

**NAME**

debootstrap – Bootstrap a basic Debian system

**SYNOPSIS**

**debootstrap** [**OPTION...**] *SUITE TARGET* [*MIRROR* [*SCRIPT*]]

**debootstrap** [**OPTION...**] **—second-stage**

**DESCRIPTION**

**debootstrap** bootstraps a basic Debian system of *SUITE* into *TARGET* from *MIRROR* by running *SCRIPT*. *MIRROR* can be an http:// or https:// URL, a file:/// URL, or an ssh:/// URL.

The *SUITE* may be a release code name (eg, sid, stretch, jessie) or a symbolic name (eg, unstable, testing, stable, oldstable)

Notice that file:/ URLs are translated to file:/// (correct scheme as described in RFC1738 for local file-names), and file:// will **not** work. ssh://USER@HOST/PATH URLs are retrieved using **scp**; use of **ssh-agent** or similar is strongly recommended.

**Debootstrap** can be used to install Debian in a system without using an installation disk but can also be used to run a different Debian flavor in a **chroot** environment. This way you can create a full (minimal) Debian installation which can be used for testing purposes (see the **EXAMPLES** section). If you are looking for a chroot system to build packages please take a look at **pbuilder**.

**OPTIONS**

**—arch=ARCH**

Set the target architecture (use if dpkg isn't installed). See also **—foreign**.

**—include=alpha,beta**

Comma separated list of packages which will be added to download and extract lists.

**—exclude=alpha,beta**

Comma separated list of packages which will be removed from download and extract lists. WARNING: you can and probably will exclude essential packages, be careful using this option.

**—extra-suites=alpha,beta**

Also use packages from the listed suites of the archive.

**—components=alpha,beta**

Use packages from the listed components of the archive.

**—no-resolve-deps**

By default, debootstrap will attempt to automatically resolve any missing dependencies, warning if any are found. Note that this is not a complete dependency resolve in the sense of dpkg or apt, and that it is far better to specify the entire base system than rely on this option. With this option set, this behaviour is disabled.

**—log-extra-deps**

If you want to record additional dependencies when resolving package dependencies, set this option to track them through debootstrap.log.

**—variant=minbase|bulld|fakechroot**

Name of the bootstrap script variant to use. Currently, the variants supported are minbase, which only includes *required* packages and apt; bulld, which installs the build-essential packages and fakechroot, which installs the packages without root privileges. The default, with no **—variant=X** argument, is to create a base Debian installation with all packages of priority *required* and *important*, including apt.

**—merged-usr**

Create /{bin,sbin,lib}/ symlinks pointing to their counterparts in /usr/. (Default for most variants.)

- no-merged-usr**  
Do not create `{bin,sbin,lib}/` symlinks pointing to their counterparts in `/usr/`. (Default for the `buildd` variant.)
- keyring=KEYRING**  
Override the default keyring for the distribution being bootstrapped, and use *KEYRING* to check signatures of retrieved Release files.
- no-check-gpg**  
Disables checking gpg signatures of retrieved Release files.
- force-check-gpg**  
Forces checking Release file signatures, disabling automatic fallback to HTTPS in case of a missing keyring. Incompatible with the previous option.
- verbose**  
Produce more info about downloading.
- print-debs**  
Print the packages to be installed, and exit. Note that an empty or non-existing TARGET directory must be specified so that `debootstrap` can download Packages files to determine which packages should be installed, and to resolve dependencies. The TARGET directory will be deleted unless `--keep-debootstrap-dir` is specified.
- download-only**  
Download packages, but don't perform installation.
- foreign**  
Do the initial unpack phase of bootstrapping only, for example if the target architecture does not match the host architecture. A copy of `debootstrap` sufficient for completing the bootstrap process will be installed as `/debootstrap/debootstrap` in the target filesystem. You can run it with the `--second-stage` option to complete the bootstrapping process.
- second-stage**  
Complete the bootstrapping process. Other arguments are generally not needed.
- second-stage-target=DIR**  
Run second stage in a subdirectory instead of root. (can be used to create a foreign chroot) (requires `--second-stage`)
- keep-debootstrap-dir**  
Don't delete the `/debootstrap` directory in the target after completing the installation.
- cache-dir=DIR**  
Cache .deb files under directory. It should be an absolute path.
- unpack-tarball=FILE**  
Acquire .debs from gzipped tarball FILE (specified with absolute path) instead of downloading via HTTP(S).
- make-tarball=FILE**  
Instead of bootstrapping, make a gzipped tarball (written to FILE) of the downloaded packages. The resulting tarball may be passed to a later `--unpack-tarball`. Note that an empty or non-existing TARGET directory must be specified so that `debootstrap` can download packages to prepare the tarball. The TARGET directory will be deleted unless `--keep-debootstrap-dir` is specified.
- debian-installer**  
Used for internal purposes by the `debian-installer`
- extractor=TYPE**  
Override automatic .deb extractor selection to *TYPE*. Supported extractors are: `dpkg-deb` and `ar`.

**--no-check-certificate**

Do not check certificate against certificate authorities

**--certificate=FILE**

Use the client certificate stored in file (PEM)

**--private-key=FILE**

Read the private key from file

**--inrelease-path**

Determine the path to the InRelease file of the suite relative to the normal position of an InRelease file

**EXAMPLES**

To setup a *stretch* system:

```
debootstrap stretch ./stretch-chroot http://deb.debian.org/debian
```

```
debootstrap stretch ./stretch-chroot file:///LOCAL_MIRROR/debian
```

Full process to create a complete Debian installation of *sid* (unstable) in a chroot:

```
main # debootstrap sid sid-root http://deb.debian.org/debian/
[ ... watch it download the whole system ]
main # echo "proc sid-root/proc proc defaults 0 0" >> /etc/fstab
main # mount proc sid-root/proc -t proc
main # echo "sysfs sid-root/sys sysfs defaults 0 0" >> /etc/fstab
main # mount sysfs sid-root/sys -t sysfs
main # cp /etc/hosts sid-root/etc/hosts
main # chroot sid-root /bin/bash
```

**AUTHOR**

**debootstrap** was written by Anthony Towns <ajt@debian.org>. This manpage was written by Matt Kraai <kraai@debian.org>.