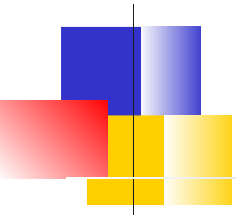


# Pull Requests

Why Use Them?

# What is a Pull Request?

# When and Why to Use PR

1. Propose a change - bug fix, improvement
2. Request help or feedback on your work
3. Request review and discussion of your work
   - do this before merging branch into master

## Unit & Integration Tests?

Case 3 should have new unit tests! And maybe integration (functional) tests.

# What Happens *after* a Pull Request?

"Interested parties" (the core dev team):

1. Review the changes
2. Test the changes
3. Discuss the value and potential impact
4. Suggest modifications

In Github Flow,

5. Approve changes for merge into master, or give reasons why not.

# References

About Pull Requests - Github

https://help.github.com/articles/about-pull-requests/

Commenting on a Pull Request (howto & examples)

https://help.github.com/en/articles/commenting-on-a-pull-request

Pull Request Tutorial - what buttons to press. Discusses squashing commits before a pull request.

https://yangsu.github.io/pull-request-tutorial/

# Pyup

Radii has 82 closed pull requests.

40 are by pyup-bot  See `https://pyup.io`

**What** does pyup-bot **do**?

# pyup.io - Security Updates

Please look at Pyup's nicely formatted docs page.
(https://pyup.io/docs/)
Try to write documentation like this!

1. Add to repo as an online service.
   * Must grant pyup.io OAuth access to the repository
   * Can automatically update project dependencies

== or ==

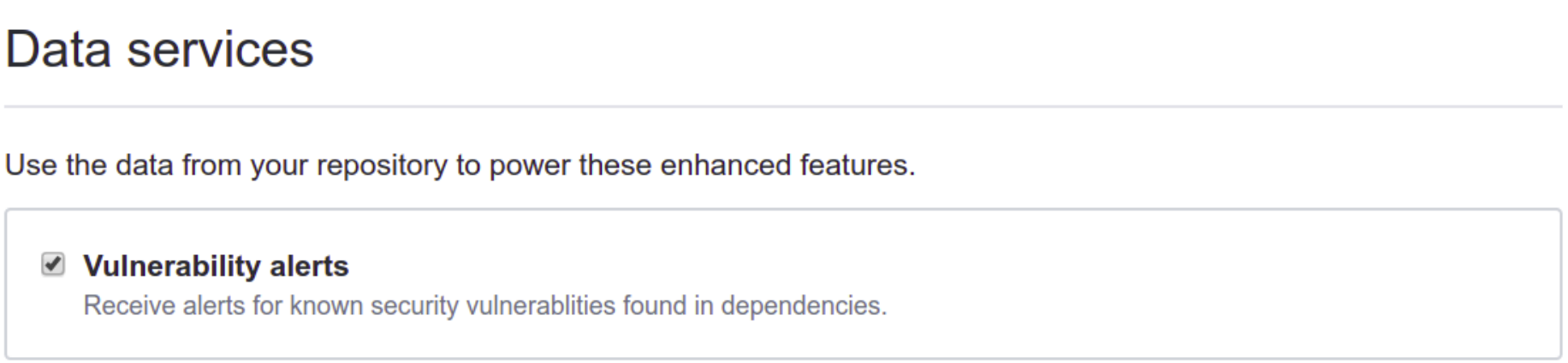
2. Run the Safety Service on CI server.

   `pip install safety` (installs several packages. use venv)

   `safety check [ -r requirements.txt ]`

# Security Alerts from Github

On Github repository settings page:




Github "vulnerabilities" based on CVE database.

## Where are Alerts?

Repository "Insights" tab -> "Alerts"

or "Dependency Graph"

# National Vulnerability Database

**`https://nvd.nist.gov`**

A global database of security problems.

Managed by U.S. NIST, but everyone can use.

Assigns CVE-# for reported & verified security problems.

**CVE-2018-15947** — Adobe Acrobat and Reader versions 2018.011.20063 and earlier, 2017.011.30102 and earlier, and 2015.006.30452 and earlier have an out-of-bounds read vulnerability. Successful exploitation could lead to information disclosure.
**Published:** October 12, 2018; 02:29:19 PM -04:00

*V3:* **5.5 MEDIUM**
*V2:* **4.3 MEDIUM**

**CVE-2018-15943** — Adobe Acrobat and Reader versions 2018.011.20063 and earlier, 2017.011.30102 and earlier, and 2015.006.30452 and earlier have an out-of-bounds read vulnerability. Successful exploitation could lead to information disclosure.
**Published:** October 12, 2018; 02:29:17 PM -04:00

*V3:* **5.5 MEDIUM**
*V2:* **4.3 MEDIUM**

# Is Django Secure?

*Search the CVE database for "Django".*

In your project `requirements.txt` put:

```
Django==2.1.0    # has a known vulnerability
```

Github will show: