DevOps Pro Europe 2019

# Securing your CI/CD Pipeline

Jeroen Willemsen –
Devops Pro Europe 2019

# About me

Jeroen Willemsen
@commjoenie
jeroen.willemsen@owasp.org

"Security architect"
"Full-stack developer"
"Mobile security"

Xebia Security

OWASP
Open Web Application Security Project

# Goal

Help you on the next step of your security journey

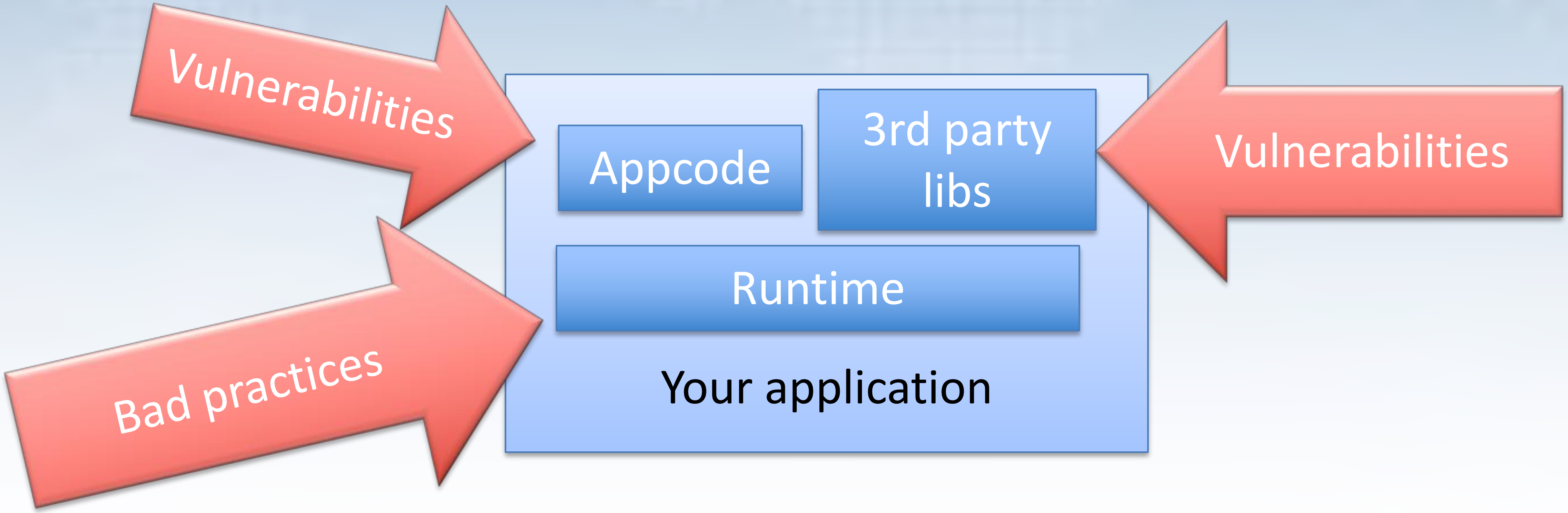
# Agenda

- A SECURE pipeline
- A security pipeline
- Recommendations

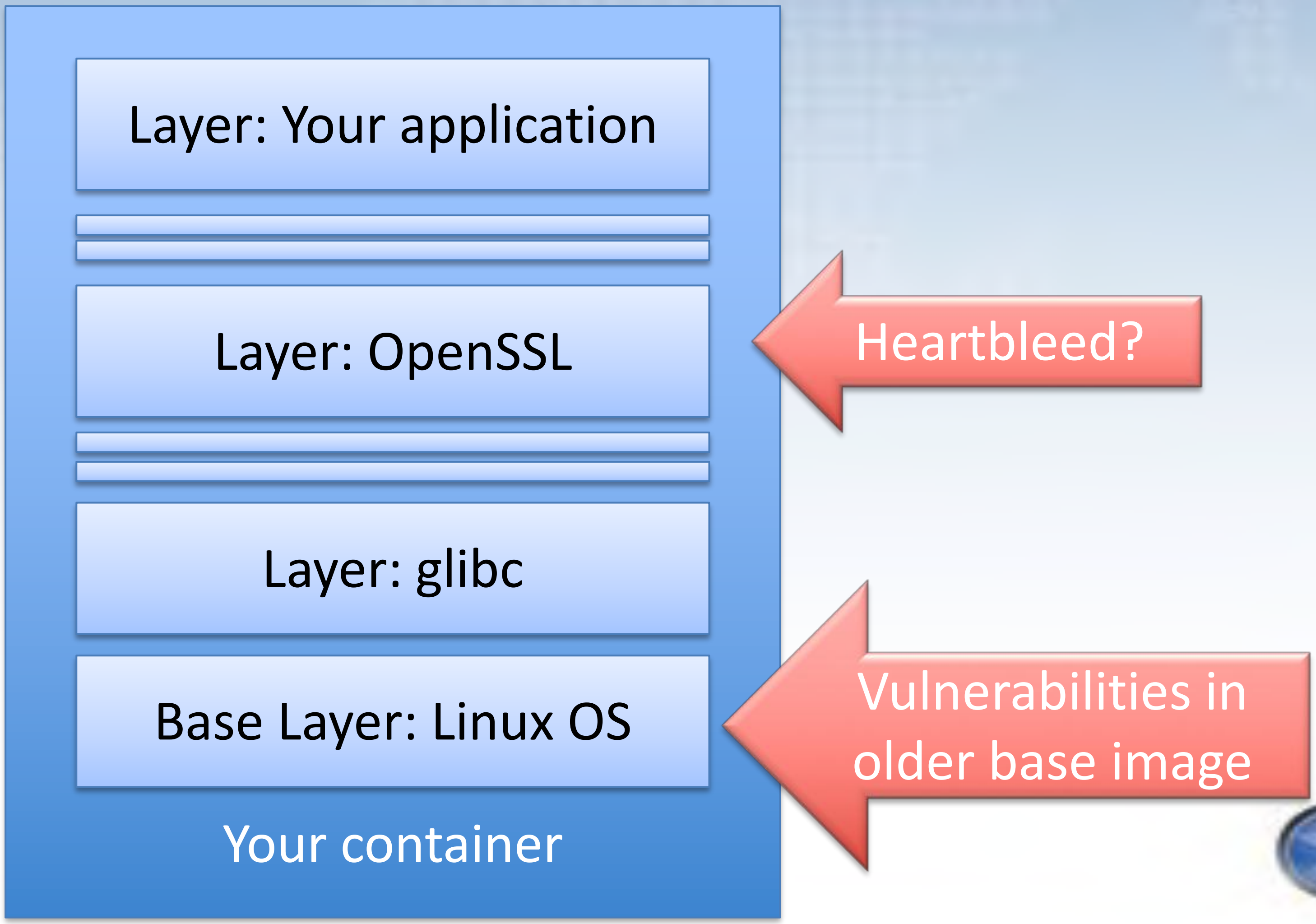
# WHY?

