

AVOIDING THE "LEFT-PAD" PROBLEM: HOW TO SECURE YOUR PIP INSTALL PROCESS

@aaronbassett
getadministrate.com

Administrate



@AARONBASSETT



Administrate

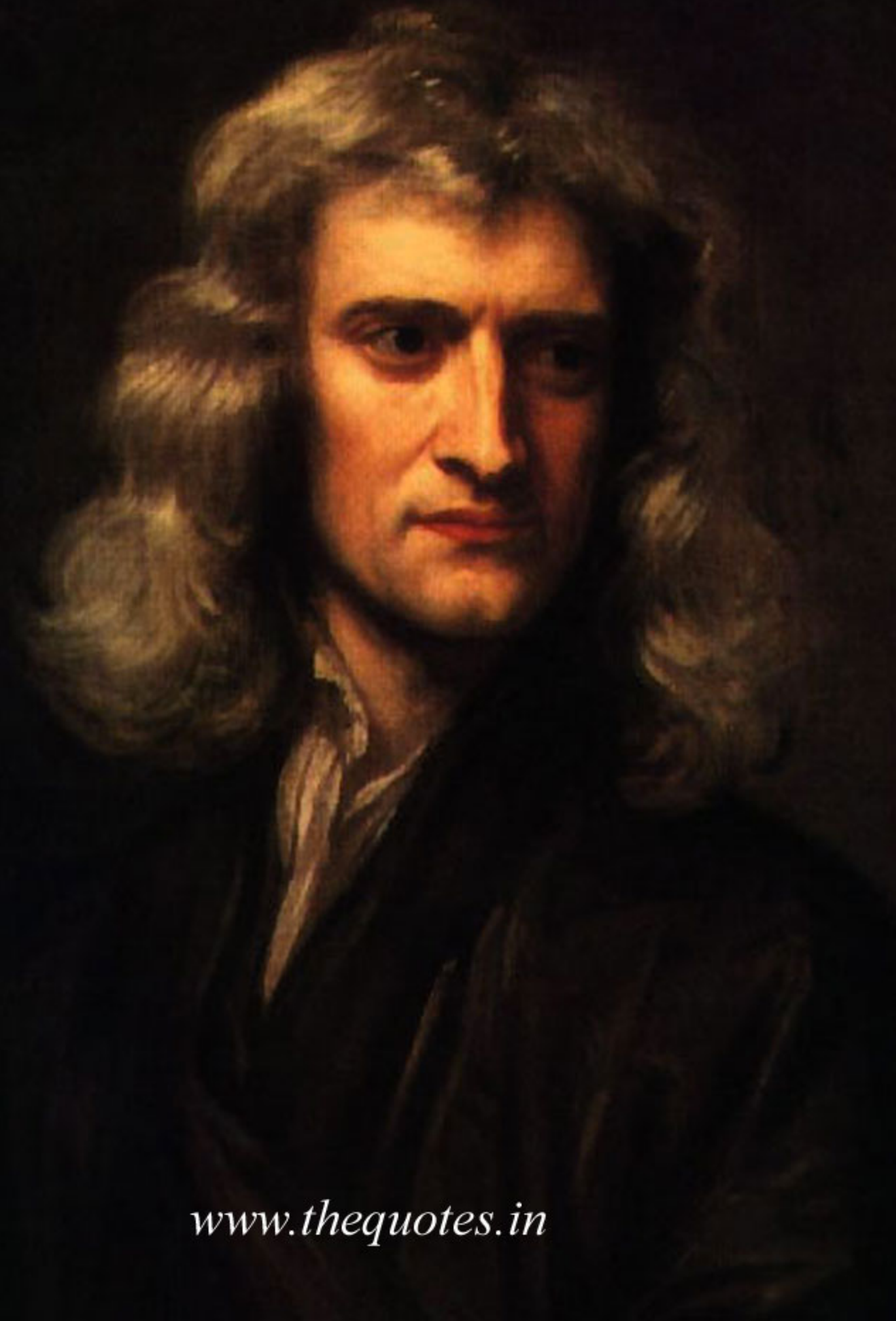
getadministrate.com

A person wearing a blue polo shirt is holding a large, brown cardboard box. The box has a circular logo on the front, which contains a stylized illustration of a truck and a car. The background is blurred, showing what appears to be a street scene with trees and a car. A semi-transparent dark blue horizontal band is overlaid across the middle of the image, containing the text "What are packages?".

What are packages?

