



Linux Ubuntu 26.04 Manual Pages on command 'pip3.1'

\$ man pip3.1

PIP3(1) pip PIP3(1)

NAME

pip3 - package manager for Python packages

SYNOPSIS

pip3 <command> [options]

DESCRIPTION

pip3 is the PyPA recommended package manager for Python packages

OPTIONS

-h, --help

Show help.

(environment variable: PIP_HELP)

--debug

Let unhandled exceptions propagate outside the main subroutine, instead of logging them to stderr.

(environment variable: PIP_DEBUG)

--isolated

Run pip in an isolated mode, ignoring environment variables and user configuration.

(environment variable: PIP_ISOLATED)

--require-virtualenv

Allow pip to only run in a virtual environment; exit with an error otherwise.

(environment variable: PIP_REQUIRE_VIRTUALENV, PIP_REQUIRE_VENV)

--python <python>

Run pip with the specified Python interpreter.

(environment variable: PIP_PYTHON)

-v, --verbose

Give more output. Option is additive, and can be used up to 3 times.

(environment variable: PIP_VERBOSE)

-V, --version

Show version and exit.

(environment variable: PIP_VERSION)

-q, --quiet

Give less output. Option is additive, and can be used up to 3 times (corresponding to WARNING, ERROR, and CRITICAL logging levels).

(environment variable: PIP_QUIET)

--log <path>

Path to a verbose appending log.

(environment variable: PIP_LOG, PIP_LOG_FILE, PIP_LOCAL_LOG)

--no-input

Disable prompting for input.

(environment variable: PIP_NO_INPUT)

--keyring-provider <keyring_provider>

Enable the credential lookup via the keyring library if user input is allowed. Spec?

ify which mechanism to use [auto, disabled, import, subprocess]. (default: auto)

(environment variable: PIP_KEYRING_PROVIDER)

--proxy <proxy>

Specify a proxy in the form scheme://[user:passwd@]proxy.server:port.

(environment variable: PIP_PROXY)

--retries <retries>

Maximum attempts to establish a new HTTP connection. (default: 5)

(environment variable: PIP_RETRIES)

--timeout <sec>

Set the socket timeout (default 15 seconds).

(environment variable: PIP_TIMEOUT, PIP_DEFAULT_TIMEOUT)

--exists-action <action>

Default action when a path already exists: (s)witch, (i)gnore, (w)ipe, (b)ackup, (a)bort.

(environment variable: PIP_EXISTS_ACTION)

--trusted-host <hostname>

Mark this host or host:port pair as trusted, even though it does not have valid or any HTTPS.

(environment variable: PIP_TRUSTED_HOST)

--cert <path>

Path to PEM-encoded CA certificate bundle. If provided, overrides the default. See 'SSL Certificate Verification' in pip documentation for more information.

(environment variable: PIP_CERT)

--client-cert <path>

Path to SSL client certificate, a single file containing the private key and the certificate in PEM format.

(environment variable: PIP_CLIENT_CERT)

--cache-dir <dir>

Store the cache data in <dir>.

(environment variable: PIP_CACHE_DIR)

--no-cache-dir

Disable the cache.

(environment variable: PIP_NO_CACHE_DIR)

--disable-pip-version-check

Don't periodically check PyPI to determine whether a new version of pip is available for download. Implied with --no-index.

(environment variable: PIP_DISABLE_PIP_VERSION_CHECK)

--no-color

Suppress colored output.

(environment variable: PIP_NO_COLOR)

--use-feature <feature>

Enable new functionality, that may be backward incompatible.

(environment variable: PIP_USE_FEATURE)

--use-deprecated <feature>

Enable deprecated functionality, that will be removed in the future.

(environment variable: PIP_USE_DEPRECATED)

--resume-retries <resume_retries>

Maximum attempts to resume or restart an incomplete download. (default: 0)

(environment variable: PIP_RESUME_RETRIES)

COMMANDS

pip3-install(1)

Install packages.

pip3-download(1)

Download packages.

pip3-uninstall(1)

Uninstall packages.

pip3-freeze(1)

Output installed packages in requirements format.

pip3-lock(1)

Generate a lock file for requirements and their dependencies.

pip3-list(1)

List installed packages.

pip3-show(1)

Show information about installed packages.

pip3-check(1)

Verify installed packages have compatible dependencies.

pip3-search(1)

Search PyPI for packages.

pip3-wheel(1)

Build wheels from your requirements.

pip3-hash(1)

Compute hashes of package archives.

pip3-help(1)

Show help for pip commands.

AUTHOR

pip developers

COPYRIGHT

The pip developers