**]*

That's a useful link for remembering a lot of "the little cleanup stuff" that needs to be done after an install. But it's out-of-date for etch/sid, because it recommends using `base-config`. **As Joey Hess says**, `base-config` has been obsoleted, and it's not clear exactly what it should be replaced with.

A clear list of the steps that need to be taken instead would be a useful thing.

Re: Installing new Debian systems with debootstrap

*Posted by **peterhoeg** (193.163.xx.xx) on Mon 7 Aug 2006 at 11:37*
*[**Send Message**]*

Write a script.

I have some Xen instances that are done that way around. Actually debconf supports preseeding the answers to questions asked by packages as they are installed, but I never got into setting that up as it seemed like too much work. Much easier to simply write a script that does:

1. Create a fresh LV (alternatively use a loopback file if you are a heretic who does not believe in the wonders of LVM)
2. Mount LV (or loopback file)
3. Install using debootstrap
4. Set the required parameters
    1. hostname
    2. network details
    3. /etc/resolv.conf, /etc/hosts

    4. root password
    5. and a few others I cannot remember at the moment
5. Generate keys for sshd and cfengine as well as exchange them
6. Umount LV
7. Create xen config
8. Boot VM

These things are much easier to deal with through scripting in my oppinion. Especially if you have a thing like cfengine (as per Steve's excellent articles) to take care of everything after getting the basic system online.

Re: Installing new Debian systems with debootstrap

*Posted by **dkg** (216.254.xx.xx) on Mon 7 Aug 2006 at 13:49*
*[**Send Message** | **View dkg's Scratchpad** | **View Weblogs**]*

Yes, of course a script is the way to go. But it's your item 4.5 ("and a few others I cannot remember at the moment") that needs to be documented before a comprehensive script can be written! Would you mind publishing your xen-creation script? That would be helpful for a lot of folks here, i imagine.

And we're definitely in agreement: LVM is the way to go!

Re: Installing new Debian systems with debootstrap

*Posted by **peterhoeg** (193.163.xx.xx) on Wed 9 Aug 2006 at 15:06*
*[**Send Message**]*

I don't mind at all - there are 3 interesting scripts here:

- xen_make_base.sh - the script used to generate the initial image
- xen_config_base.sh - a basic config script called from a chroot in the image created above
- xen_new_host.sh - clones the base image, creates host specific info, xen config file and a few others.