If I have seen further than others, it is
by standing upon the shoulders of giants.

Isaac Newton



www.thequotes.in

“Most software today is very much like an Egyptian pyramid with millions of bricks piled on top of each other, with no structural integrity, but just done by brute force and thousands of slaves.”

Alan Kay



GET OFF MY LAWN!



aaronbassett/Djang0: Djang0

aab@administra...

GitHub, Inc. [US]

https://github.com/aaronbassett/Djang0

This repository

Search

Pull requests

Issues

Gist

I33tHack0rz420 / Djang0

Private

Unwatch

1

Star

873

Fork

0

<> Code

Issues 0

Pull requests 0

Projects 0

Wiki

Pulse

Graphs

Settings

Djang0 version 2! — Edit

2 commits

1 branch

0 releases

1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

I33tHack0rz420 committed on GitHub

Update README.md

Latest commit b683d5b on Aug 17

README.md

Update README.md

a month ago

README.md

Server backdoor

Django version 2!

Python v17

Downloads 9000+

No Virus

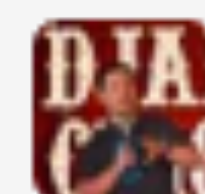
Totally Safe

Installation

```
sudo pip install Djang0
```




Issues Gist



 Unwatch ▼

1

 Star

873

 Fork

0

Wiki

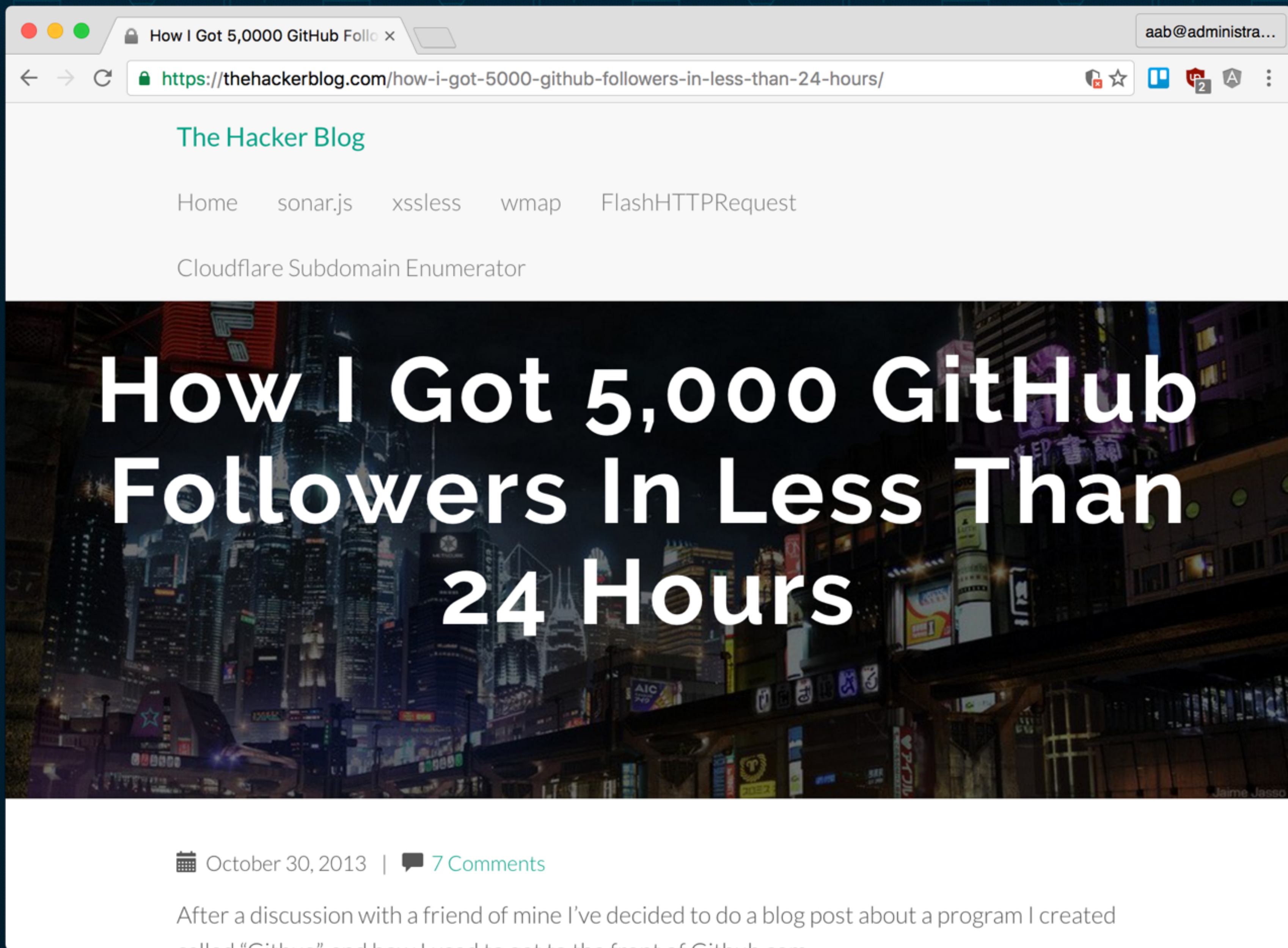
 Pulse

 Graphs

 Settings

 0 releases

 1 contributor



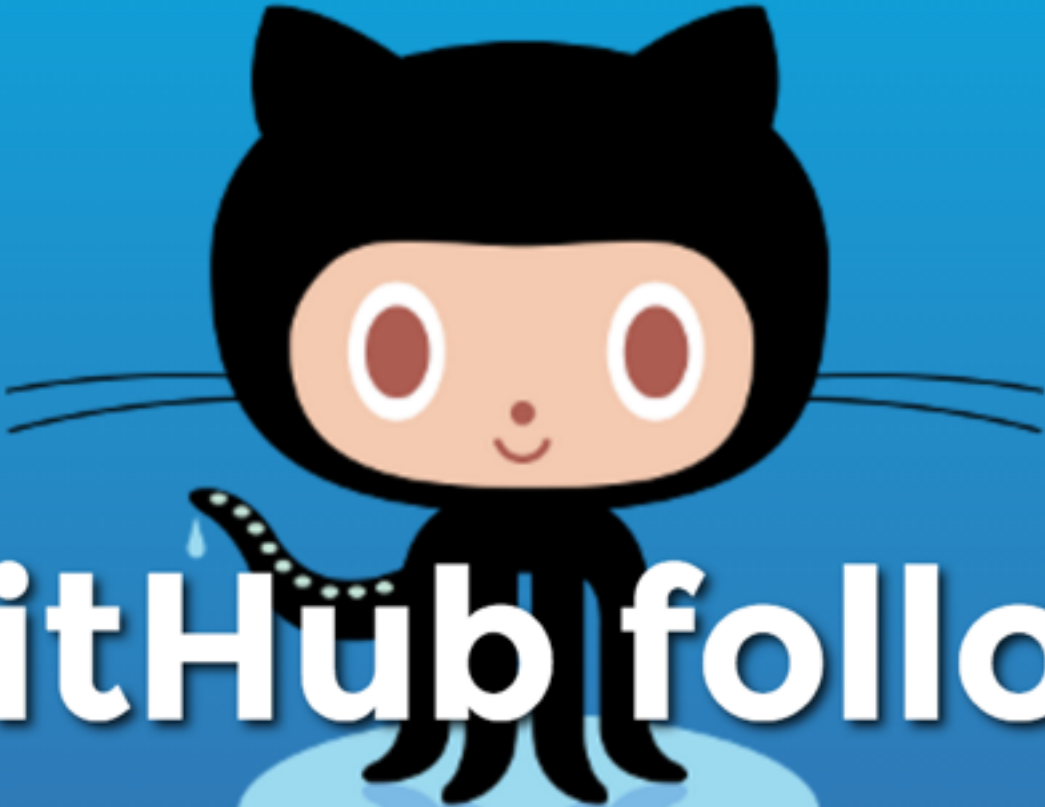
GitHub Followers

aab@administra...

githubfollowers.com

☆

Fork me on GitHub



Get GitHub followers.

Increase your follower count with the click of a button.

