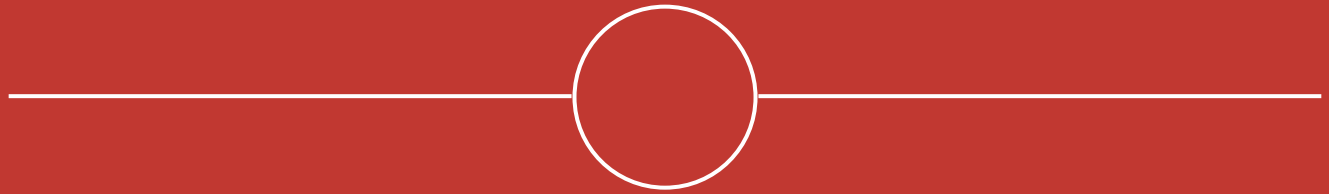
mozilla



Test Driven Security

In Continuous Integration

Julien Vehent - @jvehent - Enigma 2017



“

I HAVE FOUND SOURCE CODE DISCLOSURE ON WWW.MOZILLA.ORG!!!

RIGHT CLICK VIEW SOURCE AND YOU CAN STEAL ALL SOURCE CODE OF THE SERVER!!!

THIS IS A MAJOR VULNERABILITY!!!

Can I have a bounty now?

“

“Dear security researcher, we appreciate your participation to Mozilla’s bug bounty program, however this is not a vulnerability but simply a **feature of the web**, and thus not available for a bounty reward.”

“WHY DON’T YOU PAY A BOUNTY??? IT’S A MAJOR VULNERABILITY! I HAVE HACKED THE NSA WITH IT. YOU MUST PAY ME A BOUNTY!

Or maybe a t-shirt. Can I have a tshirt?

Tadj Youssouf ▾ (Reporter) (New to Bugzilla)

Tag Reply -

Description • a month ago

User Agent: Mozilla/5.0 (Windows NT 6.3; WOW64; rv:50.0) Gecko/20100101 Firefox/50.0
Build ID: 20161208153507

Steps to reproduce:

Hello I found a reflected xss affecting hg.mozilla.org
Just visit the link to reproduce

https://hg.mozilla.org/integration/fx-team/pushloghtml?user="><script>alert(/XSS/)</script>

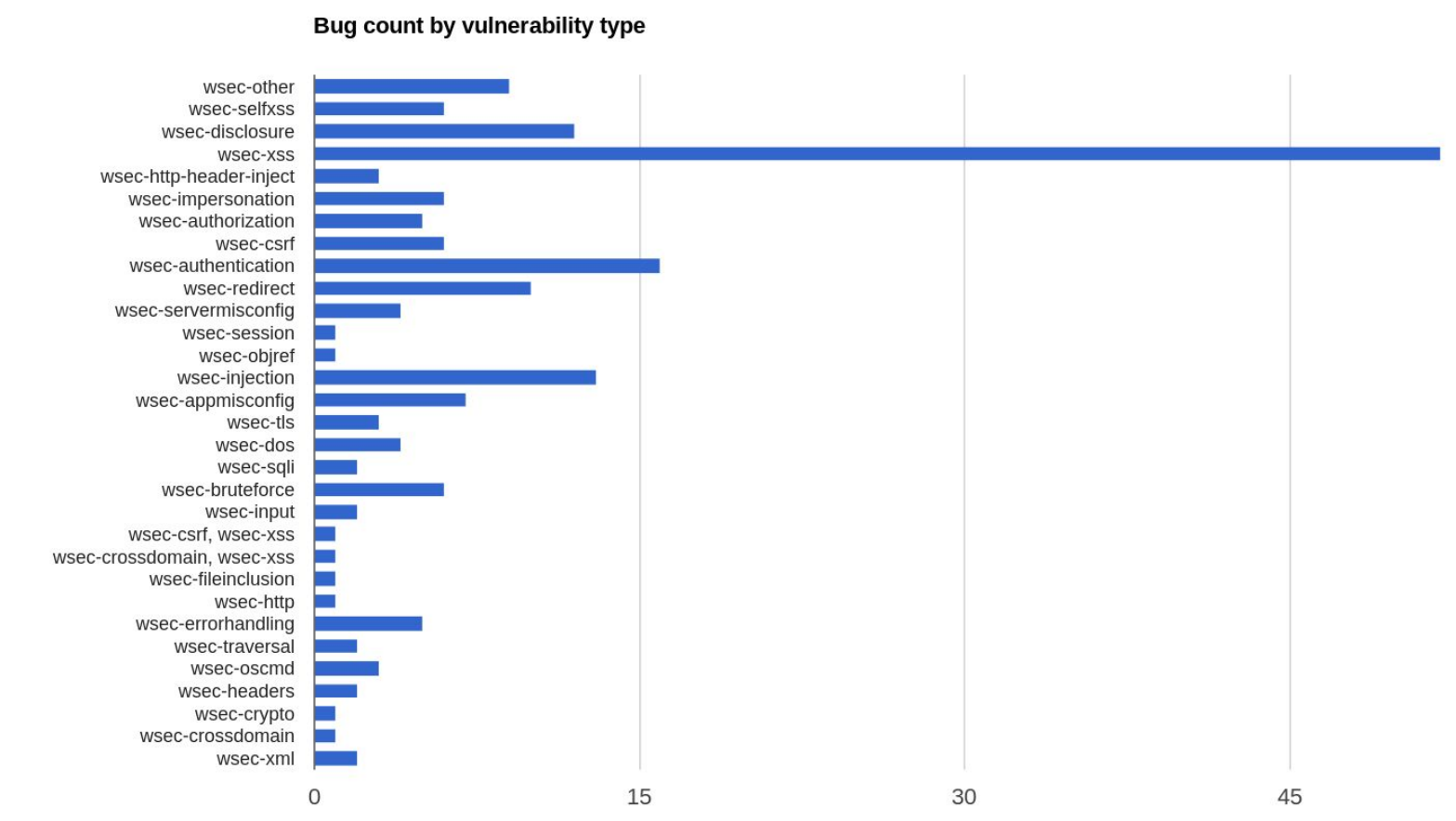
https://hg.mozilla.org/integration/fx-team/pushloghtml?user="> <iframe src="https://www.openbugbounty.org/"></iframe>

Actual results:

Reflected XSS with Iframe injection

Web vulns on Mozilla sites (2016)

mozilla



Bug Bounty payments

- Average is $2,400
- Paid only on high and critical issues (~25%)
- Bounty program cost is coming close to a a full time employee.



Can we increase
applications security
by finding these issues

BEFORE

they reach our
production systems?

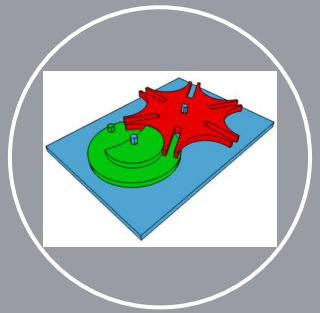
DevOps


OPS WHO ARE DEVS
WHO LIKE DEVS TO BE OPS
WHO DO OPS LIKE THEY'RE DEVS
WHO DO DEV LIKE THEY'RE OPS
always should be
SOMEONE YOU REALLY LOVE
Mountain View ✪ June 25th 2010
devopsdays.org
OPS WHO ARE DEVS
WHO LIKE DEVS TO BE OPS
WHO DO OPS LIKE THEY'RE DEVS
WHO DO DEV LIKE THEY'RE OPS
always should be
SOMEONE YOU REALLY LOVE
Hamburg ✪ October 15-16
devopsdays.org

DevOps is the process on continuously improving software products



Rapid release cycles

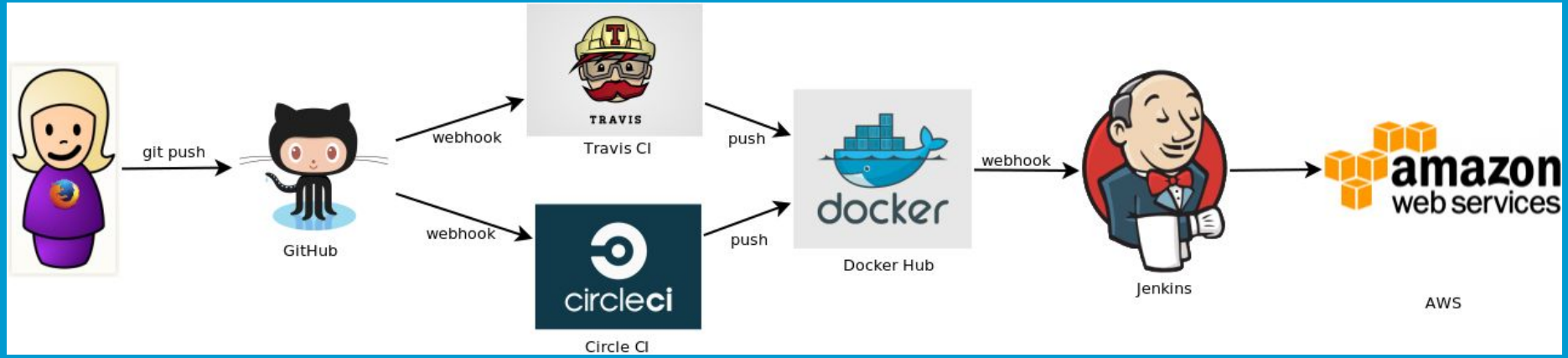


Global automation of integration & delivery pipelines



Close collaboration between teams

A DevOps pipeline

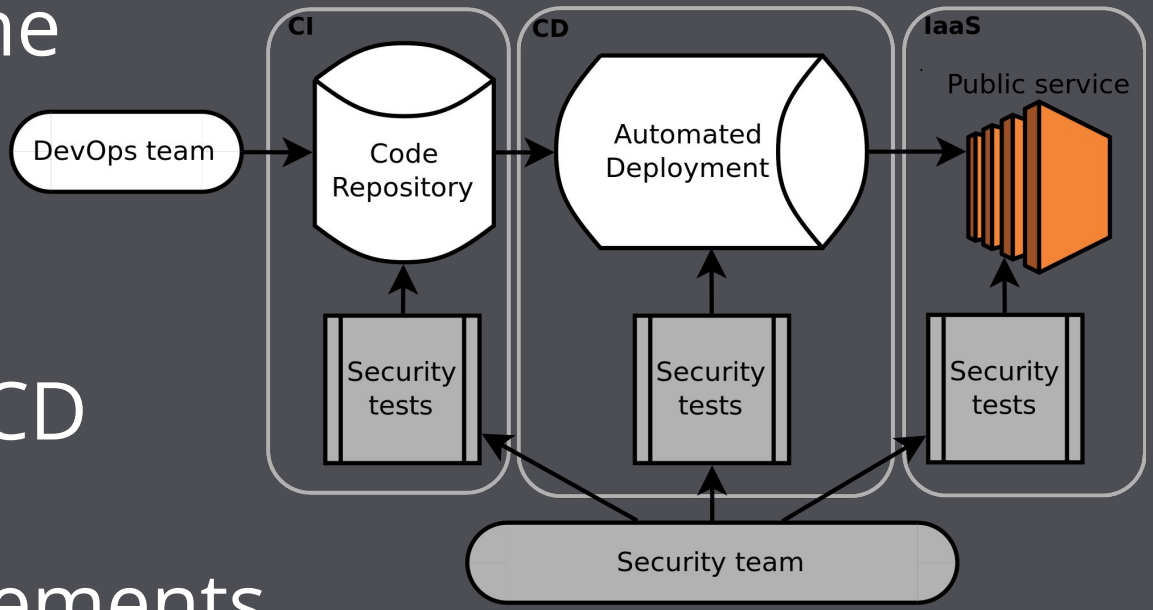


Can we integrate security testing directly into DevOps pipelines?

To allow Devs & Ops to detect issues as code travels through the pipeline.

Test Driven Security

1. Define a baseline
2. Write tests
3. Run tests in CI/CD
4. Socialize requirements
5. Require tests to pass to deploy prod



1. Defining a Security baseline

mozilla

Test Scores

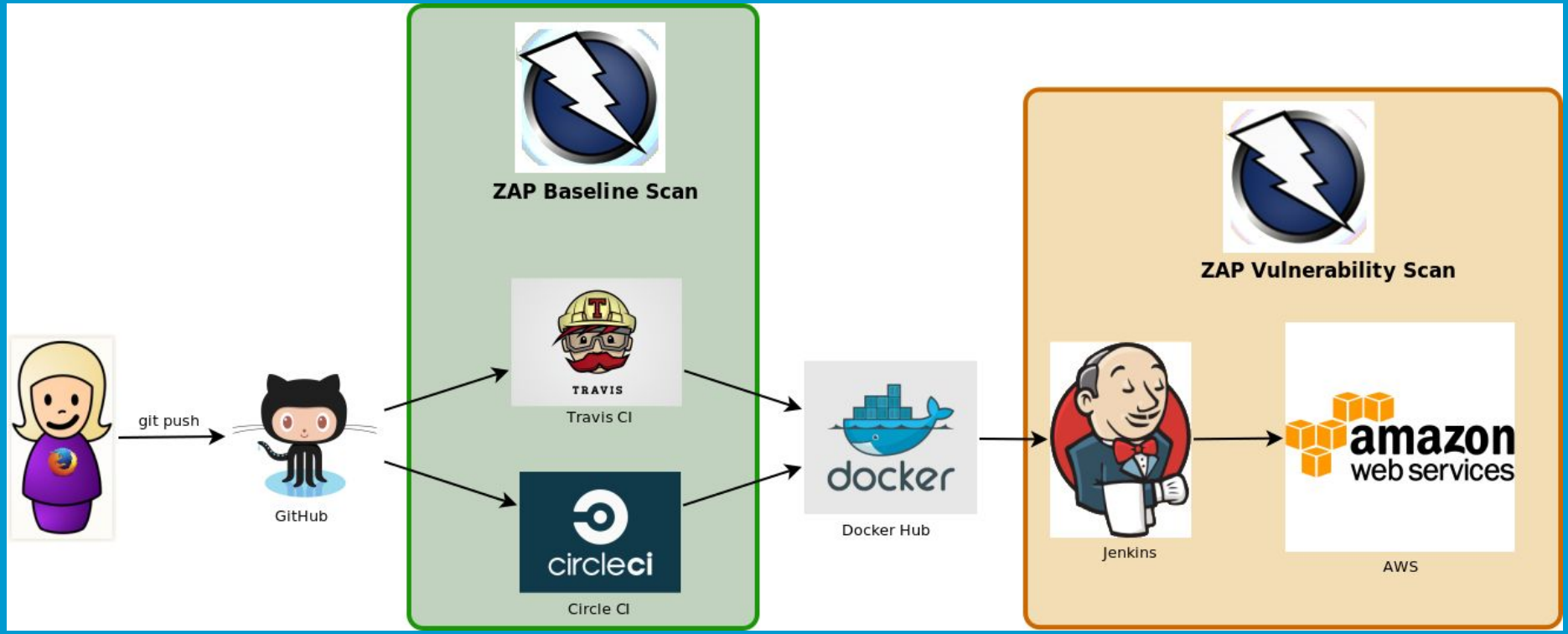
Test	Pass	Score	Explanation
Content Security Policy	✔	0	Content Security Policy (CSP) implemented with unsafe sources inside `style-src`. This includes `'unsafe-inline'`, `data:` or overly broad sources such as https:.
Contribute.json	✔	0	Contribute.json implemented with the required contact information
Cookies	✔	0	All cookies use the `Secure` flag and all session cookies use the `HttpOnly` flag
Cross-origin Resource Sharing	✔	0	Content is not visible via cross-origin resource sharing (CORS) files or headers
HTTP Public Key Pinning	✔	+5	Preloaded via the HTTP Public Key Pinning (HPKP) preloading process
HTTP Strict Transport Security	✔	+5	Preloaded via the HTTP Strict Transport Security (HSTS) preloading process
Redirection	✔	0	All hosts redirected to are in the HTTP Strict Transport Security (HSTS) preload list
Referrer Policy	–	0	Referrer-Policy header not implemented (optional)
Subresource Integrity	✘	-5	Subresource Integrity (SRI) not implemented, but all external scripts are loaded over https
X-Content-Type-Options	✔	0	X-Content-Type-Options header set to `"nosniff"`
X-Frame-Options	✔	0	X-Frame-Options (XFO) header set to `SAMEORIGIN` or `DENY`
X-XSS-Protection	✔	0	X-XSS-Protection header set to `"1; mode=block"`

2. Writing tests

Custom tests are built on top of **ZAP**, NSP, eslint, Mozilla Observatory, ...

