

---

# **ebuildtester Documentation**

***Release 41***

**Nicolas Bock**

**Jun 21, 2022**



---

## Contents

---

|          |                                           |           |
|----------|-------------------------------------------|-----------|
| <b>1</b> | <b>Dependencies</b>                       | <b>3</b>  |
| <b>2</b> | <b>Setting up a developer environment</b> | <b>5</b>  |
| <b>3</b> | <b>ebuildtester package</b>               | <b>7</b>  |
| 3.1      | Submodules . . . . .                      | 7         |
| 3.2      | ebuildtester.atom module . . . . .        | 7         |
| 3.3      | ebuildtester.docker module . . . . .      | 7         |
| 3.4      | ebuildtester.main module . . . . .        | 8         |
| 3.5      | ebuildtester.options module . . . . .     | 8         |
| 3.6      | ebuildtester.parse module . . . . .       | 8         |
| 3.7      | ebuildtester.utils module . . . . .       | 8         |
| 3.8      | Module contents . . . . .                 | 8         |
| <b>4</b> | <b>Introduction</b>                       | <b>9</b>  |
| 4.1      | Requirements . . . . .                    | 9         |
| 4.2      | Usage . . . . .                           | 9         |
| 4.3      | Command line arguments . . . . .          | 10        |
|          | <b>Python Module Index</b>                | <b>13</b> |
|          | <b>Index</b>                              | <b>15</b> |



This project supports Python 3.7, 3.8, 3.9, and 3.10. Other Python versions might work as well but are not regularly tested.

For locally testing changes it is very handy to install *tox* which automates the creation of Python virtual environments.



# CHAPTER 1

---

## Dependencies

---

- *docker*
- *fuse*





## CHAPTER 2

---

### Setting up a developer environment

---

```
$ python -m virtualenv venv
$ source venv/bin/activate
$ (venv) pip install -r requirements.txt
```

Install *ebuildtester* in the *virtualenv*:

```
$ (venv) python setup.py install
```

Run the development version:

```
$ (venv) ebuildtester ...
```



### 3.1 Submodules

### 3.2 ebuildtester.atom module

An Atom.

```
class ebuildtester.atom.Atom(atom)  
    Bases: object
```

```
exception ebuildtester.atom.AtomException  
    Bases: Exception  
    An exception in this class.
```

### 3.3 ebuildtester.docker module

```
class ebuildtester.docker.Docker(local_portage, overlay_dirs)  
    Bases: object
```

The Docker class.

```
cleanup()  
    Clean up.
```

```
execute(cmd)  
    Execute command in container.  
    cmd is a string which is executed within a bash shell.
```

```
remove()  
    Remove the docker container.
```

**shell()**

Run an interactive shell in container.

**exception** `ebuildtester.docker.ExecuteFailure`

Bases: `Exception`

Failure to execute command.

## 3.4 ebuildtester.main module

`ebuildtester.main.main()`

The main function.

## 3.5 ebuildtester.options module

Options and some initializations.

`ebuildtester.options.set_logfile(logfile)`

Add a logfile to logging.

## 3.6 ebuildtester.parse module

`ebuildtester.parse.parse_commandline(args)`

Parse the command line.

## 3.7 ebuildtester.utils module

`ebuildtester.utils.message_string(string)`

Return a string.

## 3.8 Module contents

# CHAPTER 4

---

## Introduction

---

This is a tool to test a Gentoo ebuild and its dependencies. The idea is that the ebuild is emerged in a clean (and current) stage3 Docker container.

### 4.1 Requirements

Using require [Docker](#) and [FUSE](#). Docker must be configured to use the [devicemapper](#) storage driver. This can be achieved with the following inside `/etc/docker/daemon.json`:

```
{
  "storage-driver": "devicemapper"
}
```

### 4.2 Usage

We are going to assume that the user has a local git clone of the portage tree in

```
/usr/local/git/gentoo
```

We have added a new ebuild and would like to verify that the build dependencies are all correct. We can build the package (ATOM) with:

```
ebuildtester --portage-dir /usr/local/git/gentoo \
  --atom ATOM \
  --use USE1 USE2
```

where we have specified two USE flags, USE1 and USE2. The *ebuildtester* command will now create a docker container and start installing the ATOM. All specified dependencies will be installed as well.