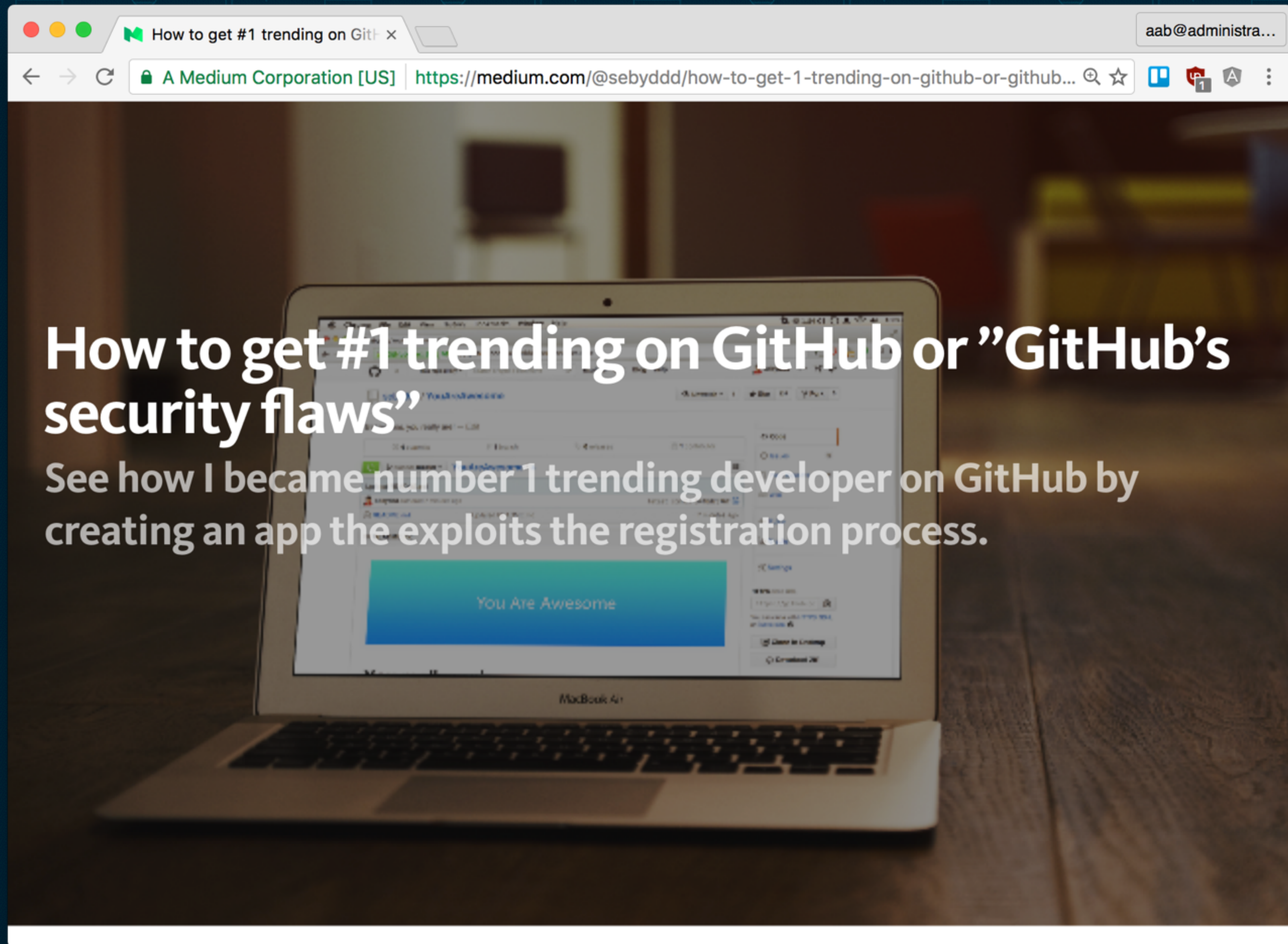
Login with GitHub

How it works

Ever wondered **how to get more GitHub followers**? **GitHub Followers** uses your OAuth2 token from GitHub's API to follow other users registered with our website. In return, we use those tokens to have them follow you back.

How to get #1 trending on GitHub or "GitHub's security flaws"

See how I became number 1 trending developer on GitHub by creating an app the exploits the registration process.



~~Server backdoor~~

Django version 2!

