Security Audit of Dependencies in CI

at

flywire


about.me
📌 Cluj-Napoca, Ro
👨‍💻 IT Audit, Security Officer, Risk Management, Architecture, Engineering
from Sep 2019 - Sec Eng, Product Sec
🚴 #getoutside | /alexandrudima/
29 hr
2,946 km
29 hr
2,853 km
28 hr
London
Netherlands
Brussels
Cologne
Germany
Belgium
Prague
Czechia
Paris
Munich
Vienna
Slovakia
Budapest
Hungary
Cluj-Napoca
France
Slovenia
Zagreb
Romania
Croatia
Belgrade
Београд
Bucharest
Bosnia and
Sarajevo
Sofia
София
Montenegro
Podgorica
Подгорица
Kosovo
Bulgaria
Barcelona
Rome
Madrid
Albania
Tyrrhenian Sea
Spain
Greece
Athens
Αθήνα
İzmir
Granada
Málaga
Gibraltar
Algiers
مدينة الجزائر
Tunis
تونس
Malta
Mediterranean Sea
Tunisia
Google

t FlyMates
he world

2000+
INSTITUTIONS

MILLIONS
OF TRANSACTIONS

BILLIONS
IN PAYMENTS

240
COUNTRIES
& TERRITORIES

CU

POWERFUL GLOBAL PAYMENT NETWORK

DevSecOps

- Development + Security + Operations

DevSecOps

- Development + Security + Operations
- DevSecOps - Security/Compliance as Code

DevSecOps

- Development + Security + Operations
- DevSecOps - Security/Compliance as Code
- Shift left. Involving Security and Compliance from the very beginning

DevSecOps

- Development + Security + Operations
- DevSecOps - Security/Compliance as Code
- Shift left. Involving Security and Compliance from the very beginning
- Proactivity and early involvement have better results. Less re-work to be done

DevSecOps

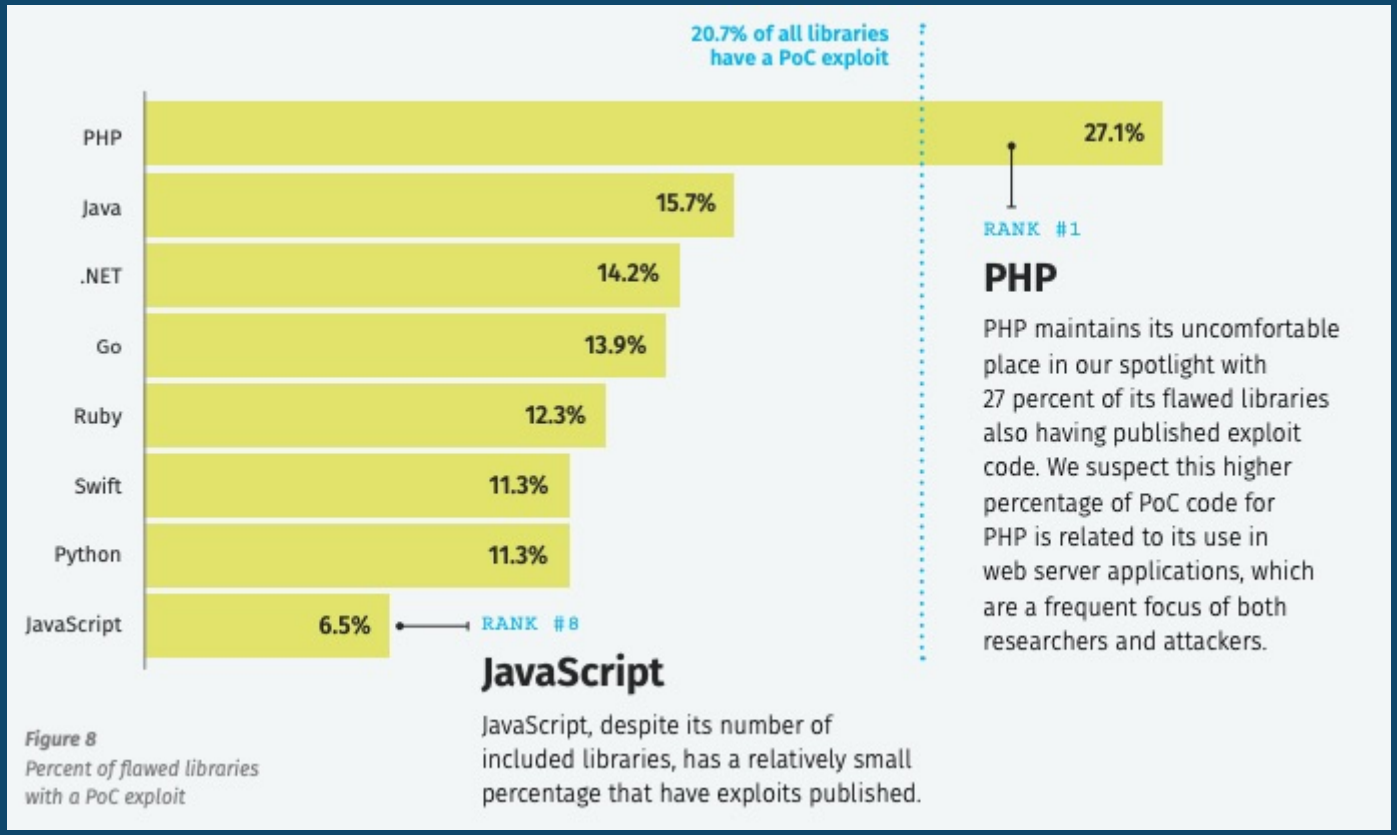
- Development + Security + Operations
- DevSecOps - Security/Compliance as Code
- Shift left. Involving Security and Compliance from the very beginning
- Proactivity and early involvement have better results. Less re-work to be done
- Include controls in the pipeline

DevSecOps

- Development + Security + Operations
- DevSecOps - Security/Compliance as Code
- Shift left. Involving Security and Compliance from the very beginning
- Proactivity and early involvement have better results. Less re-work to be done
- Include controls in the pipeline

Security is everyone's responsibility

Exploits



* Veracode report - https://www.veracode.com/sites/default/files/pdf/resources/reports/state-of-software-security-open-source-edition-veracode-report.pdf

Ezines/196 - AppSec Ezine

```
 99  Description: wget HTTP integer overflow (CVE-2017-13089).
100
101  URL: https://edoverflow.com/2017/ruby-resolv-bug/
102  Description: Bypassing SSRF filters by abusing a bug in Ruby's resolver (CVE-2017-0904).
103
104
105  '
```

Showing the top two matches Last indexed on 29 Jun 2018

Ezines/336 - AppSec Ezine

```
19  URL: https://research.securitum.com/html-sanitization-bypass-in-ruby-sanitize-5-2-1/
20  Description: HTML sanitization bypass in Ruby Sanitize < 5.2.1.
```

Showing the top two matches Last indexed on 24 Jul

Ezines/38 - AppSec Ezine

```
38  URL: https://github.com/m4rco-/dorothy2
39  Description: A malware/botnet analysis framework written in Ruby.
40
41  URL: https://github.com/sektioneins/pcc/wiki/PHP-htaccess-injection-cheat-sheet
```

Showing the top match Last indexed on 29 Jun 2018

https://github.com/Simpsonpt/AppSecEzine/

Audits

AICPA SOC
a cpa org/soc4so
SOC for Service Organizations | Service Organizaciones
SOC 2 TYPE II CERTIFIED

PCI DSS COMPLIANT

Privacy Shield Certified

Fixing is mostly non-breaking!

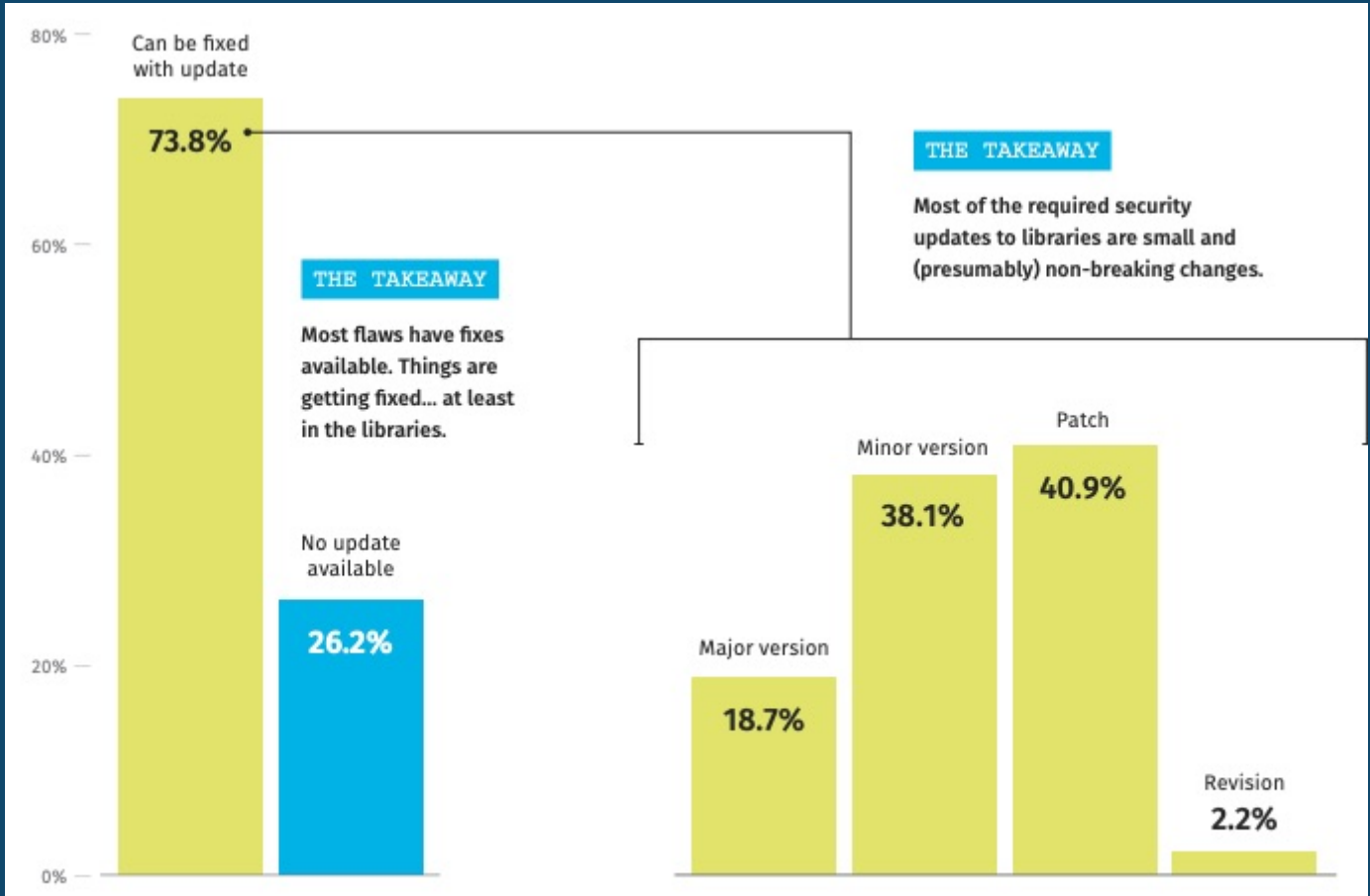


Figure 16 Percent of library flaws with an available update

Figure 17 Types of updates available for library flaws

* Veracode report - https://www.veracode.com/sites/default/files/pdf/resources/reports/state-of-software-security-open-source-edition-veracode-report.pdf

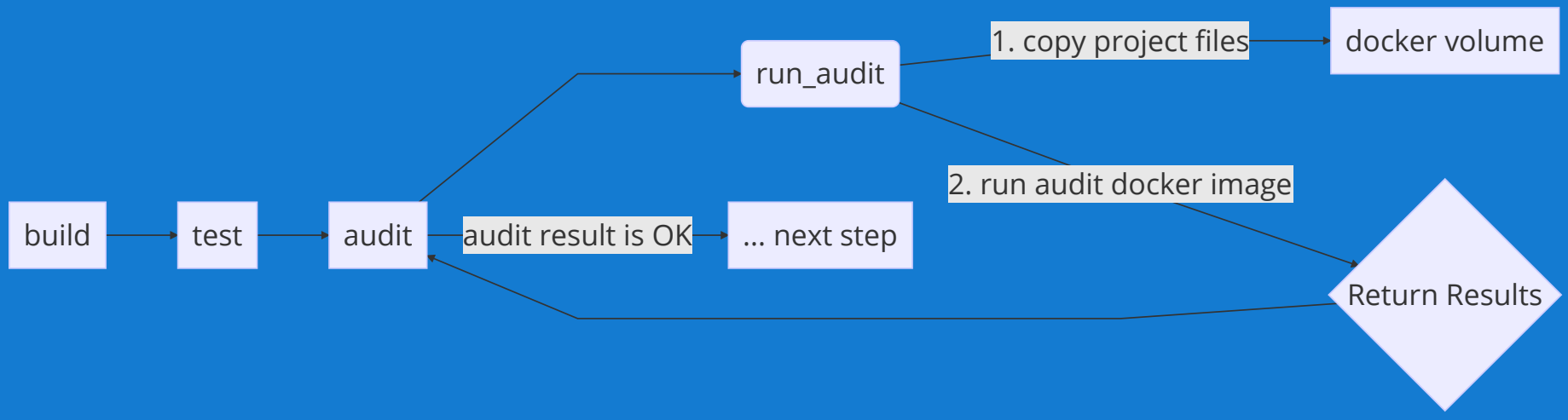
Intro to Victoria

- Victoria is Flywire's internal PaaS
- It makes the development, testing, and deployment of applications quick, simple, and cost-effective.
- It has cloud best practices built in, such as scalability, high-availability, observability, operability, etc
- It is SOC 2 Compliant by default
- Follows 12factor.net methodology

Main Victoria features

- DSL
  Decoupling applications from the underlying infrastructure
- Secrets management
  Follows SOC 2 segregation of duties requirements
- Standardized CI/CD Pipeline
  With hooks for customization in every stage
- Security team integrated in the SDLC
  Manual and automated controls

CI Flow* for Audit



*** 12factor.net**

V. Build, release, run - Strictly separate build and run stages

X. Dev/prod parity - Keep development, staging, and production as similar as possible

flywire

Setup

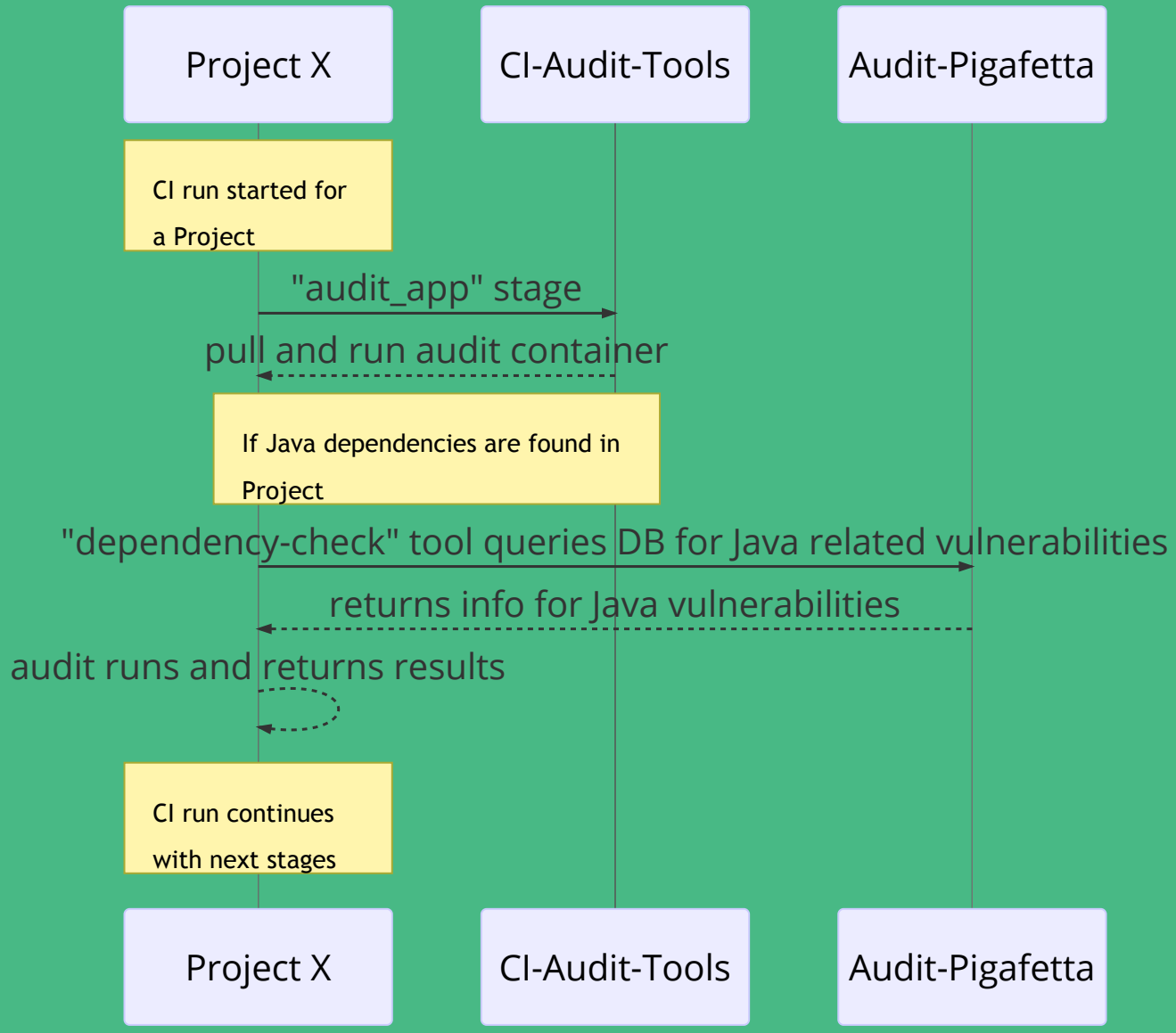
```
RUN cd /src && bundle install && cd - && \
    wget -qO- -O owasp.zip ${DEPENDENCY_CHECK_URL} && \
    unzip owasp.zip && rm owasp.zip && mv dependency-check / && \
    wget -q -P /dependency-check/plugins/ ${POSTGRESQL_PLUGIN_URL} && \
    chmod u+x /src/*.sh && /src/update_osa.sh
CMD ["/src/run_osa.sh"]
```

- update: install & update tools
- run: find dependencies files per language and audit them

Tools Used for Audit

- ruby: `bundle-audit`
  (https://github.com/rubysec/bundler-audit)
- node: `npm audit`
  (https://docs.npmjs.com/cli/v6/commands/npm-audit)
- python: `safety`
  (https://pypi.org/project/safety/)
- java: `dependency-check` with external DB
  (https://jeremylong.github.io/DependencyCheck/)

What happens in case of Java deps



History Time

Venetian scholar and explorer. He joined the expedition to the Spice Islands led by explorer Ferdinand Magellan. During the expedition, he served as Magellan's assistant and kept an accurate **journal**

flywire

About audit-pigafetta

- Keeps journal of Java vulnerabilities
- “Victorized” Sinatra app - hello world
- /health endpoint

  `{"node_name":"ip-172-31-202-134.ec2.internal","services_health":{"postgres":"ok"}}`
- when deployed it loads structure of the DB if it not exists*

  deploy CI stage -> `rake db:structure:load`
  - structure provided by the tool developer as .sql file

