



Immutable Page

Info

Attachments

More Actions:

Ubuntu Wiki

Login

Help

RaspberryPi

Raspberry Pi

With the release of the Raspberry Pi 2 Model B and its ARMv7-based BCM2709 processor, it is now possible to run Ubuntu directly on the Raspberry Pi.

Note that the information on this page currently only applies to the Raspberry Pi 2 and 3, not the original Raspberry Pi. The Raspberry Pi 3 does not (yet) work with official Ubuntu images out of the box, but unofficial images are available.

Ubuntu 16.04 LTS

Download

- Raspberry Pi 2: [ubuntu-16.04-preinstalled-server-armhf+raspi2.img.xz](#) (4G image, 214M compressed)
 - Official image.
- Raspberry Pi 3: [ubuntu-16.04-preinstalled-server-armhf+raspi3.img.xz](#) (4G image, 216M compressed)
 - This [Raspberry Pi 3](#) image is built by [Ryan Finnie](#) using modified Xenial packages in a PPA ([ppa:ubuntu-raspi2/ppa-rpi3](#)), and is not supported directly by Canonical.

Login username is "ubuntu", password is "ubuntu". You will be asked to change the password on first login.

Installation

Installation is the same as other Raspberry Pi images; [a generic installation guide from raspberrypi.org](#) is available [here](#).

For example:

```
xzcat ubuntu.img.xz | sudo dd of=/dev/mmcblk0
```

Or using ddrescue (must decompress the image first):

```
unxz ubuntu.img.xz
sudo ddrescue -D -d --force ubuntu.img /dev/mmcblk0
```

Optional PPA

While the official image includes compatible firmware, bootloader and kernel, there are a few packages available in an unofficial PPA ([ppa:ubuntu-raspi2/ppa](#)) which are useful on the Raspberry Pi, including:

- `libraspberrypi-bin` - [VideoCore](#) utilities from <https://github.com/raspberrypi/userland> such as `vcgencmd`, `raspistill`, etc.
- `libraspberrypi-bin-nonfree` - Binary [VideoCore](#) utilities not provided in the open source userland repository, currently `vcdbg` and `edidparser`.
- `xserver-xorg-video-fbturbo` - An accelerated x.org video driver, though this is limited to hardware accelerated window moving/scrolling on the Raspberry Pi.
- `hello-dkms` - Not strictly to do with the Raspberry Pi, but a small example DKMS project to test building kernel DKMS modules.

To install:

```
sudo add-apt-repository ppa:ubuntu-raspi2/ppa
sudo apt-get update
```

Desktop

This is a small ubuntu-server image. If you want a full desktop, go ahead and do so:

```
$ sudo apt-get install xubuntu-desktop # or
$ sudo apt-get install lubuntu-desktop # or
$ sudo apt-get install kubuntu-desktop # etc
```

Ubuntu (Unity) and Ubuntu-GNOME just display a blank screen, presumably because they require 3D compositing. Kubuntu works but is slow unless you

Table of Contents

Contents

- Raspberry Pi**
 - Ubuntu 16.04 LTS**
 - [Download](#)
 - [Installation](#)
 - [Optional PPA](#)
 - [Desktop](#)
 - [Accelerated X driver](#)
 - [VideoCore](#)
 - [Cross-upgrading 14.04 to 16.04](#)
 - Ubuntu 14.04 LTS**
 - [Download](#)
 - [Usage](#)
 - [Root resize](#)
 - [Swap](#)
 - [Wifi firmware](#)
 - [SSH server](#)
 - [Serial console](#)
 - [Kernel](#)
 - [Building](#)
 - [Links](#)
- Snappy Ubuntu Core**

turn off desktop effects under System Settings. Xubuntu and Lubuntu work fine out of the box.

