

# 10 Tips Learned from Running A Python [Pyramid] Web App in Production



Michael Kennedy  
@mkennedy

# Getting the source code

The screenshot shows a GitHub repository page for 'mikekennedy/python-virtual-conf-web-tips'. The repository has 4 commits, 1 branch, 0 packages, 0 releases, 1 contributor, and is licensed under MIT. The latest commit was made 44 minutes ago. The repository contains files like .idea, design\_patterns, saferapp, static\_content, .gitignore, and LICENSE.

mikekennedy / python-virtual-conf-web-tips

Code and demos from my Python Virtual Conference 2020 talk

Manage topics

-o 4 commits    1 branch    0 packages    0 releases    1 contributor    MIT

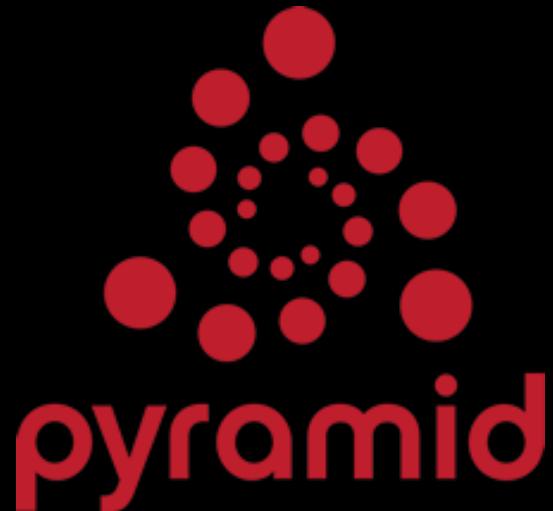
Branch: master ▾    New pull request    Create new file    Upload files    Find file    Clone or download ▾

File/Folder	Description	Time
mikekennedy Design patterns readme.	Latest commit 59bd468 44 minutes ago	
.idea	Design patterns readme.	44 minutes ago
design_patterns	Design patterns readme.	44 minutes ago
saferapp	Static files tip.	1 hour ago
static_content	Static files tip.	1 hour ago
.gitignore	Initial commit	2 hours ago
LICENSE	Initial commit	2 hours ago