* 12factor.net | XII. Admin processes
Run admin/management tasks as one-off processes

flywire

Onboarding Steps

- Victoria core operation pushes changes to ci/audit file of projects

- Project image needs to
  - be built and pushed before audit stage
  - have dependencies files available (Gemfile.lock, package-lock.json, jar files, etc.)
- Guides available

flywire

What happens during audit

```
docker volume create --name code_from_image

docker run --rm --user 0 -v code_from_image:/to "${PROJECT_IMAGE}" cp -a . /to

docker pull "${AUDIT_IMAGE}"

docker run --rm -v "code_from_image:/app" "${AUDIT_IMAGE}"

docker volume rm code_from_image
```

flywire

What happens during audit

```
docker volume create --name code_from_image

docker run --rm --user 0 -v code_from_image:/to "${PROJECT_IMAGE}" cp -a . /to

docker pull "${AUDIT_IMAGE}"

docker run --rm -v "code_from_image:/app" "${AUDIT_IMAGE}"

docker volume rm code_from_image
```

Creates a docker volume

flywire

What happens during audit

```
docker volume create --name code_from_image

docker run --rm --user 0 -v code_from_image:/to "${PROJECT_IMAGE}" cp -a . /to

docker pull "${AUDIT_IMAGE}"

docker run --rm -v "code_from_image:/app" "${AUDIT_IMAGE}"

docker volume rm code_from_image
```

Copies project files to volume. Should include dependecies too

flywire

What happens during audit

```
docker volume create --name code_from_image

docker run --rm --user 0 -v code_from_image:/to "${PROJECT_IMAGE}" cp -a . /to

docker pull "${AUDIT_IMAGE}"

docker run --rm -v "code_from_image:/app" "${AUDIT_IMAGE}"

docker volume rm code_from_image
```

Pulls and runs audit container

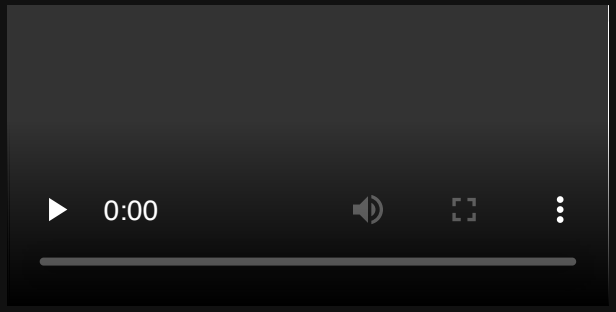
flywire

audit file used in CI

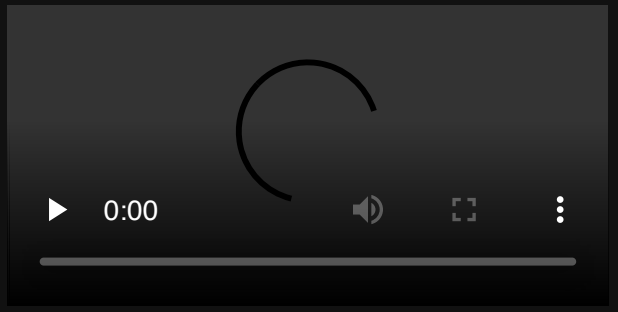
```
source ./deploy/scripts/functions.sh

run_audit
```

it can be customized if needed

Results



Exceptions



Takeaways

flywire

Takeaways

- Involve security in the pipeline, shift left; Security is everyone’s responsibility

flywire

Takeaways

- Involve security in the pipeline, shift left; Security is everyone’s responsibility
- Be open to changes and adapt as needed

flywire

Takeaways

- Involve security in the pipeline, shift left; Security is everyone’s responsibility
- Be open to changes and adapt as needed
- Canary releases, incremental work towards blocking mode

flywire

Takeaways

- Involve security in the pipeline, shift left; Security is everyone’s responsibility
- Be open to changes and adapt as needed
- Canary releases, incremental work towards blocking mode
- Fixing some issues is better than none

flywire

Curious?

https://www.flywire.com/careers

flywire

THANK YOU