

Welcome! [Log In](#) [Create A New Profile](#)

[Home](#) > [Debian](#) > [Topic](#)

[Advanced](#)

Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

Posted by [Vlad](#)

Vlad Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC October 04, 2011 08:20PM	Registered: 4 years ago Posts: 128
--	---------------------------------------

While browsing this forum I noticed that many people are looking for a working debian image from Jeff's script because for some reason theirs is not working any more. Very often they try to run Jeff's script again and again, which as of now is not working properly (at least the part where Debian Squeeze is installed). Of course, you can still use Shyd's script and so on but that's not the point. In fact, there's a better way to get a working rootfs without too much hassle. Note, that this howto is about creating rootfs only. It has nothing to do with the bootloader and thus assumes that your device already has a modified bootloader (U-Boot or UBIT). All you need is a x86 Debian/Ubuntu machine and a USB flash drive. Let's start.

First, we install debootstrap, create an img-file for the rootfs (1GB should be fine) and mount it

```
sudo -s
apt-get install debootstrap
dd if=/dev/zero of=debian_rootfs.img bs=1M count=1024
mkfs.ext3 -F debian_rootfs.img
mkdir /media/debian
mount debian_rootfs.img /media/debian -o loop
```

Now let's run the first stage of debootstrap and get Debian Squeeze armel port.

```
debootstrap --verbose --arch armel --variant=minbase --foreign squeeze /media/debian http://ftp.debian.org/debian
```

The output should look like [this](#). With the aid of [Binfmt_misc](#) we are able to chroot into armel rootfs on a x86 machine.

If you use Ubuntu do

```
apt-get install binfmt-support qemu-kvm-extras-static
```

Please note that the whole procedure doesn't work when using a Ubuntu Live CD. It does, however, work if performed under a normal Ubuntu installed to the HDD.

End Ubuntu section

If you use Debian you might need to use the latest version of qemu-user-static from the Debian testing repository. Especially if you get weird error messages like "Unsupported ioctl: cmd=0xc020660b" or "qemu: Unsupported syscall: 341"

To enable the testing repo, put this

```
Package: *
Pin: release a=stable
Pin-Priority: 900
```

```
Package: *
Pin: release o=Debian
Pin-Priority: -10
```

to your "/etc/apt/preferences" and this

```
APT::Default-Release "stable";
```

to your "/etc/apt/apt.conf". Then edit your "/etc/apt/apt.conf" and add this

```
deb http://ftp.debian.org/debian/ testing main contrib
```

Finally, install the packages

```
apt-get update
apt-get install -t testing qemu-user-static binfmt-support
```

End Debian section

Let's continue

```
modprobe binfmt_misc
cp /usr/bin/qemu-arm-static /media/debian/usr/bin
mkdir /media/debian/dev/pts
```

```
mount devpts /media/debian/dev/pts -t devpts
mount -t proc proc /media/debian/proc
chroot /media/debian/
```

If you see something like "I have no name!@hostname:/#", you're inside. Let's proceed with the second stage of debootstrap

```
/debootstrap/debootstrap --second-stage
```

The output should look like [this](#). At the end you should see

```
I: Base system installed successfully.
```

Now we can start installing necessary packages and configuring our rootfs. But before we must add the package sources.

```
cd /root
cat <<END > /etc/apt/sources.list
deb http://security.debian.org/ squeeze/updates main contrib non-free
deb-src http://security.debian.org/ squeeze/updates main contrib non-free
deb http://ftp.debian.org/debian/ squeeze main contrib non-free
deb-src http://ftp.debian.org/debian/ squeeze main contrib non-free
END
apt-get update
```

At this point it's a good idea to configure the language settings

```
exit
mount devpts /media/debian/dev/pts -t devpts
mount -t proc proc /media/debian/proc
chroot /media/debian/
export LANG=C
apt-get install apt-utils
apt-get install dialog
apt-get install locales
cat <<END > /etc/apt/apt.conf
APT::Install-Recommends "0";
APT::Install-Suggests "0";
END
dpkg-reconfigure locales
```

If you need English only, just select en_US.UTF-8 in both dialogs and run

```
export LANG=en_US.UTF-8
```

after that. For other languages, select en_US.UTF-8 and the appropriate locale for your language (xx_YY.UTF-8). For example, if you speak French, select en_US.UTF-8 and fr_FR.UTF-8 in the first dialog and choose fr_FR.UTF-8 in the second dialog. After that run

```
export LANG=fr_FR.UTF-8
```

This way Debian will speak your mother tongue. The output should look like [this](#). It's on time to install some important packages.

```
apt-get install linux-image-2.6-kirkwood flash-kernel module-init-tools dhcp3-client udev netbase ifupdown iproute
```

The output should look like [this](#). Now we are going to reproduce some basic configuration steps which are also done in Jeff's script

Configure eth0 and dhcp

```
cat <<END > /etc/network/interfaces
auto lo eth0
iface lo inet loopback
iface eth0 inet dhcp
END
```

Setup machine's name (in this case it will be "debian")

```
echo debian > /etc/hostname
```

Some onboard flash specific settings

```
cat <<END > /etc/fw_env.config
# MTD device name      Device offset   Env. size      Flash sector size      Number of sectors
/dev/mtd0 0xc0000 0x20000 0x20000
END
```

File system mounts

```
cat <<END > /etc/fstab
# /etc/fstab: static file system information.
#
# <file system> <mount point> <type> <options> <dump> <pass>
/dev/root / ext3 noatime,errors=remount-ro 0 1
tmpfs /tmp tmpfs defaults 0 0
END
```

Activate remote console and disable local consoles

```
echo 'T0:2345:respawn:/sbin/getty -L ttyS0 115200 linux' >> /etc/inittab
sed -i 's/^\([1-6]:.* tty[1-6]\)/#\1/' /etc/inittab
```

