

Instantly share code, notes, and snippets.

saurabhshri / **pip.md**

Last active 2 weeks ago

☆ Star

<> Code   🔗 Revisions 5   ☆ Stars 85   🔗 Forks 15

Install and use pip in a local directory without root/sudo access.

 **pip.md**

# Install and use pip in a local directory without root/sudo access.

By: @saurabhshri

## Why?

Many users when are given server access, do not have root (or sudo) privileges and can not simply do `sudo apt-get install python-pip` . Here's an easy way you can install and use pip without root (or sudo) access in a local directory. **Note** : This works without `easy_install` too.

## How?

1. Download pip from an online repository : `wget https://bootstrap.pypa.io/get-pip.py`
2. Install the downloaded package into a local directory : `python get-pip.py --user`  
This will install pip to your local directory (`.local/bin`) .
3. Now you may navigate to this directory (`cd .local/bin`) and then use `pip` or better set your `$PATH` variable this directory to use `pip` anywhere :  
`PATH=$PATH:~/.local/bin` followed by `source ~/.bashrc` to apply the changes.

And that's it. You may now install all python modules you require via pip in a local directory without root.

## Gist of Gist

---

```
wget https://bootstrap.pypa.io/get-pip.py && python get-pip.py --user
cd .local/bin
./pip install <package_name> --user
```

**greggman** commented on Nov 26, 2017

This came up as the first hit on Google for installing python packages as user but it's unfortunately misleading

This does not install in a local directory, it installs an OS specific user folder which is global for that user. It doesn't install in `.local/bin` in the current folder. It installs in `~/.local/bin` (in your case). In my case it installs in `~/Library/Python/2.7/bin`.

In either case that's not "my local directory" it's effectively my "home directory" which is sadly not what I was looking for since I have various projects that need incompatible packages so needed actual solution for "local directory"

**emteeoh** commented on Nov 30, 2017

@greggman you can, probably even should, be using virtualenv. You create one env per project, and then install packages to that env. `--local` will, AFAIK, still work too.

```
pip install virtualenv --user
cd project1_folder
source bin/activate
virtualenv project1
pip install db=1.0.0 --user

cd ../project2_folder
virtualenv project2
source bin/activate
pip install db=2.0.0 --user
```

**kenichi-shibata** commented on Dec 5, 2017

Fixed the formatting for you

## Install and use pip in a local directory without root/sudo access.

---

By: @saurabhshri

## Why?

Many users when are given server access, do not have root (or sudo) privileges and can not simply do `sudo apt-get install python-pip`. Here's an easy way you can install and use pip without root (or sudo) access in a local directory. Note : This works without `easy_install` too.

## How?

Download pip from an online repository : `wget https://bootstrap.pypa.io/get-pip.py`

Install the downloaded package into a local directory : `python get-pip.py --user` This will install pip to your local directory (`.local/bin`).

Now you may navigate to this directory (`cd .local/bin`) and then use pip or better set your `$PATH` variable this directory to use pip anywhere : `PATH=$PATH:~/.local/bin` followed by `source ~/.bashrc` to apply the changes.

And that's it. You may now install all python modules you require via pip in a local directory without root.

## Gist of Gist

```
wget https://bootstrap.pypa.io/get-pip.py && python get-pip.py --user
cd .local/bin
./pip install <package_name> --user
```

**Moritz90** commented on Mar 28, 2018

@emteeoh @greggman

These days, you should probably be using [pipenv](#) instead of managing your virtualenv and your dependencies manually.

**texastony** commented on Dec 24, 2018

@emteeoh @greggman

These days, you should probably be using [pipenv](#) instead of managing your virtualenv and your dependencies manually.

In March of 2018, I tried to use pipenv and ran into many issues with locking. I decided to come back to it later, after a few more releases. It worked great on my local and servers I was the admin for, but where I was just the user it struggled where virtualenv did not. @Moritz90, have you continued to have success in various environments with pipenv?

**saurabhshri** commented on Sep 3, 2020

I have zero recollection of when I wrote this gist, I just found it back while searching how to do it myself. Sadly Github doesn't notify about gist comments, so I missed the activity here.

**@kenichi-shibata** Thank you for fixing, I have updated formatting in the gist.

**@greggman** I have been using virtual environments to manage that too, as suggested by others.

**Duttonide** commented on Dec 19, 2020

I am trying to local install through pip using the following command:

```
# while positioned at the repository's root
pip3 install -e installator
```

and getting the following error:

```
installator should either be a path to a local project or a VCS url beginning with svn+, git+, hg+, or
bzz+
```

can you kindly help me with this?

**Qutadah** commented on Mar 6, 2022

I have zero recollection of when I wrote this gist, I just found it back while searching how to do it myself. Sadly Github doesn't notify about gist comments, so I missed the activity here.

**@kenichi-shibata** Thank you for fixing, I have updated formatting in the gist.

**@greggman** I have been using virtual environments to manage that too, as suggested by others.

Can you please share your solution? Thanks!

**DeepaliKushwaha** commented on Jul 15, 2022

I'm getting an error when i run `wget https://bootstrap.pypa.io/get-pip.py`  
ERROR: no certificate subject alternative name matches requested host name 'bootstrap.pypa.io'.  
To connect to bootstrap.pypa.io insecurely, use `--no-check-certificate`.

So I changed it to `wget --no-check-certificate https://bootstrap.pypa.io/get-pip.py` . It worked.

But then `python get-pip.py --user` again gave error.

```
File "get-pip.py", line 1
<!DOCTYPE html>
^
SyntaxError: invalid syntax
```

I'm unable to resolve this issue.