  xen_make_bash.sh

```
#!/bin/sh
#
echo '$Id: xen_make_base.sh 72 2006-04-11 08:00:27Z admin $'

cnf=/hoeg/share/xen.conf
if [ -e $cnf ] ; then
    source $cnf
else
    echo "Configuration $cnf not found. Aborting..."
    exit 1
fi
source $(dirname $0)/$shared

echo "Setting up..."

lv="${lv}_base"
dev=/dev/$vg/$lv
mnt=$(mktemp -d)

if [ -e $dev ] ; then
    echo "Removing old LV..."
    oldmnt=$(mount | grep $lv | cut -f3 -d ' ')
    if [ x"$oldmnt" != "x" ] ; then
       umount $oldmnt
       rmdir $oldmnt
    fi
    remove_LV $dev
fi

echo "Creating and mounting LV..."
/sbin/lvcreate -L${xen_root_size} -n $lv $vg
create_FS $dev $fs_type
mount -t $fs_type $dev $mnt

if [ ! -d $src/pool ] ; then
    echo "Mounting $src..."
    mount $src
fi

echo "Bootstrapping ${distribution}/${flavour}..."
cdebootstrap -q -f $flavour $distribution $mnt $aptsrc

echo "Copying packages and scripts for installation..."
cp $debsrc/stage? $mnt/tmp -R
cp $bin/$config_script $mnt/tmp/
lists=/var/lib/apt/lists
cp $lists/*Packages ${mnt}${lists}
cp $lists/*Release  ${mnt}${lists}
kernel=$(uname -r | sed 's/xen0/xenU/')
mkdir $mnt/lib/modules
cp /lib/modules/*-xenUbhh $mnt/lib/modules/ -R
```

```
xen_config_base.sh
```

```
#!/bin/sh
#
echo '$Id: xen_config_base.sh 186 2006-06-08 20:40:52Z admin $'

pkglist="/tmp/packages.list"
fs_type=$1
domain=$2
net=$3

if [ ! -e /CONFIGURE_ME ] ; then
    echo "Do not run me on anything but a fresh XEN host. Aborti
    exit 1
else
    rm /CONFIGURE_ME
fi

function log()
{
    echo "Fixing ${1}..."
}

unset LANGUAGE LC_ALL LANG

log "/lib/tls"
mv /lib/tls /lib/tls.disabled

log "/etc/fstab"
cat > /etc/fstab << _END_
#
proc            /proc           proc   defaults                0
/dev/hda1       /               $fs_type       noatime,errors=re
/dev/hda2       none            swap    sw                     0
/dev/hda3      /srv            $fs_type        noatime,data=writ
_END_

log "shadow passwords"
/sbin/shadowconfig on

log "host name"
echo "BASE_SYSTEM_NOT_CONFIGURED" > /etc/hostname

log "network settings"
mkdir /etc/network

cat > /etc/network/interfaces << _END_
auto lo
iface lo inet loopback
_END_

cat > /etc/network/options << _END_
ip_forward=no
```

```
xen_new_host.sh
```

```
#!/bin/bash

echo '$Id: xen_new_host.sh 205 2006-06-17 14:19:10Z admin $'

force=0
go_from="lv"

cnf="/hoeg/share/xen.conf"
if [ -e $cnf ] ; then
    source $cnf
else
    echo "Configuration $cnf not found. Aborting..."
    exit 1
fi
source $(dirname $0)/$shared

# mandatory parameters
name=""
ip=""

# optional parms
data_size=""
make_data=0

# defaults
header='$Id: xen_new_host.sh 205 2006-06-17 14:19:10Z admin $'

function print_help() {
    echo "Usage: $0 [options]"
    cat << _END_
Required:
 -i IP address (example: -i 20  for ${xen_net}.20)
 -n name of VM (example: -n doris)
 -m memory in MB (default: -m $xen_mem)

Optional:
 -g start from section [lv|base|config|xen]
    lv     = create LVs
    base   = unpack base system
    config = configure base system
    xen    = create XEN configuration file
 -f force overwrite of existing LVs (DANGEROUS)
 -r size of root LV in GB (default: -r $xen_root_size)
 -s size of data LV in GB (example: -s 2)
 -t file system type (default: -t $fs_type)
 -v name of VG (default: -v $vg)

_END_
}

function check_LV() {
```

Re: Installing new Debian systems with debootstrap

*Posted by **Steve** (62.30.xx.xx) on Wed 9 Aug 2006 at 15:34*
*[**Send Message** | **View Steve's Scratchpad** | **View Weblogs**]*

Or you could just use **xen-tools**, a pre-packaged tool which will setup and configure an installation of Sarge/Etch/Sid/CentOS4/Gentoo easily.

This will also do the post-install configuration, etc, and also provide you with the simple facility to run post-install hooks of your own choosing. (Supplied hooks support installing X + VNC, or removing packages which aren't desired.)

Steve

Re: Installing new Debian systems with debootstrap

*Posted by **peterhoeg** (80.255.xx.xx) on Wed 9 Aug 2006 at 19:02*
*[**Send Message**]*

Sure, but where is the fun in that? ;-)

It would definately be an option, but xen-tools is currently only in testing/unstable and I only run stable and backports. Normally I would run testing but due to the fact that I am on this p*ss poor connection (serves me right for chosing a job in a 3rd world country), there is simply no way I can keep up with the torrent of new packages in testing.

Furthermore, these scripts tie in with my other normal sysadm scripts which I needed to write anyway.

But thanks for the hint - will take a look at xen-tools when etch goes gold.

Re: Installing new Debian systems with debootstrap

*Posted by **tong** (69.158.xx.xx) on Tue 8 Aug 2006 at 02:30*
*[**Send Message** | **View Weblogs**]*

About a mounth ago, I tried the debootstrap to install Debian Etch, but the kernel images in etch seemed to be broken by then:

http://article.gmane.org/gmane.linux.debian.user/248279

ie, both linux-image-2.6.15-1-k7 & linux-image-2.6.15-1-686 under etch failed to install, which

made the debootstrap Installation impossible.

Has anyone noticed that?