Tell the system that we have no hardware clock

```
echo HWCLOCKACCESS=no >> /etc/default/rcS
```

Slightly speed up the boot process

```
echo CONCURRENCY=shell >> /etc/default/rcS
```

Prolong the life of your USB flash drive

```
rm -f /etc/blkid.tab
ln -s /dev/null /etc/blkid.tab
rm -f /etc/mtab
ln -s /proc/mounts /etc/mtab
```

[Optional, not in Jeff's script] This are some packages I consider to be rather useful. However, it's not necessary to install them

```
apt-get install cron anacron bash-completion hdparm rsync bzip2 p7zip unrar unzip zip p7zip-full screen less usbutils
```

Since GoFlex/Dockstar have no hardware clock, it might be useful to sync local time with time servers. This one is also optional.

```
dpkg-reconfigure tzdata
-> Choose your continent and country
apt-get remove ntpdate
apt-get install ntp
nano /etc/ntp.conf
```

replace

```
server 0.debian.pool.ntp.org iburst
server 1.debian.pool.ntp.org iburst
server 2.debian.pool.ntp.org iburst
server 3.debian.pool.ntp.org iburst
```

by

```
server time-a.nist.gov
server ptbtime1.ptb.de
server ntp.ubuntu.com
```

and add this to the end of the file

```
restrict time-a.nist.gov
restrict ptbtime1.ptb.de
restrict ntp.ubuntu.com
restrict default notrust nomodify nopeer
```

save and exit nano.

[End Optional section]

Dockstar section

So far we've already installed Debian mainline kernel. Dockstar is officially supported so you just need to create uImage and uInitrd. When a new kernel is installed the following new files (maybe with slightly different numbers) usually appear in "/boot"

```
vmlinuz-2.6.32-5-kirkwood
initrd.img-2.6.32-5-kirkwood
System.map-2.6.32-5-kirkwood
```

Here vmlinuz-2.6.32-5-kirkwood is a compressed image of the kernel, while initrd.img-2.6.32-5-kirkwood is a compressed image of the initramfs, a small filesystem that is extracted to the RAM during the boot process and which can perform some useful tasks before the kernel switches to the "real" rootfs on your drive.

U-Boot and UBIT are not able to use vmlinuz and initrd directly. Instead, you must create special images out of them, which are the familiar uImage and uInitrd. This is done by the mkimage utility

```
mkimage -A arm -O linux -T kernel -C none -a 0x00008000 -e 0x00008000 -n Linux-2.6.32-5-kirkwood -d /boot/vmlinuz
mkimage -A arm -O linux -T ramdisk -C gzip -a 0x00000000 -e 0x00000000 -n initramfs-2.6.32-5-kirkwood -d /boot/initrd
cd /boot
ln -s ../uImage uImage
ln -s ../uInitrd uInitrd
```

I prefer to put both images to the root of the filesystem and then symlink them to /boot. That's because u-Boot usually looks for uInitrd in

/boot while UBIT thinks that it should be in the root of the rootfs. This way I get both bootloaders covered.

End Dockstar section

GoFlex section

If you have a GoFlex Net/Home, things get a little bit more complicated. First of all, GoFlex is not supported by the Debian kernel. It's possible to use it if you set the arcNumber to 2097 or 2998 so that the kernel thinks you have a Pogo or Dockstar. In this case, however, the SATA ports will not work. Thus there are two conditions which must be met in order to get the most of your GoFlex

- The arcNumber must be set to 3089 for the GoFlex Net or 3338 for the GoFlex Home
- You must use a kernel that supports GoFlex Net/Home.

As I've already said, Debian's kernel doesn't support GoFlex by default, so you'll have to get a working kernel elsewhere. Here's what you can do

- Use one of the kernels compiled by people from this forum. For instance, there is a 2.6.37 kernel from [varkey](#) or a 2.6.37 kernel from [Jeff](#)
- Compile the kernel yourself. There are kernel patches available for 2.6.37-3.0.0 kernel versions. Check out my howtos on compiling a [vanilla 2.6.38.8](#) kernel and patching a [Debian 3.0.0 kernel](#) to work with GoFlex Net/Home

You can install Jeff's kernel while in the chroot environment

```
cd /root
wget http://jeff.doozan.com/debian/goflex/v0.6/v2.6.37/boot.tar.gz
wget http://jeff.doozan.com/debian/goflex/v0.6/uInitrd
tar xzvf boot.tar.gz
cp boot/uImage /
cp uInitrd /
cd /boot
ln -s ../uImage uImage
ln -s ../uInitrd uInitrd
cd /root
wget http://jeff.doozan.com/debian/goflex/v0.6/v2.6.37/modules.tar.gz
tar -C / -xzvf modules.tar.gz
rm -r boot.tar.gz modules.tar.gz 2.6.37-ubit-01715-g1e5cda0 boot uImage
```

HINT: If you have UBIT and always boot into the initramfs instead of the rootfs, simply delete uInitrd and perform the initrd hack described below.

Supposing that you don't want to use Jeff's kernel but rather try something new, you will probably end up downloading a deb file containing the kernel. To install it use

```
ls /boot
dpkg -i install name-of-the-package.deb
ls /boot
```

As usual you will see new vmlinuz-xxx and initrd.img-xxx corresponding to the installed kernel in /boot . Use Something like

```
mkimage -A arm -O linux -T kernel -C none -a 0x00000000 -e 0x00000000 -n Linux-xxx -d /boot/vmlinuz-xxx /uImage
mkimage -A arm -O linux -T ramdisk -C gzip -a 0x00000000 -e 0x00000000 -n initramfs-xxx -d /boot/initrd.img-xxx /uInitrd
cd /boot
ln -s ../uImage uImage
ln -s ../uInitrd uInitrd
```

to prepare uImage and uInitrd and allow your bootloader use the new kernel.