<https://github.com/mikekennedy/python-virtual-conf-web-tips>



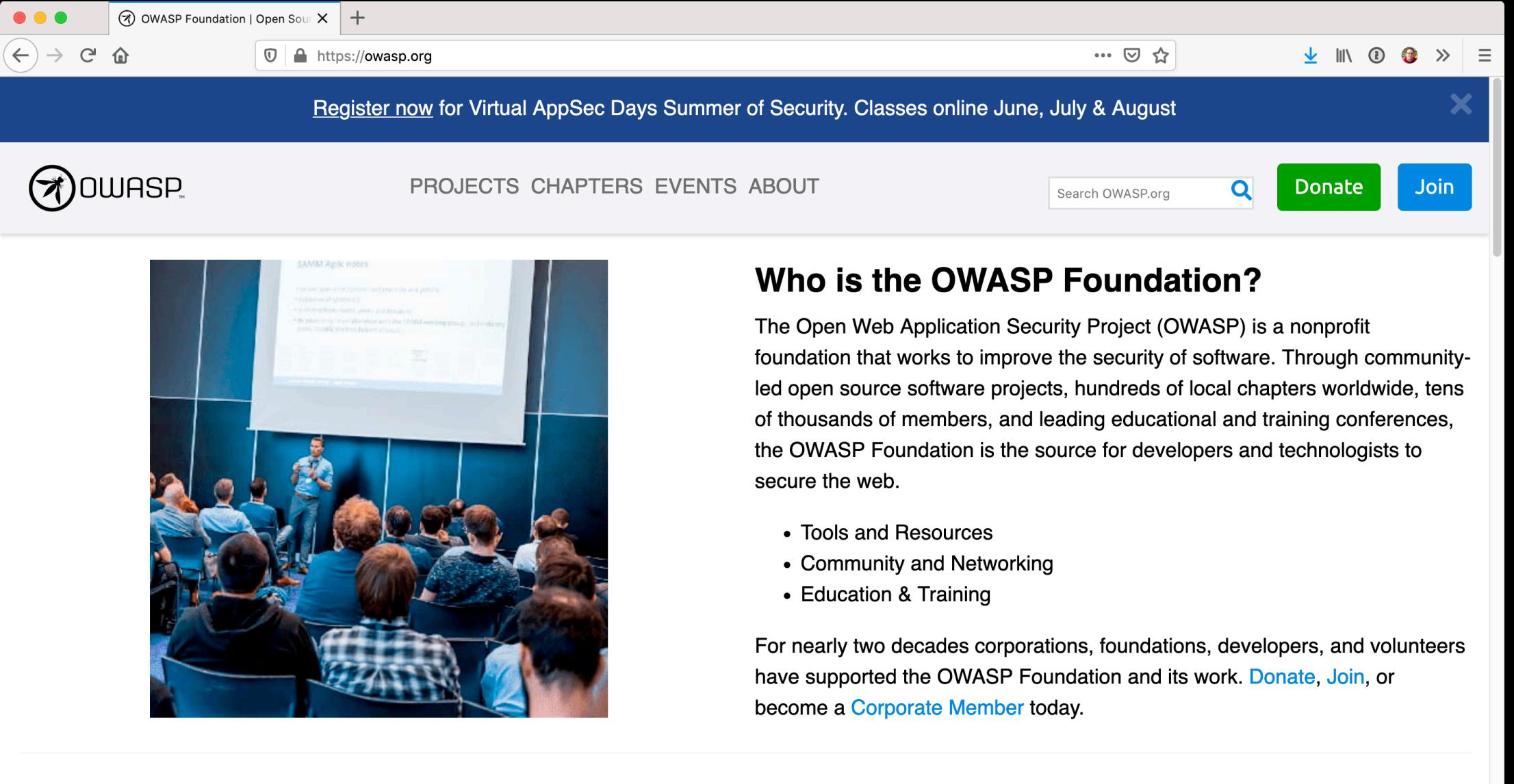
Most tips work for most frameworks



# Secure.py



# Secure.py



The screenshot shows the OWASP Foundation website. At the top, there is a banner with the text "Register now for Virtual AppSec Days Summer of Security. Classes online June, July & August". Below the banner, the OWASP logo is on the left, followed by navigation links: PROJECTS, CHAPTERS, EVENTS, and ABOUT. To the right of these are a search bar, a green "Donate" button, and a blue "Join" button. A large image of a person giving a presentation to an audience is on the left side of the main content area. The main text on the right is titled "Who is the OWASP Foundation?" and describes the organization as a nonprofit working to improve software security through community-led projects, chapters, members, and conferences. It lists three main areas of focus: Tools and Resources, Community and Networking, and Education & Training. At the bottom, it encourages users to support the foundation by donating, joining, or becoming a corporate member.

OWASP Foundation | Open Source

https://owasp.org

Register now for Virtual AppSec Days Summer of Security. Classes online June, July & August

PROJECTS CHAPTERS EVENTS ABOUT

Search OWASP.org

Donate

Join

JAMM Agile notes

Everyone apart from the JAMM team practices and partly  
+ 10 classes of JAMM 2.0  
+ 100+ attendees, points and badges  
+ No passing on, in collaboration with the JAMM working group, and including  
points, points and badges accordingly.

## Who is the OWASP Foundation?

The Open Web Application Security Project (OWASP) is a nonprofit foundation that works to improve the security of software. Through community-led open source software projects, hundreds of local chapters worldwide, tens of thousands of members, and leading educational and training conferences, the OWASP Foundation is the source for developers and technologists to secure the web.

- Tools and Resources
- Community and Networking
- Education & Training

For nearly two decades corporations, foundations, developers, and volunteers have supported the OWASP Foundation and its work. [Donate](#), [Join](#), or become a [Corporate Member](#) today.

# Integrating secure.py

```
def set_secure_headers(handler, registry):
    import secure
    secure_headers = secure.SecureHeaders()

    def tween(request):
        response = handler(request)
        secure_headers.pyramid(response)
        return response

    return tween

def add_secure_headers(config):
    tween_name = 'saferapp.infrastructure.secure_tween.set_secure_headers'
    config.add_tween(tween_name)
```

# Static content out of nginx



# site.nginx static settings

```
{  
    # ...  
  
    location /static {  
        alias /apps/saferapp/saferapp/static/;  
        expires 365d; # <- Yes, one year!  
        # We ensure the URLs include a hash of the actual file for images, css, js, etc  
        # e.g /static/all.min.css?cache_id=f8b1d8771478c8623969b754682134fb  
    }  
}
```