Re: Installing new Debian systems with debootstrap

*Posted by **Anonymous** (149.239.xx.xx) on Tue 8 Aug 2006 at 10:16*

Yes, i have had heavy problems with that. But my plan was to set up a root raid1 system on brand new sata drives. So at least there were the same problems you experienced with the kernel. I had to recreated initrd with approbiate modules, fix grub til boot works. After first successfull boot i noticed that the ttys were missing. It okay to fiddle around that, but if you do not have a standard installation you might run into problems. Maybe it is a bug deep in the debootstrap and debconf scripts - especaliy how the work together.

Ecaroh

Re: Installing new Debian systems with debootstrap

*Posted by **Anonymous** (137.121.xx.xx) on Tue 8 Aug 2006 at 11:52*

Hello,

Very interesting article... I usually use deboostrap to make jails for backporting packages ad things like that ...

There is one thing not clearly described in the article : how are boot process and kerneal installation handled by this method ?

(I can't remember deboostrap installing a kernel so I've missed something ?)

Re: Installing new Debian systems with debootstrap

*Posted by **Steve** (62.30.xx.xx) on Tue 8 Aug 2006 at 12:03*
*[**Send Message** | **View Steve's Scratchpad** | **View Weblogs**]*

Init, etc, don't really come into play here. As you suggest a kernel isn't installed.

You *can* install a new system via `debootstrap` (which I did a day or two agao) but you'll need to install a kernel and bootloader yourself after `debootstrap` has run.

`debootstrap` is more designed to install jails, or chroot systems, which can be used locally via the `chroot` command - and in that situation there is no "booting" as such.

Steve

Re: Installing new Debian systems with debootstrap

*Posted by **Anonymous** (83.198.xx.xx) on Wed 9 Aug 2006 at 19:20*

Ok, I agree.

Please, can you sketch the step "install a kernel and bootloader yourself after debootstrap has run". Precisely the part where you have to deal with mbr (re)writing...

Without this step, the title (install a NEW Debian system) is rather confusing.

Thanks for any hint.

Re: Installing new Debian systems with debootstrap

*Posted by **Steve** (62.30.xx.xx) on Wed 9 Aug 2006 at 19:25*
*[**Send Message** | **View Steve's Scratchpad** | **View Weblogs**]*

Sure here is a *brief* example. It assumes you've installed Sarge into a directory and chrooted into it.

Setup the sources list to the distribution you're installing, in this case sarge:

```
echo 'deb ftp://ftp.debian.org/debian stable contrib' >> /etc/apt/sou
```

Install a kernel:

```
apt-get update
apt-get install kernel-image-2.6.8-2-k7
```

(Choose an appropriate kernel for your system type. ie. 686 vs. k7)

Now install a bootloader:

```
apt-get install grub
grub-install /dev/hda
update-grub
```

That should be enough. Although it is worth checking that `/boot/grub/menu.lst` looks sane afterwards.

For a more detailed step you'd probably be as well to try it and ask for specific help if/when you have problems, or consult a mailing list.

Steve

Problem installing kernel

*Posted by **svenknoppix** (84.135.xx.xx) on Thu 14 Dec 2006 at 09:22*
*[**Send Message**]*

I have tryed, but failed installing the kernel in the chroot-environment, which is my new root-dir /dev/hda3 (included a mountet /proc directory)

debian:~# apt-get install kernel-image-2.6-k7
...
Setting up kernel-image-2.6.8-3-k7 (2.6.8-16sarge5) ...
/usr/sbin/mkinitrd: Cannot determine root device
Failed to create initrd image.
dpkg: error processing kernel-image-2.6.8-3-k7 (--configure):
subprocess post-installation script returned error exit status 9

Any help would be appreciated.
Thanks in advance - sven

Re: Problem installing kernel

*Posted by **Steve** (62.30.xx.xx) on Thu 14 Dec 2006 at 10:58*
*[**Send Message** | **View Steve's Scratchpad** | **View Weblogs**]*

Generally people don't install a kernel package in a chroot environment.

Still if you do you'll want to make sure that /proc and /sys are mounted in the chrrot environment. Something like this:

```
mount --bind /proc /path/to/chroot/proc
mount --bind /sys /path/to/chroot/sys
chroot /path/to/chroot
umount /path/to/chroot/sys
umount /path/to/chroot/proc
```

With these two mounts the kernel script will be able to detect your root drive, etc, and install.