Another problem concerning GoFlex is that the bootloader usually fails to find the flash drive containing Debian image if there are HDDs connected to the SATA ports. The solution depends on whether you have u-Boot v0.5 or UBIT v0.6

Independent of the solution you are going to use, it's mandatory to label your rootfs partition as "rootfs". To do this, connect your USB flash drive to a Linux box and issue

```
sudo tune2fs -L "rootfs" /dev/whatever
```

where /dev/whatever is the path to the partition containing the rootfs for the GoFlex

If you have u-Boot run

```
fw_setenv usb_init "run usb_scan; setenv usb_root LABEL=rootfs"
```

then reboot and you're done. For more information on this solution check out [this thread](#)

If you have UBIT v0.6 you must hardcode the label of the rootfs into your initrd, because as of now it's the only known way to get the system booting properly. For this you will need to install a suitable kernel deb, as Jeff's kernel included in the rootfs you've created using this guide contains only uInitrd but not the original initrd.img. Let's suppose you've installed a kernel package and the initrd corresponding to the kernel is called initrd.img-2.6.38.8-goflex-ubit. Then you should do the following.

```
cd /boot
mkdir initrd.img-new
cp initrd.img-2.6.38.8-goflex-ubit initrd.img-new/
cd initrd.img-new/
gzip -dc initrd.img-2.6.38.8-goflex-ubit | cpio -id
rm initrd.img-2.6.38.8-goflex-ubit
nano init
```

Find the section which looks like

```
if [ -n "${noresume}" ]; then
export noresume
unset resume
else
resume=${RESUME:-}
fi
```

and replace it by

```
if [ -n "${noresume}" ]; then
export noresume
unset resume
else
resume=${RESUME:-}
fi
ROOT="/dev/disk/by-label/rootfs"
```

Save your changes (Ctrl+W) and exit (Ctrl+X) nano. Now run

```
find ./ | cpio -H newc -o > initrd.img-2.6.38.8-goflex-ubit
gzip -9 initrd.img-2.6.38.8-goflex-ubit
mv initrd.img-2.6.38.8-goflex-ubit.gz ../initrd.img-2.6.38.8-goflex-ubit_mod
cd ..
rm -r initrd.img-new
mkimage -A arm -O linux -T ramdisk -C gzip -a 0 -e 0 -n Linux-2.6.38.8 -d /boot/initrd.img-2.6.38.8-goflex-ubit_mod
cd /boot
ln -s ../uInitrd uInitrd
```

With the modified initrd your GoFlex should boot fine regardless of the number of SATA HDDs or USB drives connected to it.

Finally, set root password

```
passwd
```

Now we're done here

```
exit
```

The size of the rootfs should be about about 300-400MB

```
umount /media/debian/proc
umount /media/debian/dev/pts
du -sh /media/debian/
```

So we have a working rootfs image which we want to put on a USB flash drive. Connect your USB flash drive (Here I assume it to be /dev/sdb) and use gparted to create a primary ext3 partition (sdb1) and a primary swap partition (sdb2).

It's very important that the ext3 partition is the very first partition of your flash drive. Mount the newly created partition and copy rootfs onto it.

```
mount /dev/sdb1 /mnt
cp -a /media/debian/* /mnt
```

Finally, unmount the partition

```
umount /mnt
```

plug the flash drive into your GoFlex/Dockstar and power it up. Please make sure that no other USB flash drives or HDD are connected to the system, otherwise it won't boot. You should be able to ssh into it as root using the password you've provided via passwd.

If you have the u-Boot v0.5 bootloader (that's what you get when unlocking the device using Jeff's script) and you still didn't enable the netconsole, you really should do this now. Follow Jeff's instructions in [this thread](#). On UBIT v0.6 netconsole is enabled by default.

Sources:

<http://www.digriz.org.uk/debootstrap-qemu>

<http://forum.xda-developers.com/showthread.php?t=1172504>

Edited 16 time(s). Last edit at 11/06/2011 12:04PM by Vlad.

varkey

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

October 05, 2011 09:16AM

Registered: 5 years ago

Posts: 136

Thank you for the detailed guide. This will be really helpful! :)

orange

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

October 09, 2011 01:53AM

Registered: 4 years ago

Posts: 4

Hi,

qemu-kvm-extras-static is not available anymore and I cannot get the above to work. Any help is appreciated.

Vlad

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

October 09, 2011 07:45AM

Registered: 4 years ago

Posts: 128

orange Wrote:

> Hi,

>

> qemu-kvm-extras-static is not available anymore

That's hardly possible, because qemu-kvm-extras-static is definitely in the Ubuntu repos

<http://packages.ubuntu.com/lucid/qemu-kvm-extras-static>

If you use Ubuntu, do you have universe repos enabled?

In Debian it's called qemu-user-static

<http://packages.debian.org/squeeze/qemu-user-static>

Edited 1 time(s). Last edit at 10/09/2011 08:09AM by Vlad.

Sebastian Vettel

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

October 09, 2011 07:51AM

apt-get install qemu-user-static

Vlad

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

October 09, 2011 08:06AM

Registered: 4 years ago

Posts: 128

Sebastian Vettel Wrote:

> apt-get install qemu-user-static

Right. Ubuntu kind of split Debian's qemu-user-static into qemu-user-static and qemu-kvm-extras-static. In Debian it's still qemu-user-static. Thanks for the hint.

whoelse

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

October 10, 2011 06:03PM

Registered: 5 years ago

Posts: 4

Hi Vlad

I followed the tutorial but couldn't login or ping the dockstar.

The solution was to install a DHCP client.

apt-get install dhcpd

Hope this helps anyone who is stuck.