# Let's Encrypt

# Steps for Let's Encrypt

```
# Update and set the dependencies
$ sudo apt-get update
$ sudo apt-get install software-properties-common
$ sudo add-apt-repository universe
$ sudo apt-get update

# Install certbot
$ sudo apt-get install certbot python3-certbot-nginx

# Detect nginx setups, find domains, create SSL setup
$ sudo certbot --nginx
```

# Design Patterns



# Design Patterns

**Pyramid:** <https://github.com/talkpython/data-driven-web-apps-with-pyramid-and-sqlalchemy>

**Flask:** <https://github.com/talkpython/data-driven-web-apps-with-flask>

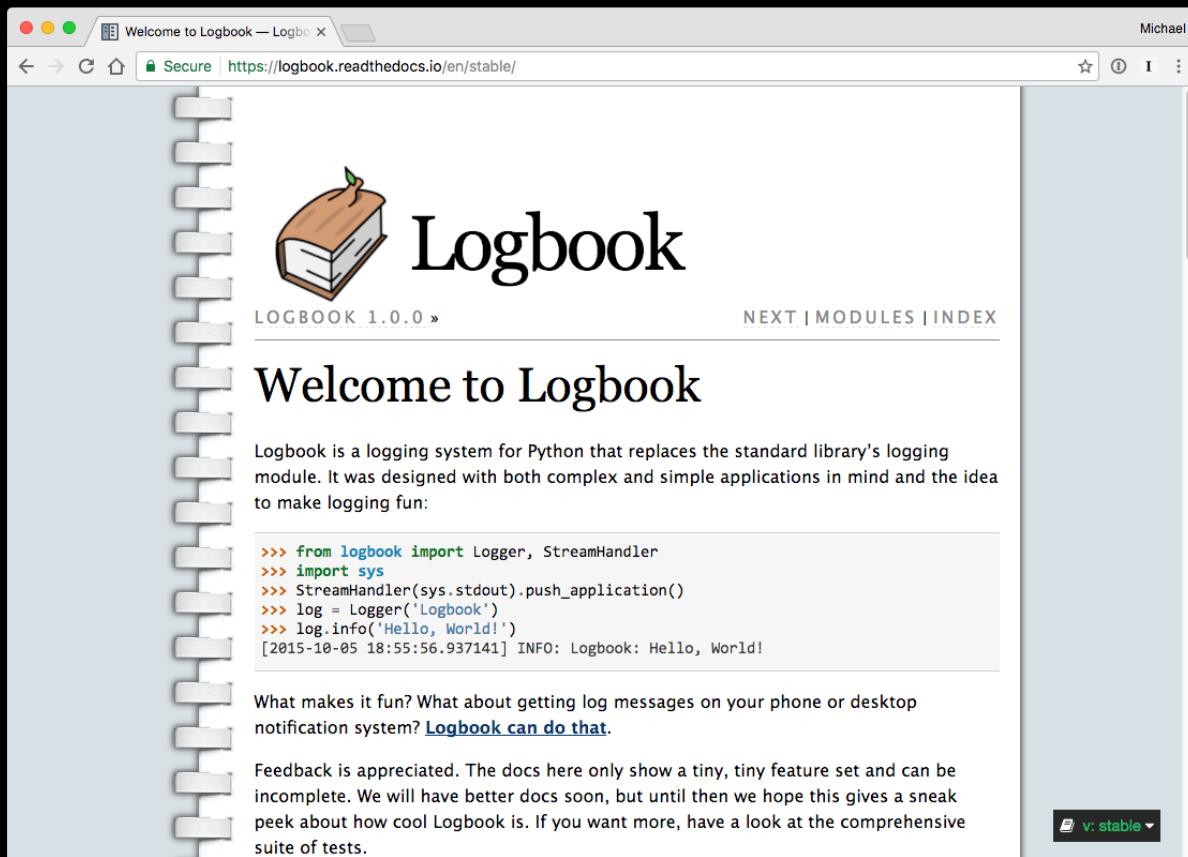
Patterns:

- ViewModels
- Services
- Structured folders (view\_module.view\_name)