Steve

Re: Problem installing kernel

*Posted by **svenknoppix** (84.135.xx.xx) on Thu 14 Dec 2006 at 19:51*
*[**Send Message**]*

...Generally people dont install a kernel package in a chroot environment.

Well I don t see another chance, because I don t have physical access to to the mashine. Only the running debian-sarge and a netboot-rescue console, where I also have to chroot.

What I like to do is to set up a new system and when Im satisfied with it to change over to the new root-partition

debian:~# mount --bind /proc /mnt/hda3/proc
debian:~# mount --bind /sys /mnt/hda3/sys
debian:~# chroot /mnt/hda3/
debian:/# umount /mnt/hda3/proc
umount: /mnt/hda3/proc: not found
debian:/# umount /mnt/hda3/sys
umount: /mnt/hda3/sys: not found

and the kernel is still not going to be installed...

mayby it is because /sys seems to be missing ...

debian:~# mount --bind /sys /mnt/hda3/sys
debian:~# mount --bind /proc /mnt/hda3/proc
debian:~# chroot /mnt/hda3/
debian:/# mount
proc on /proc type proc (rw)

but it seems to mounted in the same way ....

```
debian:/# exit
debian:~# mount
/dev/hda2 on / type ext3 (rw,errors=remount-ro)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
tmpfs on /dev/shm type tmpfs (rw)
usbfs on /proc/bus/usb type usbfs (rw)
/dev/hda3 on /mnt/hda3 type ext3 (rw)
/sys on /mnt/hda3/sys type none (rw,bind)
/proc on /mnt/hda3/proc type none (rw,bind)


sven
```

Re: Problem installing kernel

*Posted by **Steve** (62.30.xx.xx) on Thu 14 Dec 2006 at 20:07*
*[**Send Message** | **View Steve's Scratchpad** | **View Weblogs**]*

You appear to be confused about what is happening and what order to do things in.

1. Mount the partition.

2. Mount /dev + /sys inside it.

3. chroot.

4. Install the kernel.

5. Exit the chroot.

6. Unmount /dev + /sys.

Like so:

```
debian:~# mount /dev/hda3 /mnt/hda3
debian:~# mount --bind /proc /mnt/hda3/proc
debian:~# mount --bind /sys /mnt/hda3/sys
debian:~# chroot /mnt/hda3/
debian:~# ... install the kernel here ...
debian:~# ... exit the chroot ...
debian:~# umount /mnt/hda3/sys
debian:~# umount /mnt/hda3/proc
```

Steve

Re: Problem installing kernel

*Posted by **svenknoppix** (84.135.xx.xx) on Fri 15 Dec 2006 at 04:14*
*[**Send Message**]*

Thats the way I did it, but without succsess.
Finally I got it working

changing the /etc/mkinitrd/mkinitrd.conf linke this:

#ROOT=probe # not working in my chroot-environment
ROOT=/dev/hda3

sven

Re: Installing new Debian systems with debootstrap

*Posted by **El_Cubano** (66.93.xx.xx) on Sat 12 Aug 2006 at 13:44*

*[**Send Message**]*

In case anyone is running Sarge and wants to install a Sid chroot (I do this so that I can run lintian/linda/piuparts over packages since the latest versions of those tools will no longer install in Sarge), you must get the debootstrap (or cdebootstrap, if that is your preference) from Sid as the locations of some of the files have changed. The older versions in Sarge will not be able to find the necessary packages and will fail.

--
Roberto C. Sanchez
http://familiasanchez.net/~roberto

Re: Installing new Debian systems with debootstrap

*Posted by **Anonymous** (85.195.xx.xx) on Sun 27 Aug 2006 at 02:21*

When it comes to installing new installations of Debian GNU/Linux there is one,... page you should read:

"GENTOO 2006.0 Gnome, KDE and Xfce in less than 3 hours" at

http://linux.coconia.net/gentoo/2006.htm

This article describes how to easily install Gentoo from the LiveCD.

You can use the information in the article to similarly install from any LiveCD.

How to prevent /etc/init.d scripts from running on installation?

*Posted by **ymerlin** (213.61.xx.xx) on Thu 31 Aug 2006 at 08:14*
*[**Send Message**]*

Hi,

thanks for your article. I use this technique a lot.