Vlad

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

October 10, 2011 06:43PM

Registered: 4 years ago

Posts: 128

Thanks for your feedback, whoelse. I updated the howto to include dhcp3-client (I don't like dhcpd because it tends to add a long UUID

to the hostname which confuses my static DHCP on my router)

[davygravy](#)

Registered: 4 years ago

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

Posts: 490

October 11, 2011 05:23PM

hehehe, this is sweet... when I started reading it (just the general idea) my first though, "hmmm, no way to do that w/o an emulator...", lo and behold, qemu.

Thanks for writing this up. I may try it w/ ppc for some of my old PPC Linkstations. I rolled Lenny rootfs's for them when Lenny was "shiny new", but they are very long in the tooth now.

Thanks for sharing this idea here. I'd previously just done the first part in Ubuntu, the 2nd stage in qemu is a fun/nice twist. Nifty.

[hanker](#)

Registered: 5 years ago

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

Posts: 58

October 12, 2011 09:23AM

Greetings

I am running through this how-to on a Dell Mini 9 netbook (Intel Atom processor) running Debian squeeze. Everything's up-to-date.

```
$ uname -a
Linux debian-dm9 2.6.32-5-686 #1 SMP Mon Oct 3 04:15:24 UTC 2011 i686 GNU/Linux
```

Following the procedure up through here:

```
apt-get update
```

everything looks fine.

But then the next command generates a lot of things that look not so good:

```
I have no name!@debian-dm9:~# apt-get install apt-utils
Reading package lists... Done
Building dependency tree... Done
The following NEW packages will be installed:
  apt-utils
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 267 kB of archives.
After this operation, 606 kB of additional disk space will be used.
Get:1 http://ftp.ch.debian.org/debian/ squeeze/main apt-utils armel 0.8.10.3+squeeze1 [267 kB]
Fetched 267 kB in 7s (37.4 kB/s)
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
    LANGUAGE = (unset),
    LC_ALL = (unset),
    LANG = "en_US.UTF-8"
are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").
/usr/bin/locale: Cannot set LC_CTYPE to default locale: No such file or directory
/usr/bin/locale: Cannot set LC_MESSAGES to default locale: No such file or directory
/usr/bin/locale: Cannot set LC_ALL to default locale: No such file or directory
debconf: delaying package configuration, since apt-utils is not installed
Can not write log, openpty() failed (/dev/pts not mounted?)
Selecting previously deselected package apt-utils.
(Reading database ... Unsupported ioctl: cmd=0xc020660b
Unsupported ioctl: cmd=0xc020660b
Unsupported ioctl: cmd=0xc020660b
...skip lots of these...

Unsupported ioctl: cmd=0xc020660b
Unsupported ioctl: cmd=0xc020660b
(Reading database ... 6137 files and directories currently installed.)
Unpacking apt-utils (from ../apt-utils_0.8.10.3+squeeze1_armel.deb) ...
qemu: Unsupported syscall: 341
qemu: Unsupported syscall: 341
qemu: Unsupported syscall: 341
...skip lots of these...

qemu: Unsupported syscall: 341
qemu: Unsupported syscall: 341
Can not write log, openpty() failed (/dev/pts not mounted?)
Setting up apt-utils (0.8.10.3+squeeze1) ...
I have no name!@debian-dm9:~#
```

Is this working right? Is this normal?

Thank you! :)

Edited 2 time(s). Last edit at 10/12/2011 10:18AM by hanker.

Vlad

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

October 12, 2011 03:48PM

Registered: 4 years ago

Posts: 128

It looks good except for the "qemu: Unsupported syscall: 341" and "Unsupported ioctl: cmd=0xc020660b" messages. Those actually shouldn't appear. They look like a qemu bug, but if the bug is not relevant for the security, it might still be present in the Debian Squeeze

repository. You might try using qemu-user-static from the Debian testing repository or just take a Ubuntu 11.04 live CD.

hanker

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

October 13, 2011 02:57PM

Registered: 5 years ago

Posts: 58

Vlad Wrote:

> or just take a Ubuntu 11.04
> live CD.

Thank you for your reply!

I made an Ubuntu 11.04 liveCD. I booted the Dell Mini 9 from it. I have a 4 GB fat32-formatted usb flash drive mounted at /media /UNTITLED, and I'm building the image on that.

Ahhh - never mind for now. I think I just need to uncomment the lines in /etc/apt/sources.list dealing with the universe repository. Let me try that later this afternoon. Thank you! :)

Everything works fine up to and including this:

```
debootstrap --verbose --arch armel --variant=minbase --foreign squeeze /media/debian http://ftp.debian.org/debian
```

But then the next step gives an error:

> If you use Ubuntu do

```
apt-get install binfmt-support qemu-kvm-extras-static
```

I try that and I get the following error:

```
E: Unable to locate package qemu-kvm-extras-static
```

Ideas?

Any help is appreciated :)

Edited 1 time(s). Last edit at 10/13/2011 04:00PM by hanker.

Vlad

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

October 13, 2011 04:09PM

Registered: 4 years ago

Posts: 128

I guess that's because qemu-kvm-extras-static is in the universe repository which is not enabled by default in Ubuntu LiveCD. Go to Synaptic and enable universe repo. Then do the usual "sudo apt-get update". After that the package should appear.

hanker

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

October 14, 2011 06:54PM

Registered: 5 years ago

Posts: 58

I've got a USB hard disk with Ubuntu 11.04, universe repository enabled, everything up to date.