# Error logging and reporting

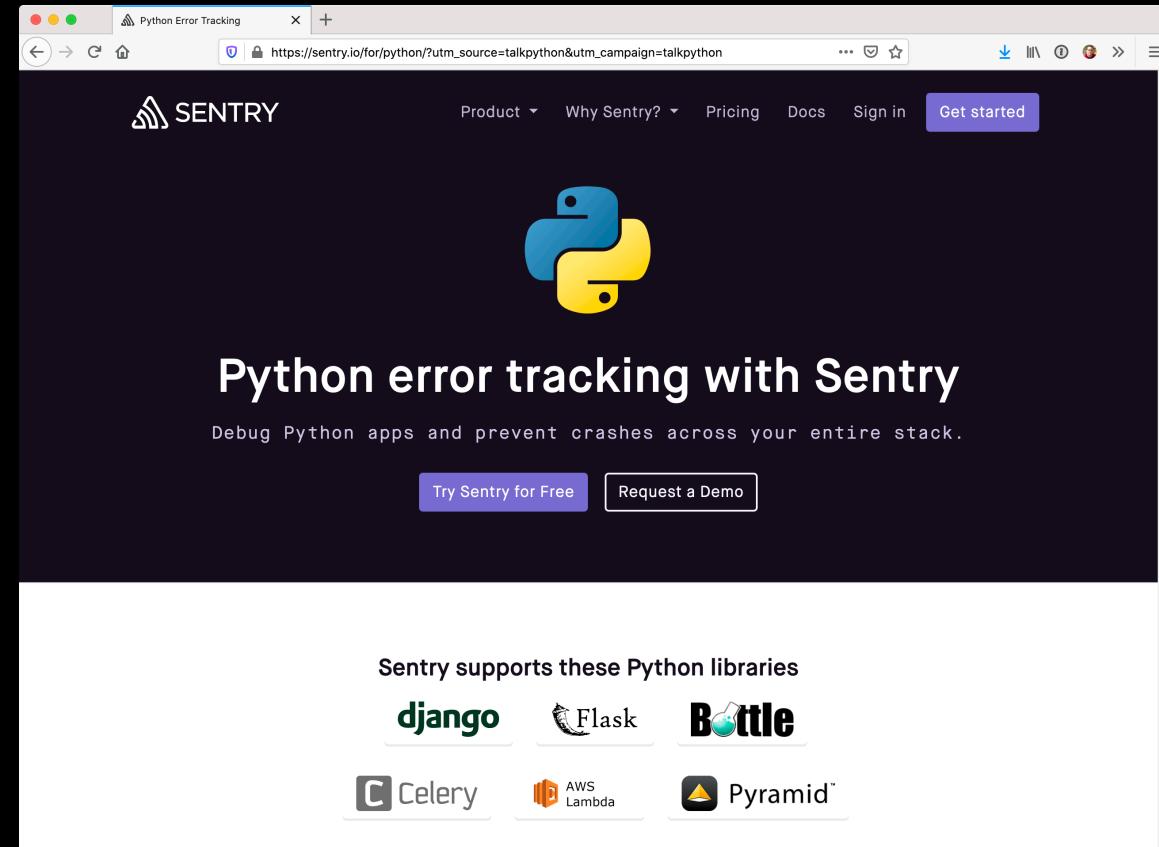


# Error logging and reporting



The screenshot shows the official documentation for Logbook, a Python logging system. The page has a clean, modern design with a white background and a light gray sidebar on the left. At the top, there's a navigation bar with links for 'Secure' and the URL 'https://logbook.readthedocs.io/en/stable/'. The main content area features a large, stylized icon of an open book with a pencil resting on it. Below the icon, the word 'Logbook' is written in a large, bold, serif font. Underneath, there are smaller links for 'LOGBOOK 1.0.0 »', 'NEXT | MODULES | INDEX'. The central part of the page is titled 'Welcome to Logbook'. It includes a brief description of what Logbook is and how it differs from the standard Python logging module. A code snippet demonstrates how to use Logbook to log 'Hello, World!' to the console. Below the code, there are two sections: one about getting log messages on a phone or desktop notification system, and another about providing feedback. At the bottom, there's a link to the full documentation at 'logbook.readthedocs.io/en/stable/'.

