Setting up a Slackware chroot

There are multiple reasons why you might want to set up a Slackware chroot:

- building 32-bit packages on a 64-bit multilib system
- building -stable packages on a -current system
- building (and testing) packages for SBo on a clean system

For this guide, we'll create a chroot at **/chroot_folder** from **slackware-14.1-install-dvd.iso**.

Obtaining a Slackware installation of your desired architecture

Start with a Slackware installation DVD. Download it from from http://www.slackware.com/getslack/ via torrent.

Installing the packages

Automatically

You can create your chroot and install Slackware into it using the following elegant script:
http://tty1.uk/scripts/slackware/mkchroot

Manually

First, create a folder which will contain the chroot:

```
mkdir /chroot_folder
```

Mount the installation ISO:

```
mount -o loop slackware-14.1-install-dvd.iso /mnt/cdrom
cd /mnt/cdrom
```

After the ISO has been mounted, the packages (found under the *slackware* or *slackware64* folder) can be installed to the chroot folder with:

```
installpkg --root /chroot-folder */*.t?z
```

Then unmount the ISO:

```
umount /mnt/cdrom
```

Last
update:
2020/11/16
12:54
(UTC)
howtos:general_admin:setting_up_a_slackware_chroot https://docs.slackware.com/howtos:general_admin:setting_up_a_slackware_chroot

Setting up required files

fstab

The following minimal snippet can be used for *chroot_folder/etc/fstab*

```
# <file system> <mount point>   <type>  <options>           <dump>  <pass>
tmpfs           /dev/shm        tmpfs   defaults           0       0
devpts          /dev/pts        devpts  noexec,nosuid,gid=tty,mode=0620 0
0
sysfs           /sys            sysfs   defaults           0       0
proc            /proc           proc    defaults           0       0
```

Entering the Chroot

Automatic mount

To have the chroot mounted permanently, add the following to your <u>host system's</u> */etc/fstab*:

```
/dev              chroot_folder/dev              none bind,auto 0 0
/proc             chroot_folder/proc             none bind,auto 0 0
/sys              chroot_folder/sys              none bind,auto 0 0
/etc/resolv.conf chroot_folder/etc/resolv.conf none bind,auto 0 0
```

And reboot. Then run the following command to mount the chroot folder:

```
mount /chroot_folder
```

Manual mount

Refer to: ([volume_mounting](#)).

Entering the Chroot

Now that the chroot is set up, one can chroot into it:

```
chroot /chroot_folder /bin/bash
```

Note

Wrote a small script to automate the chroot steps.

---

https://raw.githubusercontent.com/aadityabagga/scripts/master/chroot.sh

Updating packages

After that, you can run slackpkg and update the packages: (configure_a_package_manager).

Exiting

When you're done, exited the chroot by pressing *Ctrl+d*.

Usecase: Building 32-bit Packages

Mounting partitions

I use a chroot to build 32-bit packages on a 64-bit system. I needed to mount the partition which contained the SlackBuilds for which I wanted to make a package, As my Slackware host mount point is (in this example) at /slackware_host_mount_point, I used the command:

```
mount -B /slackware_host_mount_point /chroot_folder
```

(http://www.thegeekstuff.com/2013/01/mount-umount-examples/)

Then I built the package by executing the Slackbuild as follows:

```
ARCH=i686 ./my_package.SlackBuild
```

An alternative way to build other ARCH packages

Instead of using `ARCH=i686' before running a slackbuild, we can use a program called `setarch', which, among other things, can set the ARCH and the kernel version in the shell. It has some handy symlinks, such as linux32 or i586, which saves some typing. The `linux32' command will set the reported kernel's ARCH as `i686'.

SYNOPSIS

```
setarch arch [options] [program [argument...]]
```

When we run `linux32' it starts a new shell, so we can also use it to chroot:

```
chroot /path/to/chroot linux32 /bin/zsh
```

or it could be run after chrooting into our new root. Settings will revert to normal when we exit the new shell. One useful option is to change the kernel's reported version:

```
chroot /path/to/chroot linux32 --uname-2.6 /bin/zsh
```

That `–uname-2.6' isn't a typo, there has to be a dash between the flag and the version. The setarch(8) man page shows all the options.

– Dave

See also

- https://wiki.archlinux.org/index.php/Chroot
- http://www.linuxquestions.org/questions/slackware-installation-40/install-slackware-to-a-folder-using-chroot-825598/#post4093831

Sources

* Originally written by Aaditya

howtos

From:
https://docs.slackware.com/ - **SlackDocs**

Permanent link:
https://docs.slackware.com/howtos:general_admin:setting_up_a_slackware_chroot

Last update: **2020/11/16 12:54 (UTC)**