Everything worked correctly up until here:

```
chroot /media/debian/
```

That finished without error:

```
root@dm9-Inspiron-910:~# chroot /media/debian/  
I have no name!@dm9-Inspiron-910:/#
```

Next:

```
/debootstrap/debootstrap --second-stage
```

At the end of the output produced by that, I got this:

```
I: Configuring login...
I: Configuring libuuid1...
I: Configuring sysvinit...
I: Configuring libblkid1...
I: Configuring mount...
I: Configuring util-linux...
I: Configuring e2fsprogs...
W: Failure while configuring required packages.
I have no name!@dm9-Inspiron-910:/#
```

Hmmm...

I ran that same command again, and I got this:

```
I: Configuring debian-archive-keyring...
I: Configuring apt...
I: Base system installed successfully.
```

Also basically all install commands from here on out generate messages that look like this:

```
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
    LANGUAGE = "en_US:en",
    LC_ALL = (unset),
    LANG = "en_US.UTF-8"
are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").
/usr/bin/locale: Cannot set LC_CTYPE to default locale: No such file or directory
/usr/bin/locale: Cannot set LC_MESSAGES to default locale: No such file or directory
/usr/bin/locale: Cannot set LC_ALL to default locale: No such file or directory
Can not write log, openpty() failed (/dev/pts not mounted?)
```

Not sure if this is a problem...

Next:

```
apt-get install flash-kernel module-init-tools dhcp3-client udev netbase ifupdown iproute openssh-server iputils-ping
```

At end of output:

```
Errors were encountered while processing:
  isc-dhcp-client
  openssh-client
  openssh-server
  dhcp3-client
E: Sub-process /usr/bin/dpkg returned an error code (1)
I have no name!@dm9-Inspiron-910:/root#
```

Also:

```
apt-get install linux-image-2.6-kirkwood
```

Produced:

```
perl: warning: Falling back to the standard locale ("C").
Segmentation fault
Can not write log, openpty() failed (/dev/pts not mounted?)

Errors were encountered while processing:
```

```
linux-base
linux-image-2.6.32-5-kirkwood
linux-image-2.6-kirkwood
E: Sub-process /usr/bin/dpkg returned an error code (1)
I have no name!@dm9-Inspiron-910:/root#
```

I soldiered on through the rest, but I ended up with a usb stick that wouldn't boot.

The impression I get is that this whole qemu-debootstrap thing is maybe kind of weird and unreliable. Has this been your experience? What am I doing wrong? I'm just copying and pasting the commands, and once I've done the chroot, most of the commands produce weird messages or just fail.

Any help is appreciated! :)

Edited 1 time(s). Last edit at 10/15/2011 10:20AM by hanker.

<u>Vlad</u>	Registered: 4 years ago
Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC	Posts: 128
October 15, 2011 10:23AM	

I've updated some parts of the howto, such that there should be no more errors related to LANGUAGE or /dev/pts. I've also just tested it on my Ubuntu 11.04 and it worked flawlessly. Unfortunately, I must confirm that for some reason this doesn't work with Ubuntu Live CD. You need a usual system installed on the HDD. Or just use the qemu from the testing repo on your Debian Netbook. No idea how did you get those segfaults. I didn't have any of them.

Concerning the reliability of debootstrap. Well, actually the second phase is supposed to be performed on the device you're installing Debian to. If you're doing this via an emulator you can't expect everything working as good as on a native system. In my case, however, the whole debootstrapping works very well without any errors. I hardly believe that you can get anything more reliable, unless you do the whole thing on your Dockstar/GoFlex.

<u>hanker</u>	Registered: 5 years ago
Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC	Posts: 58
October 16, 2011 10:45AM	

I got it to work. It boots the Dockstar just fine. Thank you. This is very useful! :)

A few notes:

I'm using a Dell Mini 9 netbook (Intel Atom processor) for a dev/testing machine. It boots from an external usb hard disk with Ubuntu 11.04 and everything up to date.

One unexpected or "brittle" thing that happens during the procedure is that sometimes (but not every time) this step:

```
/debootstrap/debootstrap --second-stage
```

doesn't work. Sometimes it will say

```
W: Failure while installing base packages. This will be re-attempted up to five times.
```

or

```
W: Failure while configuring required packages.
```

I have no idea why it does this, and it doesn't do it every time. One time this happened and I just ran the command again and finally got

```
I: Base system installed successfully.
```

But then the "locales" step failed like so:

```
# apt-get install locales
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
```

```
locales
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 4757 kB of archives.
After this operation, 13.1 MB of additional disk space will be used.
Get:1 http://ftp.ch.debian.org/debian/ squeeze/main locales all 2.11.2-10 [4757 kB]
Fetched 4757 kB in 4s (1002 kB/s)
Preconfiguring packages ...
Selecting previously deselected package locales.
(Reading database ... 6347 files and directories currently installed.)
Unpacking locales (from .../locales_2.11.2-10_all.deb) ...
Setting up locales (2.11.2-10) ...
dpkg: error processing locales (--configure):
 subprocess installed post-installation script killed by signal (Segmentation fault)
configured to not write apport reports

Errors were encountered while processing:
 locales
E: Sub-process /usr/bin/dpkg returned an error code (1)
root@dm9-Inspiron-910:/#
```

So I started over from the beginning. After two tries, the

```
/debootstrap/debootstrap --second-stage
```

step finally ran through without any warnings. After that the rest of the procedure also finished without error.

> Now, if you have a Dockstar you may install the
> usual Debian stable kernel and create uImage and
> uInitrd via mkimage.

Could you go into a little more detail here and show the actual commands that I would run on my Dockstar? (I'm still kind of a n00b with this stuff.)