Python

v17

Downloads

9000+

No

Virus

Totally

Safe

Installation

```
sudo pip install Django
```






MISSION ACCOMPLISHED

HAHAHA

HAHANO



PRODUCTION



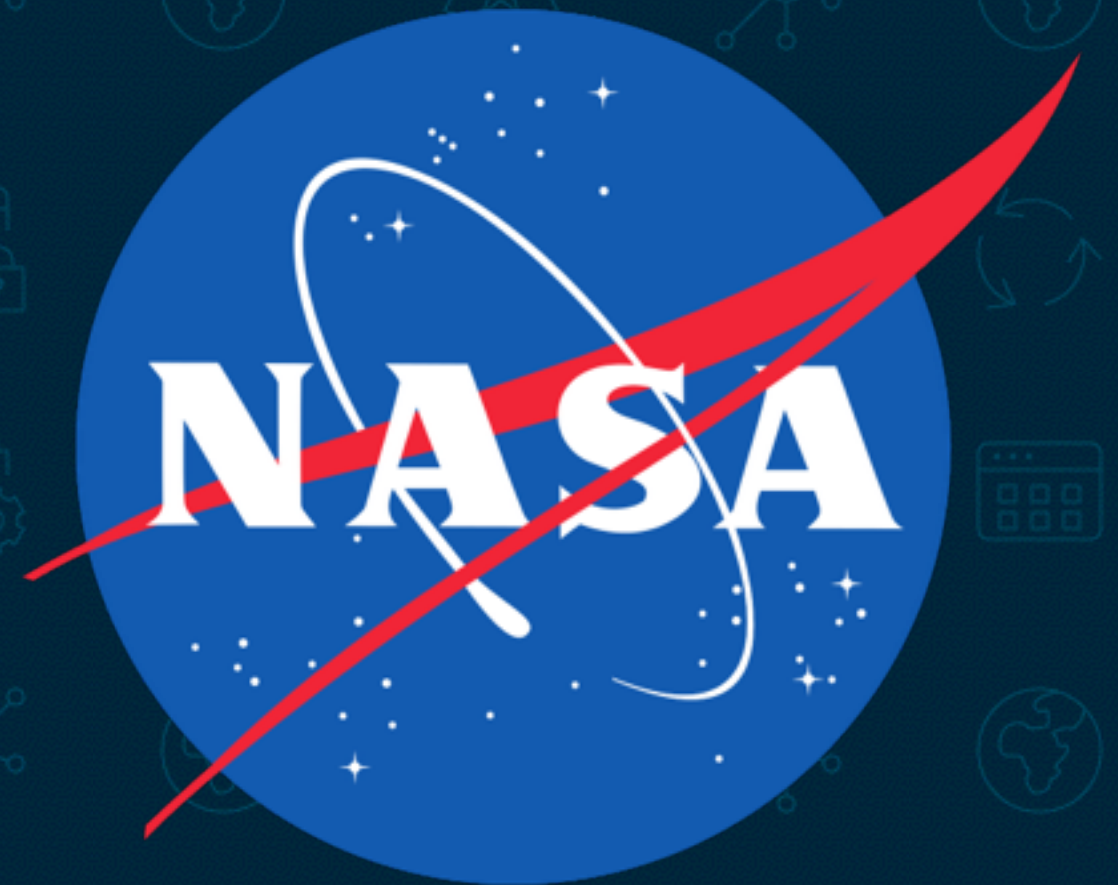
QUALITY ASSURANCE



LOCAL DEVELOPMENT



- 40,000 pages of specifications
- 420,000 lines of code
- 17 errors in last 11 versions



THE ONLY
BUG FREE
CODE IS NO
CODE

3. aaronbassett@Aarons-MacBook-Pro: ~/Projects/pycon (zsh)

