

Installing Docker on Ubuntu 12.04 64-bit

published by admin on Sat, 04/11/2015 - 20:31

install the backported kernel

```
sudo apt-get update
```

```
sudo apt-get install linux-image-generic-lts-raring linux-headers-generic-lts-raring
```

reboot

```
sudo reboot
```

Docker is available as a Debian package, which makes installation easy.

First add the Docker repository key to your local keychain. You can use the apt-key command to check the fingerprint matches: 36A1 D786 9245 C895 0F96 6E92 D857 6A8B A88D 21E9

```
sudo sh -c "wget -qO- https://get.docker.io/gpg | apt-key add -"
```

Add the Docker repository to your apt sources list, update and install the lxc-docker package.

You may receive a warning that the package isn't trusted. Answer yes to continue installation.

```
sudo sh -c "echo deb http://get.docker.io/ubuntu docker main\ > /etc/apt/sources.list.d/docker.list"
```

```
sudo apt-get update
```

```
sudo apt-get install lxc-docker
```

Now verify that the installation has worked by downloading the ubuntu image and launching a container.

```
sudo docker run -i -t ubuntu /bin/bash
```

Type exit to exit

-OR-

Docker.io provides a shell script for getting you up and running:

```
#!/bin/sh set -e # # This script is meant for quick & easy install via: # 'curl -sL https://get.docker.io/[1] | sh' # or: # 'wget -qO- https://get.docker.io/[1] | sh' # # # Docker Maintainers: # To update this script on https://get.docker.io [2], # use hack/release.sh during a normal release, # or the following one-liner for script hotfixes: # s3cmd put --acl-public -P hack/install.sh s3://get.docker.io/index # url='https://get.docker.io/[1] command_exists() { command -v "$@" > /dev/null 2>&1 } case "$(uname -m)" in *64) ;; *) echo >&2 'Error: you are not using a 64bit platform.' echo >&2 'Docker currently only supports 64bit platforms.' exit 1 ;; esac if command_exists docker || command_exists lxc-docker; then echo >&2 'Warning: "docker" or "lxc-docker" command appears to already exist.' echo >&2 'Please ensure that you do not already have docker installed.' echo >&2 'You may press Ctrl+C now to abort this process and rectify this situation.' ( set -x; sleep 20 ) fi sh_c='sh -c' if [ "$(whoami 2>/dev/null || true)" != 'root' ]; then if command_exists sudo; then sh_c='sudo sh -c' elif command_exists su; then sh_c='su -c' else echo >&2 'Error: this installer needs the ability to run commands as root.' echo >&2 'We are unable to find either "sudo" or "su" available to make this happen.' exit 1 fi fi curl=' ' if command_exists curl;
```

```
then curl='curl -sL' elif command_exists wget; then curl='wget -qO-' elif command_exists busybox
&& busybox --list-modules | grep -q wget; then curl='busybox wget -qO-' fi # perform some very
rudimentary platform detection lsb_dist="" if command_exists lsb_release; then
lsb_dist="$(lsb_release -si)" fi if [ -z "$lsb_dist" ] && [ -r /etc/lsb-release ]; then lsb_dist="$(. /etc/lsb-
release && echo "$DISTRIB_ID)" fi if [ -z "$lsb_dist" ] && [ -r /etc/debian_version ]; then
lsb_dist='Debian' fi case "$lsb_dist" in Ubuntu|Debian) export DEBIAN_FRONTEND=noninteractive
did_apt_get_update= apt_get_update() { if [ -z "$did_apt_get_update" ]; then ( set -x; $sh_c 'sleep 3;
apt-get update' ) did_apt_get_update=1 fi } # TODO remove this section once device-mapper lands
if ! grep -q aufs /proc/filesystems && ! $sh_c 'modprobe aufs'; then kern_extras="linux-image-
extra-$(uname -r)" apt_get_update ( set -x; $sh_c 'sleep 3; apt-get install -y -q "$kern_extras" ) ||
true if ! grep -q aufs /proc/filesystems && ! $sh_c 'modprobe aufs'; then echo >&2 'Warning: tried to
install "$kern_extras" (for AUFS)' echo >&2 ' but we still have no AUFS. Docker may not work.
Proceeding anyways!' ( set -x; sleep 10 ) fi fi [ ! -e /usr/lib/apt/methods/https ]; then
apt_get_update ( set -x; $sh_c 'sleep 3; apt-get install -y -q apt-transport-https' ) fi if [ -z "$curl" ];
then apt_get_update ( set -x; $sh_c 'sleep 3; apt-get install -y -q curl' ) curl='curl -sL' fi ( set -x $sh_c
"$curl ${url}gpg | apt-key add -" $sh_c "echo deb ${url}ubuntu docker main >
/etc/apt/sources.list.d/docker.list" $sh_c 'sleep 3; apt-get update; apt-get install -y -q lxc-docker' ) if
command_exists docker && [ -e /var/run/docker.sock ]; then ( set -x $sh_c 'docker run busybox echo
"Docker has been successfully installed!" ) fi exit 0 ;; Gentoo) if [ "$url" = "https://test.docker.io/" [3]
]; then echo >&2 echo >&2 ' You appear to be trying to install the latest nightly build in Gentoo.'
echo >&2 ' The portage tree should contain the latest stable release of Docker, but' echo >&2 ' if
you want something more recent, you can always use the live ebuild' echo >&2 ' provided in the
"docker" overlay available via layman. For more' echo >&2 ' instructions, please see the following
URL:' echo >&2 ' https://github.com/tianon/docker-overlay#using-this-overlay' [4] echo >&2 '
After adding the "docker" overlay, you should be able to:' echo >&2 ' emerge -av =app-
emulation/docker-9999' echo >&2 exit 1 fi ( set -x $sh_c 'sleep 3; emerge app-emulation/docker' )
exit 0 ;; esac echo >&2 echo >&2 ' Either your platform is not easily detectable, is not supported by
this' echo >&2 ' installer script (yet - PRs welcome!), or does not yet have a package for' echo >&2
' Docker. Please visit the following URL for more detailed installation' echo >&2 ' instructions:' echo
>&2 echo >&2 ' http://docs.docker.io/en/latest/installation/' [5] echo >&2 exit 1
```

* Stolen mercilessly from the good people at Docker.io
(<http://docs.docker.io/en/latest/installation/ubuntu/linux/> [6])

Source URL: <https://blackhillsystems.com/?q=node/46>

Links

- [1] <https://get.docker.io/>
- [2] <https://get.docker.io>
- [3] <https://test.docker.io/>
- [4] <https://github.com/tianon/docker-overlay#using-this-overlay>
- [5] <http://docs.docker.io/en/latest/installation/>
- [6] <http://docs.docker.io/en/latest/installation/ubuntu/linux/>