

# Isolated Debian System with 'debootstrap' and 'chroot'

Hackers United Force  
<http://www.hackers-uf.org>  
[root@hackers-uf.org](mailto:root@hackers-uf.org)

It often happens that we are in front of requests of users that needs to have a login into a Linuxbox for launching some processes, but we don't trust enough these users to accept their demand, or we need to perform some tests for some configurations or applications but we doubt that they can cause a system damage. To answer these problems are coming to help us two tools , 'debootstrap' and 'chroot' .

- debootstrap

This tool allows to automate the process of downloading the packages of one of the versions of Debian GNU/Linux with the subsequent stages of decompression, installation and configuration, all directed toward a directory on any device for the mass storage of data (whether it's a hard disk, a pendrive or whatever). Obviously we can decide which version use, among Woody, Potato, Sarge, Etch, Lenny and Sid.

- chroot

Treated rather as an instrument of security, chroot allows you to 'set' a 'virtual' / directory for one (or more) determinated users. For example, we may decide to show to the user x the directory / as home/x, to prevent him to go back to the previous directories (which moreover are particularly important).

Let's see then the step to be made to reach our goal.

1. We need debootstrap and chroot, then we continue with their installation.

```
apt-get install debootstrap chroot
```

2. We have to decide which directory use for the new Debian system and eventually create it. In this example the system will be localized into /debstrap/.

```
mkdir -p /debstrap/
```

3. At this point we can proceed to the real debootstrap process. This operation may request a few minutes among the download and various operations, while slow connections (56K, ISDN, etc) may request even a modest number of hours.

Syntax ::

```
debootstrap PATH MIRROR VERSION
```

Example ::

```
debootstrap etch /debstrap/ http://ftp.it.debian.org/debian/
```

4. Now we have to ensure that the proc filesystem will be mounted automatically at the startup of our Linuxbox. To do this let's open the file /etc/fstab and add a new line following the example.

Syntax ::

```
CAUSAL_NAME PATH/proc proc none 0 0
```

Example ::

```
debstrapfs /debstrap/proc proc none 0 0
```

5. Let's mount it .

Syntax ::

```
mount PREVIOUS_NAME PATH/proc -t proc
```

Example ::

```
mount debstrapfs /debstrap/proc -t proc
```

6. Copy the network configuration from the real system to the debootstrap.

Syntax ::

```
cp /etc/hosts PATH/etc/hosts
```

Example ::

```
cp /etc/hosts /debstrap/etc/hosts
```

7. Now we can finally try out the just made work.

Syntax ::

```
chroot PATH FAVORITE_SHELL
```

Example ::

```
chroot /debstrap/ /bin/sh
```

In this way we are into a completely isolated system from the original one (to come back to the last one type 'exit'). If you have problems or doubts please contact the *Hackers United Force* at the address written in the top of this document.