Accelerated X driver

An accelerated x.org video driver is available (fbturbo), though this is limited to hardware accelerated window moving/scrolling on the Raspberry Pi. Install the optional PPA above, then:

```
$ sudo apt-get install xserver-xorg-video-fbturbo
```

Then add this to /etc/X11/xorg.conf (create if it doesn't already exist):

```
Section "Device"
    Identifier "Raspberry Pi FBDEV"
    Driver "fbturbo"
    Option "fbdev" "/dev/fb0"
    Option "SwapbuffersWait" "true"
EndSection
```

VideoCore

As with Raspbian, VideoCore packages are available. Install the optional PPA above, then:

```
$ sudo apt-get install libraspberrypi-bin libraspberrypi-dev
```

However, since these packages are compiled from source during build, the files are installed in their "proper" locations in /usr. Some third-party scripts may expect e.g. /opt/vc/bin/vcgencmd; if so, this hack should do it:

```
$ sudo ln -s /usr /opt/vc
```


(Raspbian packages use precompiled repositories during build, which install in /opt/vc.) vcdbg and edidparser are not part of the open source package and must be installed separately:

```
$ sudo apt-get install libraspberrypi-bin-nonfree
```

Cross-upgrading 14.04 to 16.04

You can upgrade an old unofficial 14.04 installation to the official 16.04 installation, though it takes a number of additional steps.

Note that Ubuntu's setup uses u-boot as an intermediary bootloader, which is different from the previous system of the RPI2 booting the kernel directly. This will be reflected in the upgrade procedure.

 Once you begin this procedure, if you reboot the installation without completing the entire upgrade procedure, you will be left with an unbootable system.

First, remove a number of PPA packages which are obsoleted / incompatible with the 16.04.

```
apt-get --purge remove rpi2-ubuntu-errata raspberrypi-bootloader-nokernel \
    linux-image-rpi2 flash-kernel
```

Back up and remove the apt PPA configuration and module blacklists (the latter will be provided directly by the 4.4.0 kernel package).

```
mkdir -p /root/xenial-upgrade
tar zcvf /root/xenial-upgrade/etc.tar.gz \
    /etc/modprobe.d/rpi2.conf \
    /lib/modules-load.d/rpi2.conf \
    /etc/apt/preferences.d/rpi2-ppa \
    /etc/apt/sources.list.d/fo0bar-rpi2* \
    /etc/apt/trusted.gpg.d/fo0bar-rpi2*
rm -f \
    /etc/modprobe.d/rpi2.conf \
    /lib/modules-load.d/rpi2.conf \
    /etc/apt/preferences.d/rpi2-ppa \
    /etc/apt/sources.list.d/fo0bar-rpi2* \
    /etc/apt/trusted.gpg.d/fo0bar-rpi2*
```

Back up and remove the contents of /boot/firmware, which will be recreated.

```
tar zcvf /root/xenial-upgrade/firmware.tar.gz /boot/firmware/*
rm -rf /boot/firmware/*
```

Update apt sources without the old PPA configuration.

```
apt-get update
```

Run do-release-upgrade as normal. When asked to reboot at the end, **do not**, and select "n" instead.

```
do-release-upgrade -d
# -d will be unneeded once 16.04.1 is released
```

Install new firmware, u-boot and 4.4.0 kernel metapackages.

```
apt-get install u-boot-rpi u-boot-tools linux-raspi2 linux-firmware-raspi2 \
    linux-firmware flash-kernel
```

Install the RPI2 DT-compatible u-boot image.

```
wget -O /tmp/mkknlimg https://raw.githubusercontent.com/raspberrypi/tools/master/mkimage/mkknlimg
chmod 0755 /tmp/mkknlimg
/tmp/mkknlimg --dtok /usr/lib/u-boot/rpi_2/u-boot.bin /boot/firmware/uboot.bin
```

Install basic config.txt and cmdline.txt configurations. If your root device is not on the second SD partition (uncommon) or you have a more advanced configuration, recreate them here.

```
cat <<"EOM" >/boot/firmware/config.txt
kernel=uboot.bin
dtparam=i2c_arm=on
dtparam=spi=on
EOM
```

```
cat <<"EOM" >/boot/firmware/cmdline.txt
net.ifnames=0 dwc_otg.lpm_enable=0 console=ttyAMA0,115200 console=tty1 root=/dev/mmcblk0p2 rootfstype=ext4 elevator=
EOM
```

Update the initrd and re-flash the kernel configuration.

```
update-initramfs -u
flash-kernel
```

Reboot!

```
reboot
```

Optionally add ppa:ubuntu-raspi2/ppa as described above.