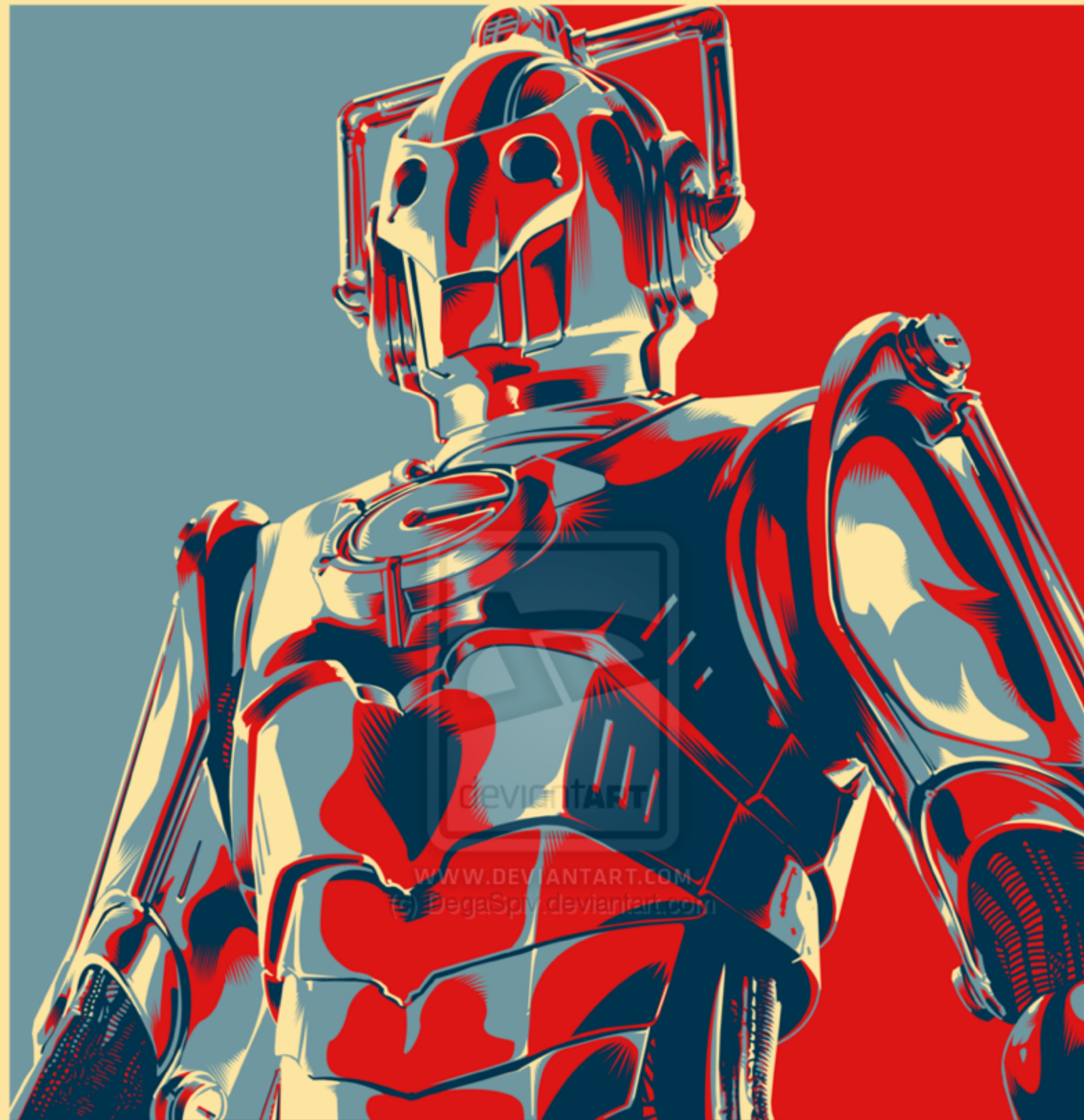
(pycon) Projects/pycon master ●

tree

```
.
├── requirements
│   ├── base.txt
│   ├── local.txt
│   ├── production.txt
│   └── staging.txt
```

1 directory, 4 files

(pycon) Projects/pycon master ●



DELETE

2. aaronbassett@Aarons-MacBook-Pro: ~/.virtualenvs/tmp-c768e8c07ab8dc5f (zsh)

(tmp-c768e8c07ab8dc5f) .virtualenvs/tmp-c768e8c07ab8dc5f

pip freeze

click==6.6

Flask==0.11.1

itsdangerous==0.24

Jinja2==2.8

MarkupSafe==0.23

Werkzeug==0.11.10

(tmp-c768e8c07ab8dc5f) .virtualenvs/tmp-c768e8c07ab8dc5f



ORPHAN PACKAGES



PIP TOOLS TO THE RESUCE

3. aaronbassett@Aarons-MacBook-Pro: ~/Projects/pycon (zsh)

(pycon) Projects/pycon master ●

tree

```
.
├── requirements
│   ├── base.in
│   ├── local.in
│   ├── production.in
│   └── staging.in
```

1 directory, 4 files

(pycon) Projects/pycon master ●

3. aaronbassett@Aarons-MacBook-Pro: ~/Projects/pycon/requirements (zsh)

(pycon) ▶ pycon/requirements ▶ master ●

cat base.in

Flask

(pycon) ▶ pycon/requirements ▶ master ●

3. aaronbassett@Aarons-MacBook-Pro: ~/Projects/pycon/requirements (zsh)

```
(pycon) ▶ pycon/requirements ▶ master ●  
└─ cat production.in  
-r base.in  
(pycon) ▶ pycon/requirements ▶ master ●  
└─
```




3. aaronbassett@Aarons-MacBook-Pro: ~/Projects/pycon/requirements (zsh)