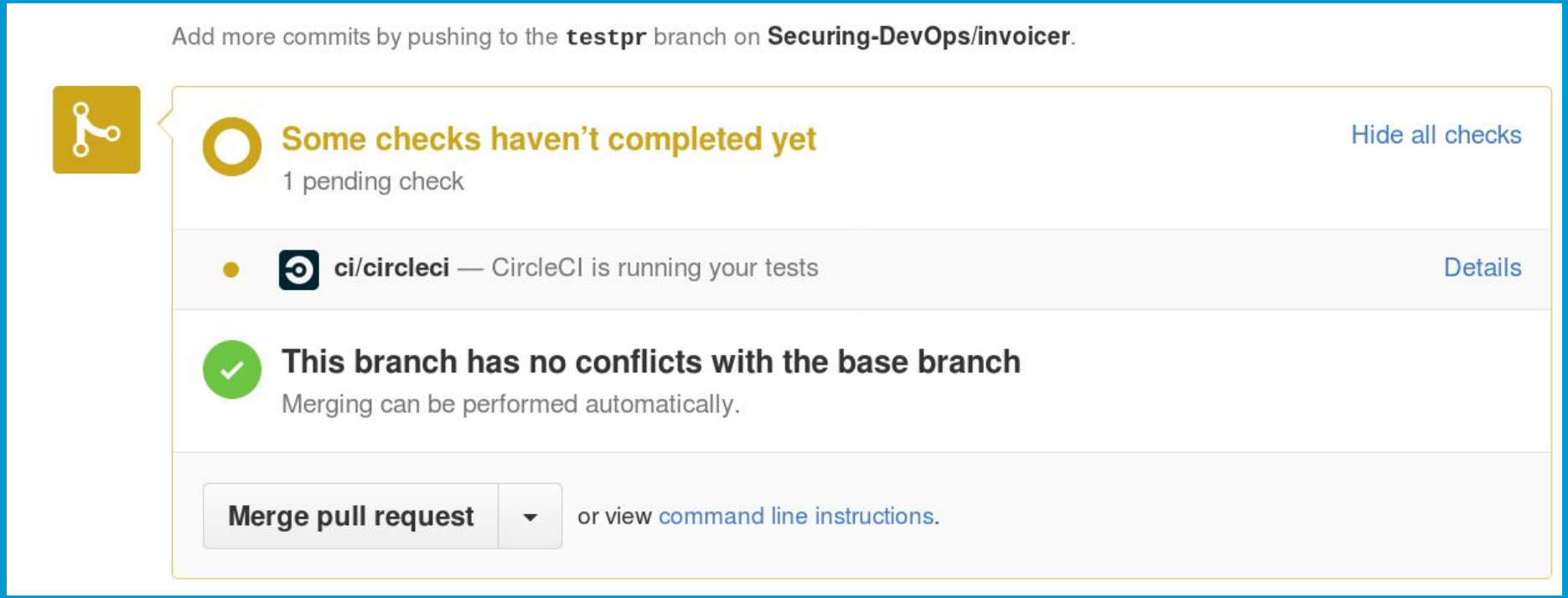


3. Running tests in CI/CD



3. Running tests in CI/CD

mozilla



3. Running tests in CI/CD

```
$ cat circle.yml
[...]
test:
  override:
    # start the application container
    - docker run mozilla/cutefox &

    # retrieve the ZAP container
    - docker pull owasp/zap2docker-weekly

    # run the ZAP scan against the application
    - >
      docker run -t owasp/zap2docker-weekly
      zap-baseline.py -t http://172.17.0.2:8080/
```