[logbook.readthedocs.io/en/stable/](https://logbook.readthedocs.io/en/stable/)



The screenshot shows the homepage of Sentry, a service for Python error tracking. The page has a dark blue header with the Sentry logo (a stylized 'S' with a signal icon) and the text 'Python Error Tracking'. Below the header, there's a large Python logo. The main title is 'Python error tracking with Sentry' with a subtitle 'Debug Python apps and prevent crashes across your entire stack.' There are two buttons: 'Try Sentry for Free' and 'Request a Demo'. In the center, there's a section titled 'Sentry supports these Python libraries' with icons for Django, Flask, Bottle, Celery, AWS Lambda, and Pyramid. At the bottom, there's a link to 'talkpython.fm/sentry'.

[talkpython.fm/sentry](https://talkpython.fm/sentry)

# [Not] storing secrets



# [Not] storing secrets

The image displays three GitHub repository cards against a black background, illustrating tools for detecting secrets in code repositories.

- dxa4481 / truffleHog**:
  - Used by 46
  - Watch 138

Code Issues 63 Pull requests 40 Actions Projects 0 Wiki Security 0

Searches through git repositories for high entropy strings and secrets, digging deep into commit history
- zricethezav / gitleaks**:
  - Used by 64
  - Pull requests 11 Actions

Code Issues 64 Pull requests 11 Actions

Scan git repos for secrets using regex and entropy 🔑

security security-tools git golang go scanning keys
- eth0izzle / shhgit**:
  - Sponsor

Code Issues 6 Pull requests 6 Actions Security 0 Insights

Code Issues 6 Pull requests 6 Actions Security 0 Insights

Ah shhgit! Find GitHub secrets in real time <https://shhgit.darkport.co.uk/>

github github-api security osint golang cyint

Shhgit finds secrets and sensitive files across GitHub code and Gists committed in near real-time by listening to the GitHub Events API.

# [Not] storing secrets

First of all, security is about layers...

1. Store them in environment variables (or non-committed files).
2. Encrypt the secrets (store the key elsewhere, use **cryptography.fernet**)
3. Use key vaults (e.g. <https://azure.microsoft.com/en-us/services/key-vault/>)

Calvin Hendryx-Parker has some good tips on using 1Password automation:

<https://sixfeetup.com/blog/managing-secrets-and-your-environment-with-1password>

direnv can make this more automatic:

<https://direnv.net/>

7

# 80/20 testing with sitemaps



# 80/20 testing with sitemaps

```
class SiteMapTests(TestCase):
    app = None

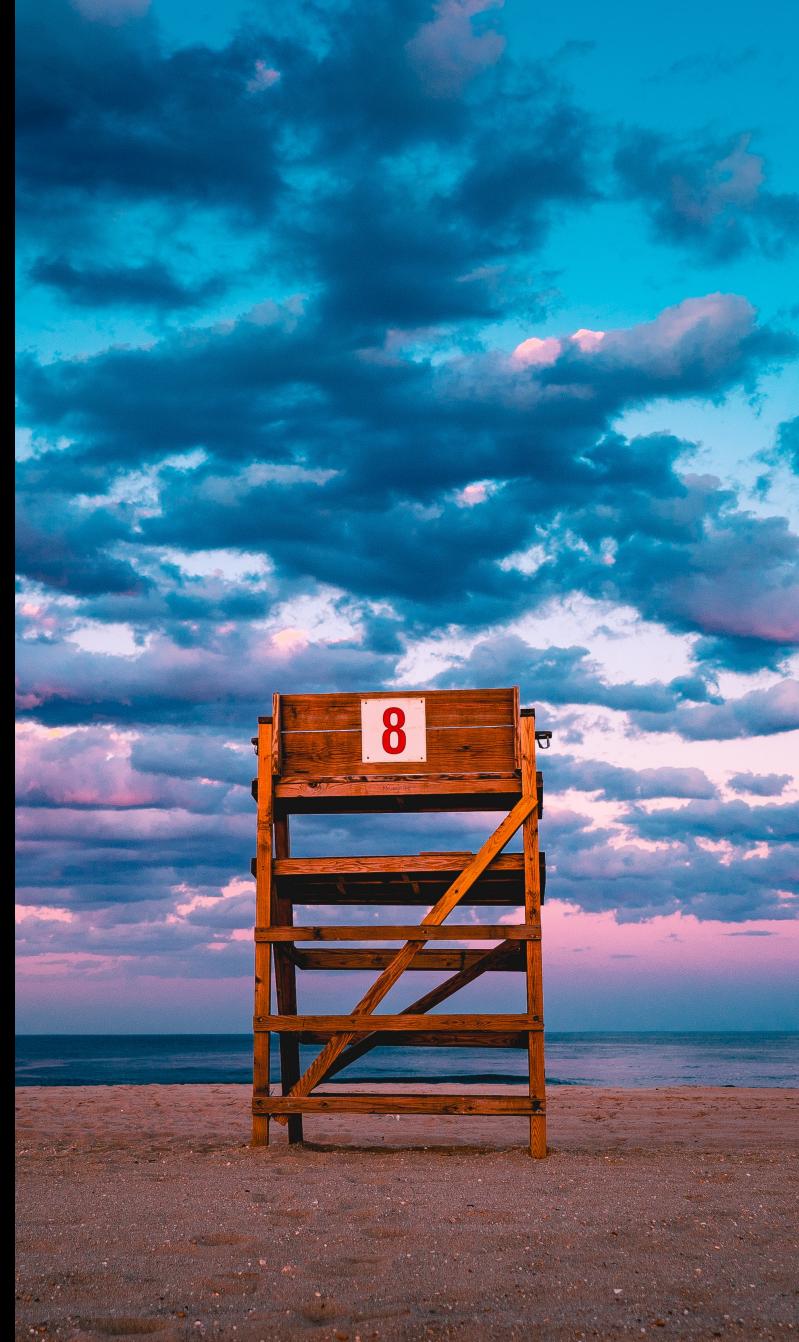
    def setUp(self):
        self.app = self.get_or_create_web_app()

    def get_sitemap_text(self):
        res = self.app.get("/sitemap.xml")
        return res.text.replace(
            'xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"', '')

    def test_site_mapped_urls(self):
        text = self.get_sitemap_text()
        x = xml.etree.ElementTree.fromstring(text)
        urls = [
            href.text.strip().replace('https://yourdomain.com', '')
            for href in list(x.findall('url/loc'))
        ]
        print(f'Testing {len(urls)} urls from sitemap...', flush=True)

        for url in urls:
            print('Testing url at ' + url)
            self.app.get(url, status=200)
```

# Site speed



# Site speed

PageSpeed Insights

https://developers.google.com/speed/pagespeed/insights/?url=https%3A%2F%2Ftraining.talkpython.fm&tab=d

PageSpeed Insights HOME DOCS

https://training.talkpython.fm/ ANALYZE

MOBILE DESKTOP

97

https://training.talkpython.fm/

0-49 50-89 90-100 ⓘ

**Field Data** — Over the last 30 days, field data shows that this page **passes** the Core Web Vitals assessment.

First Contentful Paint (FCP) 1.2 s First Input Delay (FID) 2 ms

68% 30% 2% 100%

Get notified about new courses and more.

https://developers.google.com/speed/pagespeed/insights/



# Activate a venv at server login

# Activate a venv at server login

```
● ● ● mkennedy — root@training4: ~ — ~ — ssh root@prod.training.talkpython.fm — 73x24
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-37-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

System information as of Thu Jun 18 18:49:00 UTC 2020

System load:          0.65
Usage of /:           4.8% of 77.36GB
Memory usage:         8%
Swap usage:           0%
Processes:            137
Users logged in:     0
IPv4 address for eth0: [REDACTED]
IPv4 address for eth0: [REDACTED]
IPv6 address for eth0: [REDACTED]

0 updates can be installed immediately.
0 of these updates are security updates.

Last login: Thu Jun 18 18:48:44 2020 from [REDACTED]
(virtualenv) [training-prod:] ~
```