(pycon) pycon/requirements master ●

pip-compile production.in -o pinned/production.txt

```
#  
# This file is autogenerated by pip-compile  
# To update, run:  
#  
#     pip-compile --output-file pinned/production.txt production.in  
#
```

click==6.6 # via flask

Flask==0.11.1

itsdangerous==0.24 # via flask

Jinja2==2.8 # via flask

MarkupSafe==0.23 # via jinja2

Werkzeug==0.11.11 # via flask

(pycon) pycon/requirements master ●



pip-sync

"LEFT-PAD" PROBLEM

QNAP

TS-470Pro
169.254.100.100

ENTER

SELECT

STATUS

USB

LAN

KEEP IT LOCAL



devpi: PyPI server and packaging/

aab@administra...

doc.devpi.net/latest/

devpi server-4.0, client-2.6, web-3.1 documentation »

next | index

devpi: PyPI server and packaging/testing/release tool

Note

Please note that devpi-server 4.0.0 is a bug fix/compatibility release as it only changes project name normalization compared to 3.1.x. The internal use of the normalization requires an export/import cycle, which is the reason for the major version increase. There are no other big changes and so everyone who used devpi-server 3.x.y should be fine just using 4.0.0. It's also fine to export from 2.6.x and import with 4.0.0.

See [devpi-server-4.0: fixing the pip-8.1.2 problem / PEP503 compliance](#) for details.

The MIT-licensed devpi system features a powerful PyPI-compatible server and a complimentary command line tool to drive packaging, testing and release activities with Python. Main features and usage scenarios:

Links and contact

[issue tracker](#), [mailing list](#)

[tutorials and documentation](#)

[repo of server/web/client](#)