Ubuntu 14.04 LTS

An Ubuntu 14.04 LTS (Trusty Tahr) image is available for the Raspberry Pi 2, which combines the released 14.04 distribution with a PPA containing kernels and firmware which work on the Raspberry Pi 2.

 This image, along with the one-off kernel it installs, is no longer maintained. Please use the 16.04 Xenial image instead. 

Download

- [Download 2015-04-06-ubuntu-trusty.zip](#) (GPG signature)
- 152MiB ZIP, 1.75GiB uncompressed image
- Login username is "ubuntu", password is "ubuntu"
- Raspberry Pi 3 images are not available.

Usage

ROOT RESIZE

There are no Raspbian-specific utilities included, specifically no automatic root resizer. However, it's not hard to do manually. Once booted:

```
$ sudo fdisk /dev/mmcblk0
```

Delete the second partition (d, 2), then re-create it using the defaults (n, p, 2, enter, enter), then write and exit (w). Reboot the system, then:

```
$ sudo resize2fs /dev/mmcblk0p2
```

SWAP

There is no swap partition/file included. If you want swap, it's recommended you do:

```
$ sudo apt-get install dphys-swapfile
```

You should have a (resized) SD card at least 4GB, because by default it will want to create a ~2GB swapfile.

WIFI FIRMWARE

If you are using a wifi dongle, you will likely need to get the linux-firmware package:

```
$ sudo apt-get install linux-firmware
```

SSH SERVER

If you would like to install an SSH server for remote access:

```
$ sudo apt-get install openssh-server
```

SERIAL CONSOLE

To enable the serial console, change the /boot/cmdline.txt as follows:

```
dwc_otg.lpm_enable=0 console=ttyAMA0,115200 kgdboc=ttyAMA0,115200 console=tty1 root=/dev/mmcblk0p2 rootwait
```

and add a new file /etc/init/ttyAMA0.conf:

```
start on stopped rc or RUNLEVEL=[12345]
stop on runlevel [!12345]

respawn
exec /sbin/getty -L 115200 ttyAMA0 vt102
```

Kernel

The kernel used by the Raspberry Pi 2 port is an Ubuntu-style kernel package of an "rpi2" flavor, e.g. linux-image-3.18.0-20-rpi2. Currently it is comprised of the following functionality:

- Mainline 3.18.x
- Fork of [Ubuntu-3.18.0-14.15 from the ubuntu-vivid git tree](#)
 - Provides extra functionality such as aufs
 - Also includes additional stability fixes, many of which have been rolled into mainline post-3.18.7
- Raspberry Pi-specific patches from the [rpi-3.18.y branch of Raspberry Pi's linux git tree](#)

Ubuntu's 3.18 development is no longer active, as they moved on to 3.19 to be released with 15.04 vivid. However, mainline 3.18 was designated an LTS kernel release, and is still getting active security/stability updates. Because of this, 3.18 will likely remain the "supported" kernel of this port. (Again, this is an unofficial port and no support guarantee is implied.)

Building

- The script used to build the images is available [here](#).

If you want to build an image on your x86 Ubuntu host, install qemu-user-static package and edit the script to use "qemu-debootstrap --arch armhf" instead of "debootstrap":

```
qemu-debootstrap --arch armhf $RELEASE $R http://ports.ubuntu.com/
```

Links

- [raspberrypi.org forums thread](#)
- [Reddit thread](#)
- [Ryan Finnie original blog announcement](#)
- PPAs:
 - [ppa:fo0bar/rpi2](#) - Stable packages (enabled in the default image)
 - [ppa:fo0bar/rpi2-staging](#) - Staging builds
 - [ppa:fo0bar/rpi2-nightly](#) - Nightly automatic builds of some packages

Snappy Ubuntu Core

[Snappy Ubuntu Core](#) is a new rendition of Ubuntu with transactional updates - a minimal server image with the same libraries as today's Ubuntu, but applications are provided through a simpler mechanism. [A developer preview is available](#) for the Raspberry Pi 2.

ARM/RaspberryPi (last edited 2016-05-06 19:19:11 by fo0bar)