# Dependencies



# Dependencies

No big deal: `pip install pyramid` gets me the latest

Yes, unless it's already installed: `Requirement already satisfied`

No big deal: `pip install pyramid --upgrade` gets me the latest

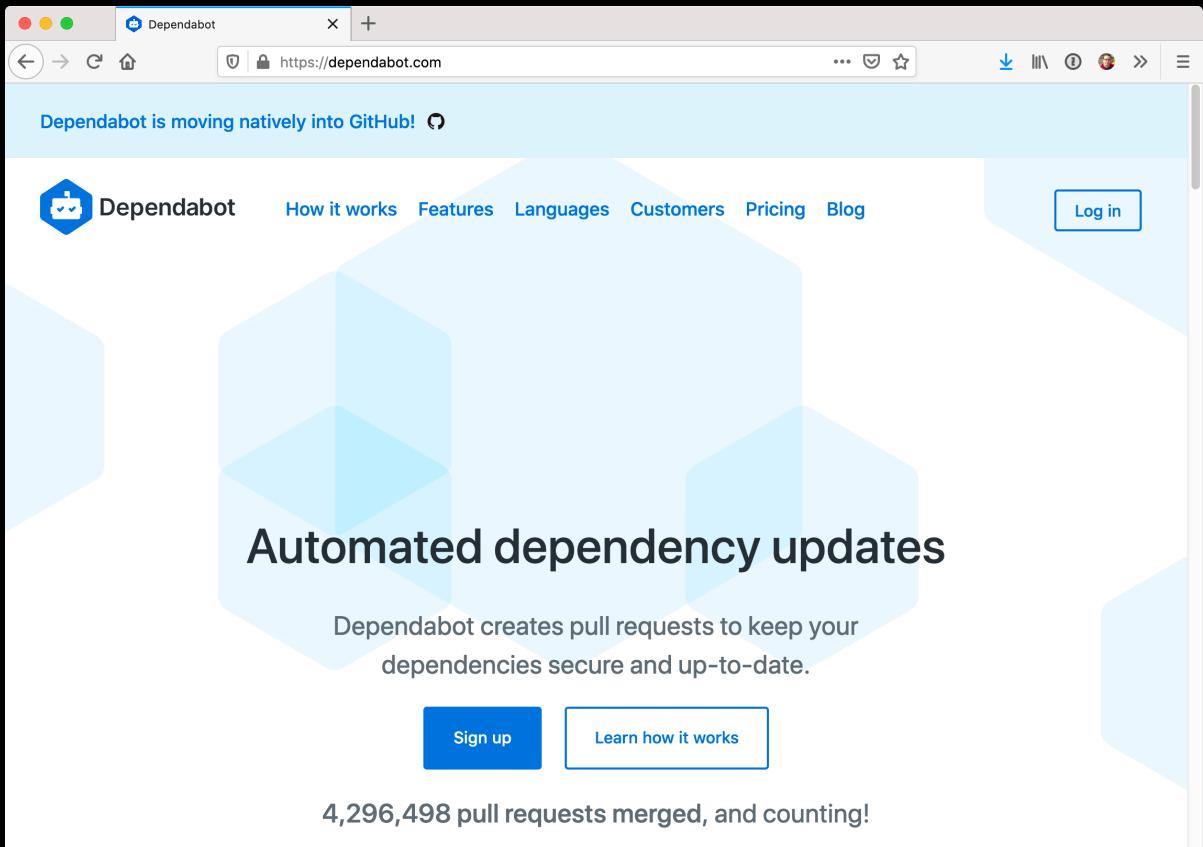
Yes, it's updated! But what about `venusian`, `WebOb`, etc.?

What about security issues that need attention right away?