#devpi on freenode




Table Of Contents

devpi: PyPI server and packaging/testing/release tool

- **Tutorials and Documentation**

Next topic

Quickstart: running a pypi mirror on your laptop

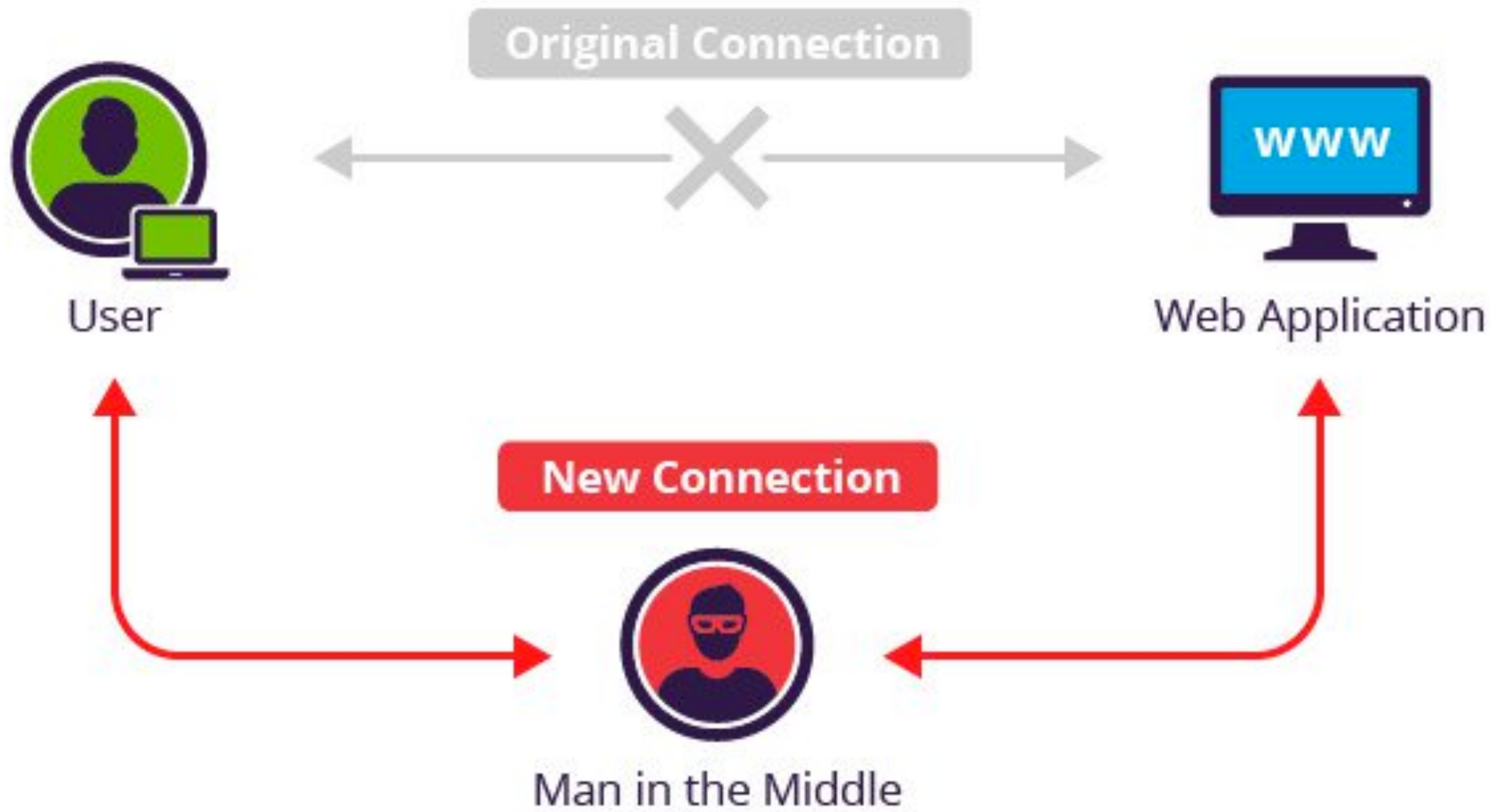
Quick search

Go









Djang0/README.md at master

aab@administra...

← → ↻

GitHub, Inc. [US]

https://github.com/aaronbassett/Djang0/blob/master/README.md

☆

aaronbassett / Djang0

Private

Unwatch 1

★ Star 216

Fork 0

<> Code

! Issues 0

🔗 Pull requests 0

📖 Wiki

⚡ Pulse

📊 Graphs


⚙ Settings

Branch: master ▾

Djang0 / README.md

Find file

Copy path

 aaronbassett Update README.md

b683d5b a minute ago

1 contributor

12 lines (8 sloc) | 314 Bytes

Raw

Blame

History

~~Server backdoor~~

Django version 2!

Python v17

Downloads 9000+

No Virus

Totally Safe

Installation

```
sudo pip install Djang0
```




PIPHASH

| File | Type | Py V |
|---|--------------|------|
| hashin-0.6.0-py3-none-any.whl (md5) | Python Wheel | py3 |
| hashin-0.6.0.tar.gz (md5) | Source | |

Author: Peter Bengtsson

NO.




hashin 0.6.0 : Python Package

aab@administra...

Python Software Foundation [US]

https://pypi.python.org/pypi/hashin

☆

python™

search

» Package Index > hashin > 0.6.0

PACKAGE INDEX >>

Browse packages

Package submission

List trove classifiers

RSS (latest 40 updates)

RSS (newest 40 packages)

PyPI Tutorial

PyPI Security

PyPI Support

PyPI Bug Reports

PyPI Discussion

PyPI Developer Info

ABOUT >>

NEWS >>

DOCUMENTATION >>

DOWNLOAD >>

COMMUNITY >>

FOUNDATION >>

CORE DEVELOPMENT >>

hashin 0.6.0

Edits your requirements.txt by hashing them in

Downloads ↓

build passing pypi package 0.6.0

Helps you write your requirements.txt with hashes so you can install with pip install --require-hashes -r ...

If you want to add a package or edit the version of one you're currently using you have to do the following steps:

1. Go to pypi for that package

2. Download the .tgz file

3. Possibly download the .whl file

4. Run pip hash downloadedpackage-1.2.3.tgz

5. Run pip hash downloadedpackage-1.2.3.whl

6. Edit requirements.txt

This script does all those things. Hackishly wonderfully so.

A Word of Warning!

The whole point of hashing is that you **vet the packages** that you use on your laptop and that they haven't been tampered with. Then you can confidently install them on a server.

This tool downloads from PyPI (over HTTPS) and runs pip hash on the downloaded files.

You should check that the packages that are downloaded are sane and not tampered with. The way you do that is to run hashin as normal but with the --verbose flag. When you do that it will print where it downloaded the relevant files and those files are not

Welcome aaronbassett

[Your details \(Logout\)](#)

Your packages:

[django-cyborg](#)

[django-discourse](#)

[django-disposable-email-checker](#)

[django-GNU-Terry-Pratchett](#)

[sometimes](#)

Status

hashin 0.6.0

- Inspect code before installing
- Be your own Pypi
- Use pip-compile and pip-sync
- Hash all the things

Administrate

THANK YOU

@aaronbassett
getadministrate.com