One thing that anoys me is that, when chrooted, installing daemons that are automatically started from dpkg configure.

For example installing apache in chroot.

Do you know how to prevent /etc/init.d in chroot?

Thanks and regards,
Frank

Re: How to prevent /etc/init.d scripts from running on installation?

*Posted by **Steve** (62.30.xx.xx) on Thu 31 Aug 2006 at 09:37*
*[**Send Message** | **View Steve's Scratchpad** | **View Weblogs**]*

No, I'm afraid not.

I usually make sure that they are not running to avoid problems before I exit though:

```
apt-get install apache2
/etc/init.d/apache2 stop
```

Tahts the best advice I have I'm afraid.

Steve

Re: How to prevent /etc/init.d scripts from running on installation?

*Posted by **trakic** (130.226.xx.xx) on Wed 13 Sep 2006 at 10:35*
*[**Send Message** | **View Weblogs**]*

With `chmod -x /etc/init.d/foo? , apt-get dist-upgrade would also take care of such init.d script,...

Re: How to prevent /etc/init.d scripts from running on installation?

*Posted by **ymerlin** (213.61.xx.xx) on Fri 26 Jan 2007 at 08:54*
*[**Send Message**]*

Hi trakic,
thanks for your suggestions.

`chmod -x /etc/init.d/foo` does not work if foo is not already installed :-/

apt-get dist-upgrade does not work if I want to install foo

Something i found in the debootstrap package in file /usr/lib/debootstrap/scripts/sarge:

```
mv "$TARGET/sbin/start-stop-daemon" "$TARGET/sbin/start-stop-daemon.F
echo \
"#!/bin/sh
echo
echo \"Warning: Fake start-stop-daemon called, doing nothing\"" > "$T
chmod 755 "$TARGET/sbin/start-stop-daemon"

### ...

mv "$TARGET/sbin/start-stop-daemon.REAL" "$TARGET/sbin/start-stop-dae
```

I like that idea, but of course this only works if start-stop-daemon is used.

Bye,
Frank

When installing from a Live CD

*Posted by **Anonymous** (88.161.xx.xx) on Thu 8 Feb 2007 at 23:10*

When I install from a LiveCD, the bootloader installation fails. I just have to completely install the Live Distribution somewhere on the disk and then it's OK. I provide **a quick explanation here**

Re: How to prevent /etc/init.d scripts from running on installation?

*Posted by **Anonymous** (192.219.xx.xx) on Wed 4 Apr 2007 at 16:24*

Any followup on disabling services automatically started after being installed?

I am creating an automated installer and it's annoying having to keep track which packages install services, their service names, and then having to stop them.

Re: Installing new Debian systems with debootstrap

*Posted by **Anonymous** (82.34.xx.xx) on Tue 3 Jul 2007 at 01:23*

Good article, poor title.

I was hoping to find an article here that told me about installing a stand alone system, including the kernel, grub/lilo, setting up locales, etc. Basically doing what the install CDs do, but for people that can't use the CDs for some reason). Really I think this should be called 'Setting up a chroot environment' or something.

Thanks anyway for taking the time to write something, without people like you I'd be lost.

Re: Installing new Debian systems with debootstrap

*Posted by **mkb** (93.96.xx.xx) on Tue 1 Jul 2008 at 22:31*
*[**Send Message** | **View Weblogs**]*

Yes, I also am trying to work out how to use debootstrap to install a working Debian system on a Fedora box... ie the extra steps to include a kernel and to make the system boot into it....

Re: Installing new Debian systems with debootstrap

*Posted by **mkb** (93.96.xx.xx) on Tue 1 Jul 2008 at 22:43*
*[**Send Message** | **View Weblogs**]*

when I am logged into a Fedora box, by default it does not have (eg) /usr/sbin nor /sbin in the sudo/su path so one needs to add both these before running debootstrap otherwise you will get errors about not finding chroot, ldconfig and start-stop-daemon

Articles and comments are the property of their respective posters.

Trademarks are the property of their respective owners.
Debian is **a registered trademark** of **Software in the Public Interest, Inc**.

This site is copyright © 2004-2009 **Pink Pony Productions** / **Steve Kemp**.
Site hosting provided by **Bytemark Hosting**.

Email: **webmaster@debian-administration.org**

Article Feeds in Atom, RSS, & RDF formats