Thanks again! :)

Vlad Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC October 22, 2011 07:40AM	Registered: 4 years ago Posts: 128
--	---------------------------------------

hanker Wrote:

```
-----
> > Now, if you have a Dockstar you may install the
> > usual Debian stable kernel and create uImage
> and
> > uInitrd via mkimage.
>
> Could you go into a little more detail here and
> show the actual commands that I would run on my
> Dockstar? (I'm still kind of a n00b with this
> stuff.)
>
> Thanks again! :)
```

I added some additional infos on installing a proper kernel. Hope that helps.

hanker Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC October 22, 2011 10:44AM	Registered: 5 years ago Posts: 58
--	--------------------------------------

Vlad Wrote:

```
-----
> hanker Wrote:

> I added some additional infos on installing a
> proper kernel. Hope that helps.
```

Yes, that's very useful. Thank you! :)

mbohupa Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC October 30, 2011 07:41PM	Registered: 5 years ago Posts: 2
---	-------------------------------------

Thanks for the detailed guide! It works great!!

bell	Registered: 5 years ago
-------------	-------------------------

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

Posts: 5

October 31, 2011 12:14AM

Vlad, thanks a lot for the great tutorial.

I have an x86 Debian, but there are no "/etc/apt/preferences" and "/etc/apt/apt.conf". Instead, I have two directories: "/etc/apt/preferences.d" and "/etc/apt/apt.conf.d". So I put one file in "/etc/apt/preferences.d" to define

```
APT::Default-Release "stable";
```

and one file in "/etc/apt/apt.conf.d" to define

```
deb http://ftp.debian.org/debian/ testing main contrib
```

Mine is a Goflex Net with Jeff's uBoot (but accidentally had "setenv arcNumber 2998"):

U-Boot 2010.09 (Oct 23 2010 - 11:53:10)

Marvell-GoflexNet by Jeff Doozan, Peter Carmichael

So I thought I can keep Jeff's kernel from this step:

```
cd /root
wget http://jeff.doozan.com/debian/goflex/v0.6/v2.6.37/boot.tar.gz
tar xzvf boot.tar.gz
cp boot/uImage /
cd /boot
ln -s ../uImage uImage
cd /root
wget http://jeff.doozan.com/debian/goflex/v0.6/v2.6.37/modules.tar.gz
tar -C / -xzvf modules.tar.gz
rm -r boot.tar.gz modules.tar.gz 2.6.37-ubit-01715-g1e5cda0 boot uImage
```

The message from the netconsole shows that my Goflex Net boots OK and start to load the kernel. But I cannot SSH to it. After I installed the "linux-image-2.6-kirkwood" kernel in the debootstrap environment (Although Vlad said "it's not really a good idea to install the kernel in a debootstrap environment"), I can SSH in.

I had followed Vlad's steps strictly but I'm not sure what I did wrong. Anybody successfully SSH in without the official "linux-image-2.6-kirkwood" kernel? In other words, SSH in with only the following files from Jeff?

<http://jeff.doozan.com/debian/goflex/v0.6/v2.6.37/boot.tar.gz>

<http://jeff.doozan.com/debian/goflex/v0.6/v2.6.37/modules.tar.gz>

<http://jeff.doozan.com/debian/goflex/v0.6/uInitrd>

I had thought maybe "setenv arcNumber 2998" messed up, but it didn't help even after I changed it to other arcNumbers.

Any thought? Thanks in advance!

Vlad

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

Registered: 4 years ago

November 02, 2011 05:14PM

Posts: 128

Hi bell,

"/etc/apt/preferences" and "/etc/apt/apt.conf" are usually not there by default and must be created, but if it works the way you did it, this doesn't really matter.

I'm not sure what the problem is, but did you try putting Jeff's uInitrd and uImage to /boot? That actually should work.

You also write that you can see how U-Boot loads the kernel, from which I assume that you're using netconsole. Do you see anything when listening to the port that lists kernel messages?

I usually open two terminal windows and run

```
sudo ifconfig eth0:0 10.10.10.5
nc -lu 10.10.10.5 6666
```

in the first one and

```
sudo ifconfig eth0:1 10.10.10.4
nc -lu 10.10.10.4 6666
```

in the second one. This way one can monitor U-Boot and kernel log simultaneously.

bell

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

Registered: 5 years ago

November 02, 2011 06:27PM

Posts: 5

Vlad,

Thanks for the suggestions.

Jeff's uImage and uInitrd did not work for me -- either the symbolic links or putting them in /boot. Does Jeff's kernel (uImage and uInitrd) have netconsole enabled by default? My home network is on 192.168.1.*, if the built-in IP addresses in the kernel are 10.10.10.4 or 10.10.10.5, will your command work? (I'll give try it out once I get home.)

When I tried to unlock the GoFlex Net, I used Jeff's script, which is broken. But I believe uBoot has been updated. Then I tried Shyd's script, which completed successfully (I can SSH in after installation). But I wanted to try your method. I believe currently I have Jeff's uBoot:

U-Boot 2010.09 (Oct 23 2010 - 11:53:10)
Marvell-GoFlexNet by Jeff Doozan, Peter Carmichael

For files such as <http://jeff.doozan.com/debian/goflex/v0.6/v2.6.37/boot.tar.gz>, are they intended for Jeff's uBoot or UBIT 0.6? Sorry for my ignorance, I don't know the difference between the uBoot and uBit.

Vlad	Registered: 4 years ago
Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC	Posts: 128
November 02, 2011 08:12PM	

Oh, I completely forgot that netconsole seems to work a bit differently on U-Boot. Then I guess the commands I provided will not work. Just check out Jeff's thread for the correct commands:

<http://forum.doozan.com/read.php?3,14,14>

As far as Jeff's 2.6.37 kernel is concerned, I guess it should work independently of the bootloader. I remember using it with Jeff's u-Boot at the very beginning but at the same time I had my arcNumber set to 3089 (GoFlex Net). Still, I believe that that kernel should also work with your current arcNumber.