## 4.3 Command line arguments

The command understands the following command line arguments:

```
usage: ebuildtester [-h] [--version] [--atom ATOM [ATOM ...]] [--binhost BINHOST] [--
  ↪live-ebuild]
                        [--manual] --portage-dir PORTAGE_DIR [--overlay-dir OVERLAY_DIR]
  ↪[--update]
                        [--install-basic-packages] [--threads N] [--use USE [USE ...]]
                        [--global-use GLOBAL_USE [GLOBAL_USE ...]] [--unmask ATOM] [--
  ↪unstable]
                        [--gcc-version VER] [--python-single-target PYTHON_SINGLE_TARGET]
                        [--python-targets PYTHON_TARGETS] [--rm] [--storage-opt STORAGE_
  ↪OPT [STORAGE_OPT ...]]
                        [--with-X] [--with-vnc] [--profile PROFILE] [--features FEATURES_
  ↪[FEATURES ...]]
                        [--docker-image DOCKER_IMAGE] [--docker-command DOCKER_COMMAND] [-
  ↪-pull]
                        [--show-options] [--ccache CCACHE_DIR]
```

A dockerized approach to test a Gentoo package within a clean stage3.

options:

|                                             |                                               |
|---------------------------------------------|-----------------------------------------------|
| -h, --help                                  | show this help message and exit               |
| --version                                   | show program's version number and exit        |
| --atom ATOM [ATOM ...]                      | The package atom(s) to install                |
| --binhost BINHOST                           | Binhost URI                                   |
| --live-ebuild                               | Unmask the live ebuild of the atom            |
| --manual                                    | Install package manually                      |
| --portage-dir PORTAGE_DIR                   | The local portage directory                   |
| --overlay-dir OVERLAY_DIR                   | Add overlay dir (can be used multiple times)  |
| --update                                    | Update container before installing atom       |
| --install-basic-packages                    | Install basic packages after container starts |
| --threads N                                 | Use N (default 8) threads to build packages   |
| --use USE [USE ...]                         | The use flags for the atom                    |
| --global-use GLOBAL_USE [GLOBAL_USE ...]    | Set global USE flag                           |
| --unmask ATOM                               | Unmask atom (can be used multiple times)      |
| --unstable                                  | Globally 'unstable' system, i.e. ~amd64       |
| --gcc-version VER                           | Use gcc version VER                           |
| --python-single-target PYTHON_SINGLE_TARGET | Specify a PYTHON_SINGLE_TARGET                |
| --python-targets PYTHON_TARGETS             | Specify a PYTHON_TARGETS                      |

(continues on next page)

(continued from previous page)

```
--rm                                Remove container after session is done
--storage-opt STORAGE_OPT [STORAGE_OPT ...]
                                Storage driver options for all volumes (same as Docker param)
--with-X                            Globally enable the X USE flag
--with-vnc                        Install VNC server to test graphical applications
--profile PROFILE                The profile to use (default = default/linux/amd64/17.1)
--features FEATURES [FEATURES ...]
                                Set FEATURES, see https://wiki.gentoo.org/wiki/FEATURES
↪(default = ['-sandbox',
            '-usersandbox', 'userfetch'])
--docker-image DOCKER_IMAGE
                                Specify the docker image to use (default = gentoo/stage3)
--docker-command DOCKER_COMMAND
                                Specify the docker command
--pull                            Download latest docker image
--show-options                    Show currently selected options and defaults
--ccache CCACHE_DIR             Path to mount that contains ccache cache
```





### e

- `ebuildtester`, 8
- `ebuildtester.atom`, 7
- `ebuildtester.docker`, 7
- `ebuildtester.main`, 8
- `ebuildtester.options`, 8
- `ebuildtester.parse`, 8
- `ebuildtester.utils`, 8



## A

`Atom` (*class in `ebuildtester.atom`*), 7  
`AtomException`, 7

## C

`cleanup()` (*`ebuildtester.docker.Docker` method*), 7

## D

`Docker` (*class in `ebuildtester.docker`*), 7

## E

`ebuildtester` (*module*), 8  
`ebuildtester.atom` (*module*), 7  
`ebuildtester.docker` (*module*), 7  
`ebuildtester.main` (*module*), 8  
`ebuildtester.options` (*module*), 8  
`ebuildtester.parse` (*module*), 8  
`ebuildtester.utils` (*module*), 8  
`execute()` (*`ebuildtester.docker.Docker` method*), 7  
`ExecuteFailure`, 8

## M

`main()` (*in module `ebuildtester.main`*), 8  
`message_string()` (*in module `ebuildtester.utils`*), 8

## P

`parse_commandline()` (*in module `ebuildtester.parse`*), 8

## R

`remove()` (*`ebuildtester.docker.Docker` method*), 7

## S

`set_logfile()` (*in module `ebuildtester.options`*), 8  
`shell()` (*`ebuildtester.docker.Docker` method*), 7