3. Running tests in CI/CD

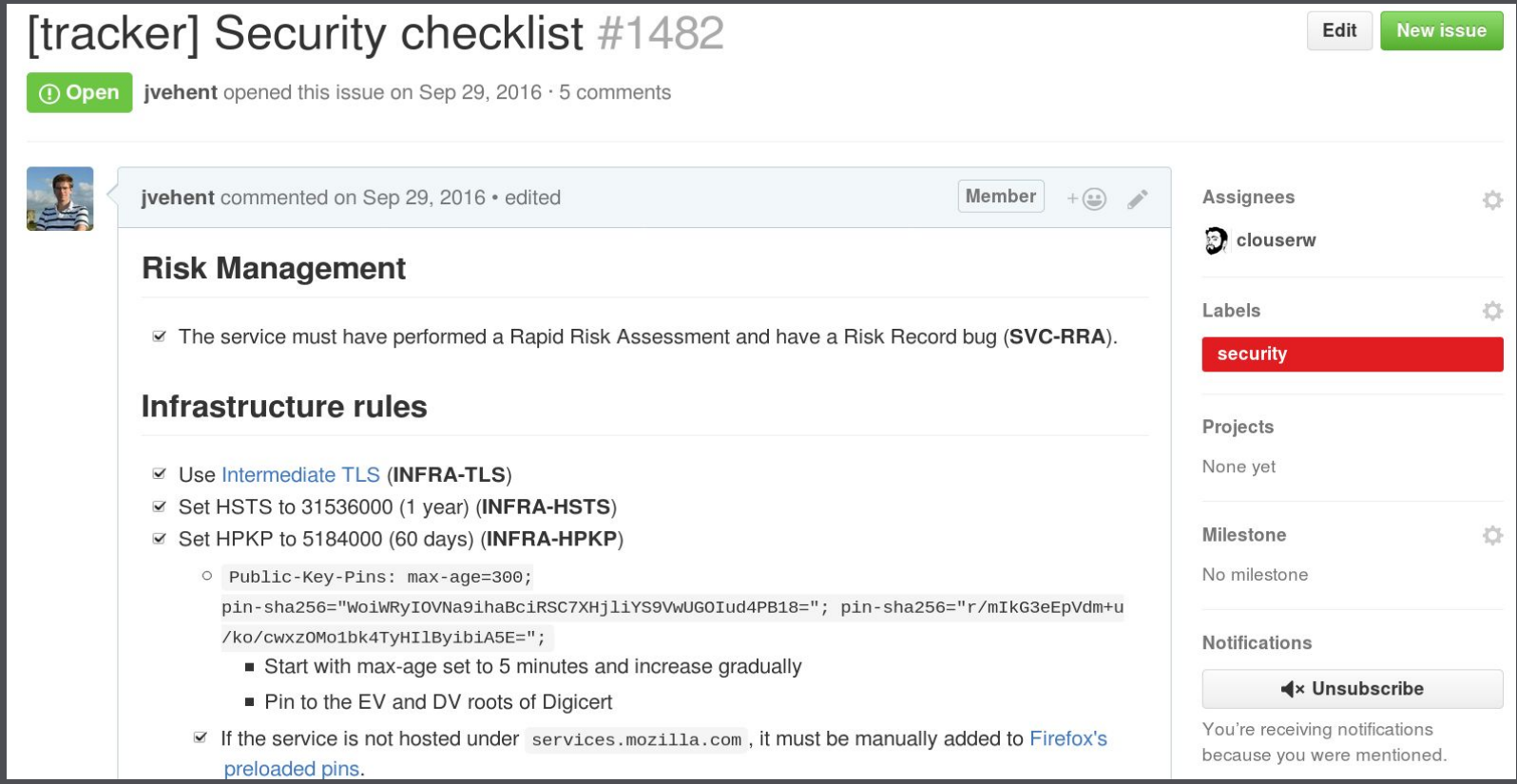
PASS: Strict-Transport-Security Header Scanner
PASS: Absence of Anti-CSRF Tokens
PASS: Secure Pages Does Not Include Mixed Content

FAIL: Web Browser XSS Protection Not Enabled x 2
http://172.17.0.2/account/preferences/
http://172.17.0.2/sitemap.xml

FAIL: Cookie Without SameSite Attribute x 1
http://172.17.0.2/en-GB/firefox/

4. Socialize requirements

mozilla



[tracker] Security checklist #1482

Edit | New issue

Open **jvehent** opened this issue on Sep 29, 2016 · 5 comments

jvehent commented on Sep 29, 2016 • edited — Member

Risk Management

- [x] The service must have performed a Rapid Risk Assessment and have a Risk Record bug (**SVC-RRA**).