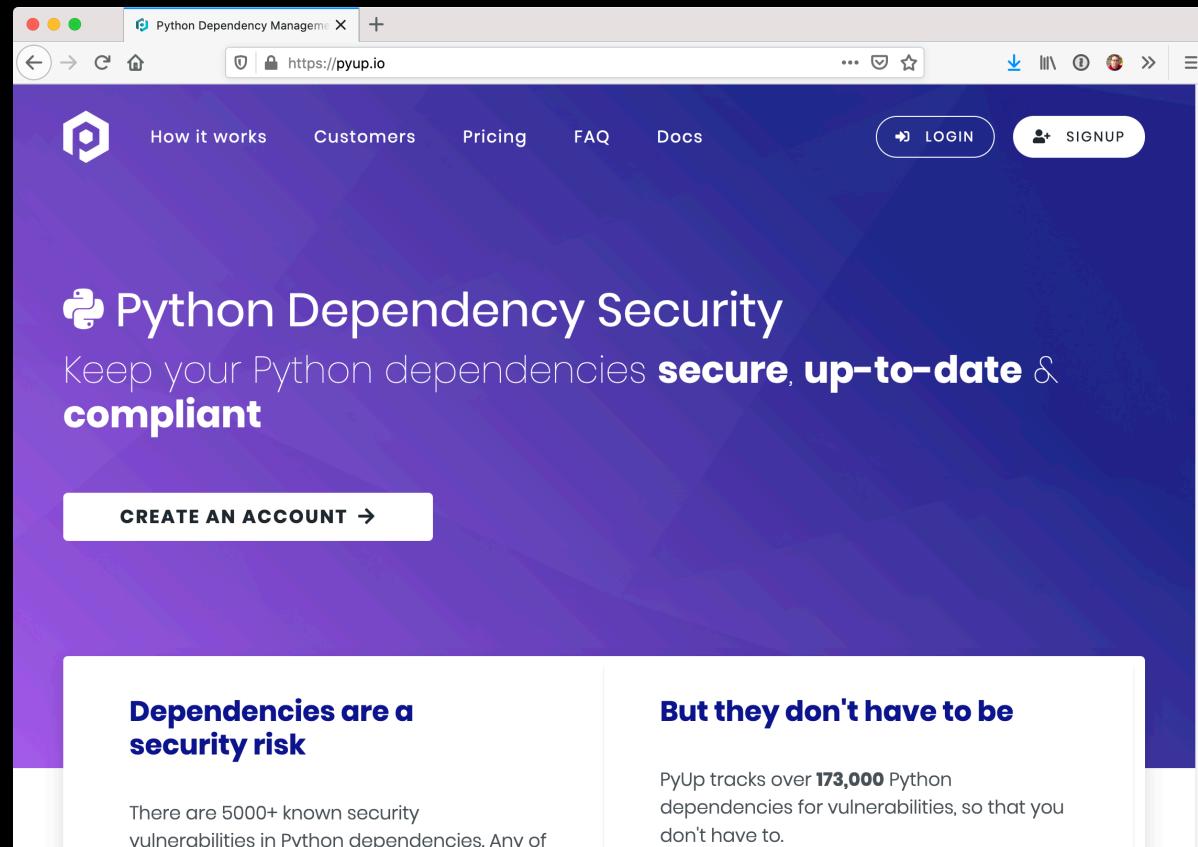
No big deal: If you pin your version, GitHub will give security updates

Yes, but what about run of the mill updates? You'll be stuck in the past!

# Dependencies



The screenshot shows the Dependabot website at https://dependabot.com. The page features a large blue hexagonal graphic with the text "Automated dependency updates". Below it, a subtext states: "Dependabot creates pull requests to keep your dependencies secure and up-to-date." At the bottom, there are two buttons: "Sign up" and "Learn how it works". A banner at the top left says "Dependabot is moving natively into GitHub! 🎉". The navigation bar includes links for How it works, Features, Languages, Customers, Pricing, and Blog, along with a "Log in" button.

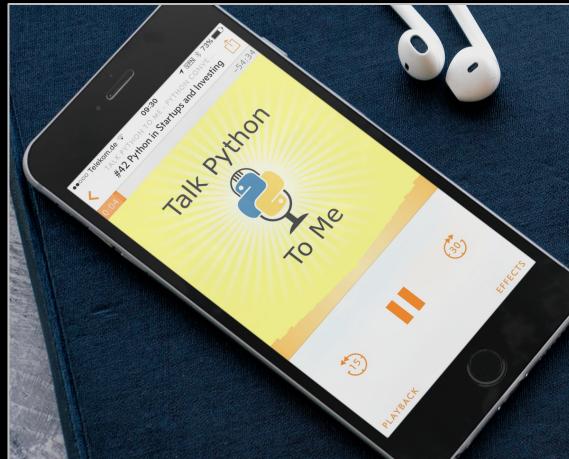


The screenshot shows the PyUp website at https://pyup.io. The main heading is "Python Dependency Security". It emphasizes keeping dependencies "secure, up-to-date & compliant". A "CREATE AN ACCOUNT →" button is visible. Below, a section titled "Dependencies are a security risk" discusses the tracked vulnerabilities. Another section, "But they don't have to be", highlights the service's tracking of over 173,000 Python dependencies. The PyUp logo, a stylized 'P' icon, is in the top left. The navigation bar includes links for How it works, Customers, Pricing, FAQ, and Docs, along with "LOGIN" and "SIGNUP" buttons.

# Thank you and where to get more



@mkennedy



Talk Python To Me Podcast



Python Bytes Podcast



Talk Python Training

<https://github.com/mikeckennedy/python-virtual-conf-web-tips>