Anyway, since you seem to have enough Linux experience, you could try to compile a kernel yourself. I wrote a howto on patching a 3.0.0 kernel from the Debian testing repo to work with GoFlex Net

<http://forum.doozan.com/read.php?2,6010>

It's pretty instructive so you might give it a try, if Jeff's kernel doesn't work for you.

hanker	Registered: 5 years ago
Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC	Posts: 58
November 03, 2011 09:14AM	

Vlad:

I have gotten this to work with an earlier version of your procedure: Dockstar rootfs, Debian squeeze with proper kernel, 2.6.32-5-kirkwood, everything up-to-date, etc. Boots fine, etc. I have that version Clonezilla-cloned to a hard disk for safekeeping.

I wanted to go through the procedure again and make a new rootfs image, and I can't get it to work again. The flash drive I produced won't boot. Netconsole shows an error where it can't find uInitrd. And looking at this block of code in the procedure:

```
cd /root
wget http://jeff.doozan.com/debian/goflex/v0.6/v2.6.37/boot.tar.gz
tar xzvf boot.tar.gz
cp boot/uImage /
cd /boot
ln -s ../uImage uImage
cd /root
wget http://jeff.doozan.com/debian/goflex/v0.6/v2.6.37/modules.tar.gz
tar -C / -xzvf modules.tar.gz
rm -r boot.tar.gz modules.tar.gz 2.6.37-ubit-01715-g1e5cda0 boot uImage
```

this installs uImage but not uInitrd. So the flash drive ends up with no uInitrd and won't boot.

I modified that block of code like so:

```
cd /root
wget http://jeff.doozan.com/debian/goflex/v0.6/v2.6.37/boot.tar.gz
tar xzvf boot.tar.gz
cp boot/uImage /
cd /boot
ln -s ../uImage uImage

cd /root
wget http://jeff.doozan.com/debian/goflex/v0.6/uInitrd
cp uInitrd /
cd /boot
ln -s ../uInitrd uInitrd

cd /root
wget http://jeff.doozan.com/debian/goflex/v0.6/v2.6.37/modules.tar.gz
tar -C / -xzvf modules.tar.gz
```

```
rm -r boot.tar.gz modules.tar.gz 2.6.37-ubit-01715-g1e5cda0 boot uImage uInitrd
```

to install a uInitrd. This makes an image that appears to boot - I see this:

```
Loading file "/boot/uImage" from usb device 0:1 (usbda1)
1 bytes read
Found bootable drive on usb 0:1
Loading file "/boot/uImage" from usb device 0:1 (usbda1)
2941544 bytes read
Loading file "/boot/uInitrd" from usb device 0:1 (usbda1)
4005544 bytes read
## Booting kernel from Legacy Image at 00800000 ...
Image Name:   Linux-2.6.37-ubit-01715-g1e5cda0
Image Type:   ARM Linux Kernel Image (uncompressed)
Data Size:    2941480 Bytes = 2.8 MiB
Load Address: 00008000
Entry Point:  00008000
Verifying Checksum ... OK
## Loading init Ramdisk from Legacy Image at 01100000 ...
Image Name:   UBIT-v0.6-00125-ga21bcca
Image Type:   ARM Linux RAMDisk Image (gzip compressed)
Data Size:    4005480 Bytes = 3.8 MiB
Load Address: 00000000
Entry Point:  00000000
Verifying Checksum ... OK
Loading Kernel Image ... OK
OK

Starting kernel ...
```

in netconsole, but it never responds to ssh. So the procedure as-is no longer works, and I'm not sure what the problem is.

I wanted to go through the procedure again because if I could get it to work again with squeeze, I was then going to try to build a rootfs with wheezy.

Thank you! :)

bell

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

November 03, 2011 12:52PM

Registered: 5 years ago

Posts: 5

Hanker,

I've seen the exactly same problem as you had. With Jeff's v2.6.37 uImage/uInitrd, I cannot SSH in. My solution was to install the "linux-image-2.6-kirkwood" kernel in the debootstrap environment (Although Vlad said "it's not really a good idea to install the kernel in a debootstrap environment"). In other words, instead of using the block of code downloading/installing Jeff's v2.6.37 uImage/uInitrd, I did

```
apt-get install linux-image-2.6-kirkwood
mkimage -A arm -O linux -T kernel -C none -a 0x00008000 -e 0x00008000 -n Linux-2.6.32-5-kirkwood -d /boot/vmlinuz
mkimage -A arm -O linux -T ramdisk -C gzip -a 0x00000000 -e 0x00000000 -n initramfs-2.6.32-5-kirkwood -d /boot/initrd
cd /boot
ln -s ../uImage uImage
ln -s ../uInitrd uInitrd
```

The USB flash drive works for my GoFlex Net, and I can SSH in.

I also have a Dockstar with Debian installed a couple of months ago (before Jeff's script was broken). When I replaced its USB drive by the one I created for GoFlex Net, I cannot SSH in. I noticed that my Dockstar debian kernel version is 2.6.32-30, but the one I just created for GoFlex Net has 2.6.32-38. Maybe some kernels work exclusively for Dockstar, and some work exclusively for GoFlex Net.

hanker

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

November 03, 2011 01:18PM

Registered: 5 years ago

Posts: 58

bell Wrote:

```
-----
> Hanker,
>
> I've seen the exactly same problem as you had.
> With Jeff's v2.6.37 uImage/uInitrd, I cannot SSH
> in. My solution was to install the
> "linux-image-2.6-kirkwood" kernel in the
```

> debootstrap environment (Although Vlad said "it's
> not really a good idea to install the kernel in a
> debootstrap environment").

Got it. Very good! I will try that and hope it works.