Infrastructure rules

- [x] Use Intermediate TLS (**INFRA-TLS**)
- [x] Set HSTS to 31536000 (1 year) (**INFRA-HSTS**)
- [x] Set HPKP to 5184000 (60 days) (**INFRA-HPKP**)
  - `Public-Key-Pins: max-age=300; pin-sha256="WoiWRyIOVNa9ihaBciRSC7XHjliYS9VwUGOIud4PB18="; pin-sha256="r/mIkG3eEpVdm+u/ko/cwxzOMo1bk4TyHIlByibiA5E=";`
    - Start with max-age set to 5 minutes and increase gradually
    - Pin to the EV and DV roots of Digicert
- [x] If the service is not hosted under `services.mozilla.com`, it must be manually added to Firefox's preloaded pins.

Assignees: clouserw

Labels: security

Projects: None yet

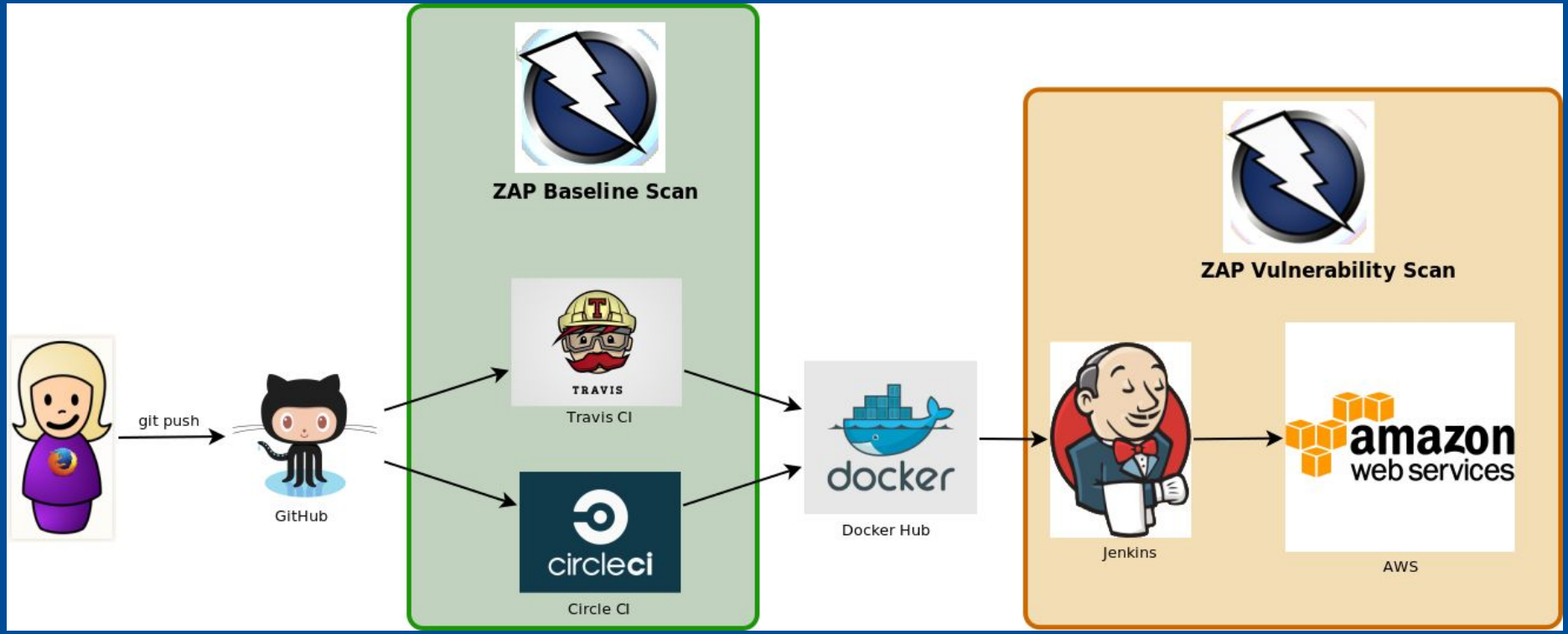
Milestone: No milestone

Notifications: Unsubscribe

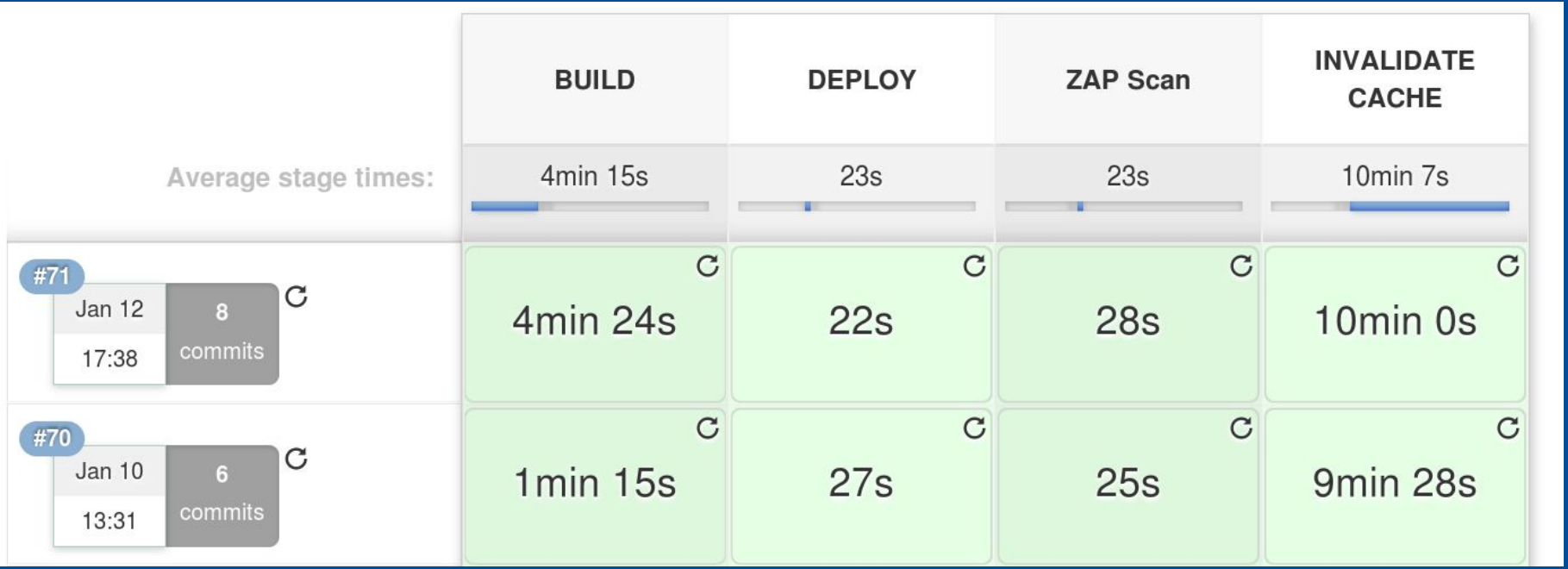
You're receiving notifications because you were mentioned.

mzl.la/2kN2dO3

5. Require tests to pass



5. Require tests to pass



	BUILD	DEPLOY	ZAP Scan	INVALIDATE CACHE
Average stage times:	4min 15s	23s	23s	10min 7s
#71 Jan 12 17:38 — 8 commits	4min 24s	22s	28s	10min 0s
#70 Jan 10 13:31 — 6 commits	1min 15s	27s	25s	9min 28s

5. Require tests to pass

- Run tests in CD (jenkins)
- Block **stage** deploy on FAIL
- Send feedback to Devs
- Don’t ever block prod!

```
$ cat pipeline.groovy

[...]

stage('ZAP Scan') {
  checkpoint 'zap'
  catchError {
    test('https://url')
  }
}
```

Does it work?

On addons.mozilla.org...

- 2010 to 2015: 106 bugs, inc. 63 XSS
- 2016: enabled baseline
- Since 2016: 8 bugs, 0 XSS

Want secure applications?

1. Define baseline security
2. Drive testing from the DevOps pipeline
3. Never deploy sub-standard code
4. Empower DevOps teams to fix issues
5. Save time to work on the complex stuff

Questions?