Thank you :)

[hanker](#)

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

November 03, 2011 05:19PM

Registered: 5 years ago

Posts: 58

bell said:

>install the "linux-image-2.6-kirkwood" kernel in the debootstrap environment

Thanks for the suggestion. The 2.6-kirkwood kernel installed in the emulated armel environment with no errors, and the boot flash drive I ended up with boots fine. This is a good approach because it makes it so that the procedure isn't dependent on custom files downloaded from doozan.com. Not that doozan.com is a bad thing ;) It's just that it seems cleaner and more fault-tolerant to use software from the debian repositories. Now I can produce rootfs for my testing Dockstar at will.

FWIW, I had all sorts of problems with the emulated armel environment in my early attempts to run through this procedure - various weird error messages, segfaults, failures of the debootstrap --second-stage to complete, qemu errors, etc. This was when I was doing the procedure on a system booted in Debian squeeze and then on a system booted in Ubuntu 10.10. I'm now using a system running an up-to-date xubuntu 11.10, and now the procedure, including the kernel install, works flawlessly. And I note that the emulated armel environment now thinks it's running a 3.0 kernel rather than a 2.6 kernel:

```
# uname -a
Linux dm9-Inspiron-910 3.0.0-12-generic #20-Ubuntu SMP Fri Oct 7 14:50:42 UTC 2011 armv7l GNU/Linux
```

Thanks for your help. :)

[hanker](#)

Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC

November 04, 2011 08:33AM

Registered: 5 years ago

Posts: 58

New article so the thread will show new traffic:

After playing with this new install for a while, I have to say that things are not exactly flawless. cron is going nuts and spiking the cpu to 100% about half the time. I haven't put anything in crontab - it's blank. This is on a fresh install, just idling.

I did this:

```
apt-get purge cron
```

That stopped cron.

I noticed that the system time is way off:

```
# date
Thu Jan  1 07:48:19 CST 1970
```

Huh? Something wrong with ntp maybe?

```
# ntpq -p
No association ID's returned
```

Hmmm... So try this:

```
# dpkg-reconfigure ntp
Stopping NTP server: ntpd.
dpkg-maintscript-helper: warning: environment variable DPKG_MAINTSCRIPT_NAME missing
dpkg-maintscript-helper: warning: environment variable DPKG_MAINTSCRIPT_PACKAGE missing
```



```
dpkg-maintscript-helper: warning: environment variable DPKG_MAINTSCRIPT_NAME missing
dpkg-maintscript-helper: warning: environment variable DPKG_MAINTSCRIPT_PACKAGE missing
Starting NTP server: ntpd.
```

Oops! So I googled a little and found this:

<https://bugs.launchpad.net/ubuntu/+source/debconf/+bug/706354>

which says

```
dpkg-maintscript-helper: warning: environment variable DPKG_MAINTSCRIPT_PACKAGE missing
...
Affects          Status          Importance    Assigned to
...
ntp (Ubuntu)     Confirmed      Low           Unassigned
...
In a very quick analysis I found that all the packages (on my system) that
use "if dpkg-maintscript-helper supports ..." are affected.
...
This bug is in Oneiric Ocelot
```

Ouch. :\

So it appears that ntp **is** messed up on this debootstrap install done under Oneiric Ocelot, and it looks like this is not going to be fixed any time soon. I'm not sure why an Ubuntu bug affects a Debian install done with debootstrap, but it evidently does. I guess the next step is to try doing the debootstrap on a system booted with Debian wheezy. Or is there another way to fix this dpkg-reconfigure problem?

Vlad	Registered: 4 years ago
Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC	Posts: 128
November 06, 2011 12:08PM	

Hi hanker,

The thing is that UBIT can boot the system even without uInitd and I thought that UBOOT should be able to do this too. I updated the relevant part of the howto, though I still have no idea why Jeff's kernel doesn't work for you and bell.

>This is a good approach because it makes it so that the procedure isn't dependent on custom files downloaded from >doozan.com. Not that doozan.com is a bad thing ;) It's just that it seems cleaner and more fault-tolerant to use software from >the debian repositories. Now I can produce rootfs for my testing Dockstar at will.

Debian kernel doesn't support GoFlex Net. bell's device still has Dockstar's arcNumber which is the reason why he can use that kernel. On the other hand, because of that he can't use the SATA ports. As soon as one changes the arcNumber to GoFlex, one needs a compatible kernel which is **not** in Debian's repo.

> Huh? Something wrong with ntp maybe?

There's nothing wrong with ntpd, except that I forgot to write how to configure it properly. Furthermore, ntpdate must be removed prior using ntpd. Check out the relevant section of the updated howto.

>So it appears that ntp **is** messed up on this debootstrap install done under Oneiric Ocelot, and it looks like this is not >going to be fixed any time soon. I'm not sure why an Ubuntu bug affects a Debian install done with debootstrap, but it >evidently does. I guess the next step is to try doing the debootstrap on a system booted with Debian wheezy. Or is there >another way to fix this dpkg-reconfigure problem?

Once again, a misconfigured ntpd has absolutely nothing to do with debootstrap. FYI, if you run "dpkg-reconfigure ntp" on a rootfs from Jeff's script, you get exactly the same warnings. Warnings are not errors and since there are no options that dpkg could configure when installing ntpd, all it can do is just restart the service.

hanker	Registered: 5 years ago
Re: Howto: Create Debian rootfs image for your GoFlex/Dockstar via debootstrap on a Ubuntu/Debian PC	Posts: 58
November 06, 2011 06:27PM	

Vlad Wrote:

> Hi hanker,
>
> The thing is that UBIT can boot the system even
> without uInitd and I thought that UBOOT should be
> able to do this too.
> I updated the relevant part of the howto, though I

> still have no idea why Jeff's kernel doesn't work
> for you and bell.

Thank you for your reply. I'm learning a lot, and I now have a working Debian wheezy/kernel 3.0.0-1 boot flash drive for my Dockstar.

Thank you again for this guide and your attention to it! :)