Why do we care?

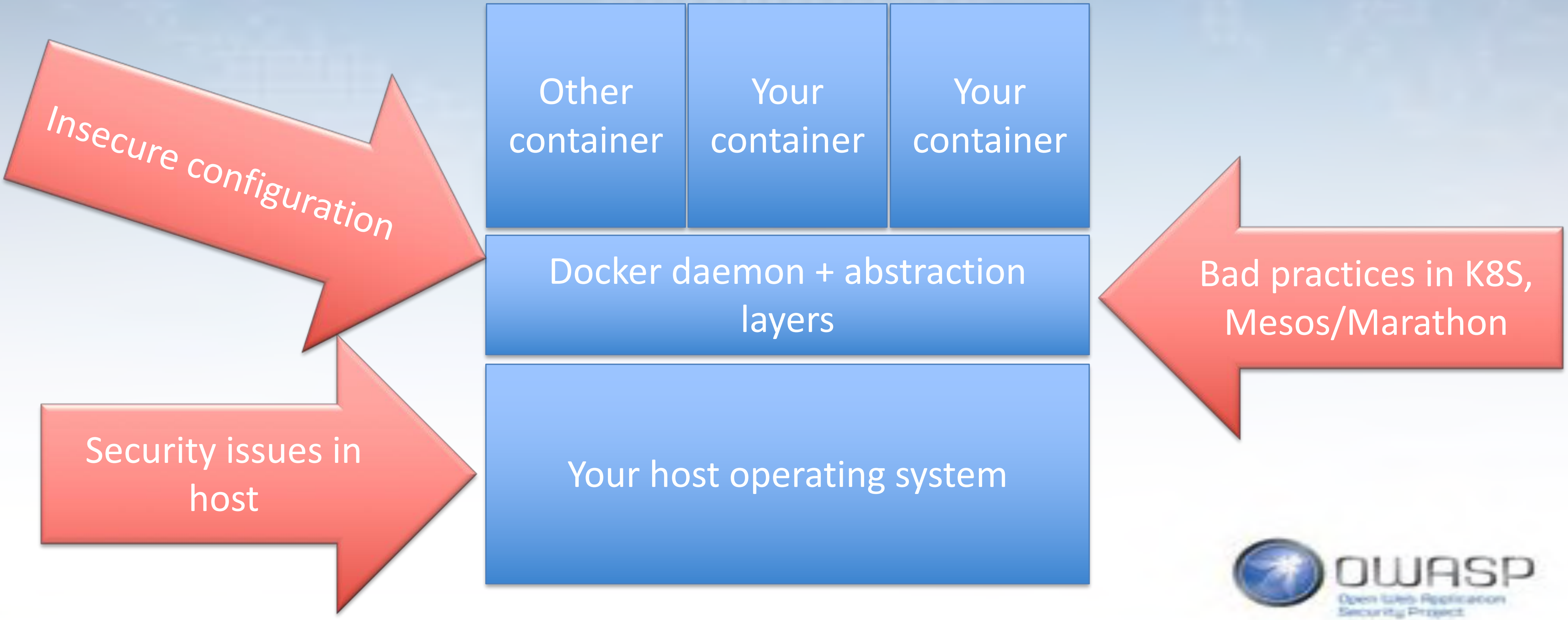
# Your application in a container…



# Your application in a container…



# Your application in a container on a host



Other container

Your container

Your container

Docker daemon + abstraction layers

Your host operating system

Insecure configuration

Security issues in host

Bad practices in K8S, Mesos/Marathon

# General issues

- Secrets leaked
- Network compromise?
- Compromised source code
- Social engineering?
- IAM issues
- “Cloud” hosting configuration

# Time to panic!

# Or maybe not?

YOU GET A SECURITY CHECK!
YOU GET A SECURITY  CHECK!
EVERYBODY GETS A SECURITY CHECK!
imgflip.com

No worries, I protected Prod.
They got our buildserver. Prod is theirs!
imgflip.com

# A SECURE PIPELINE

Access denied?

# ***Your CI/CD pipeline is your production environment…***

# Every production environment should have

- **Monitoring & Alerting**
- **Identity & Access Management**
- **Secrets management**
- **Hardening: defense in depth!**
- Automated deployments & immutability
- Been designed for easy recovery
- Automated Configuration & change management
- Properly trained teams & security processes

YOU DO NOT SEE ANYTHING
IN THE DARK.

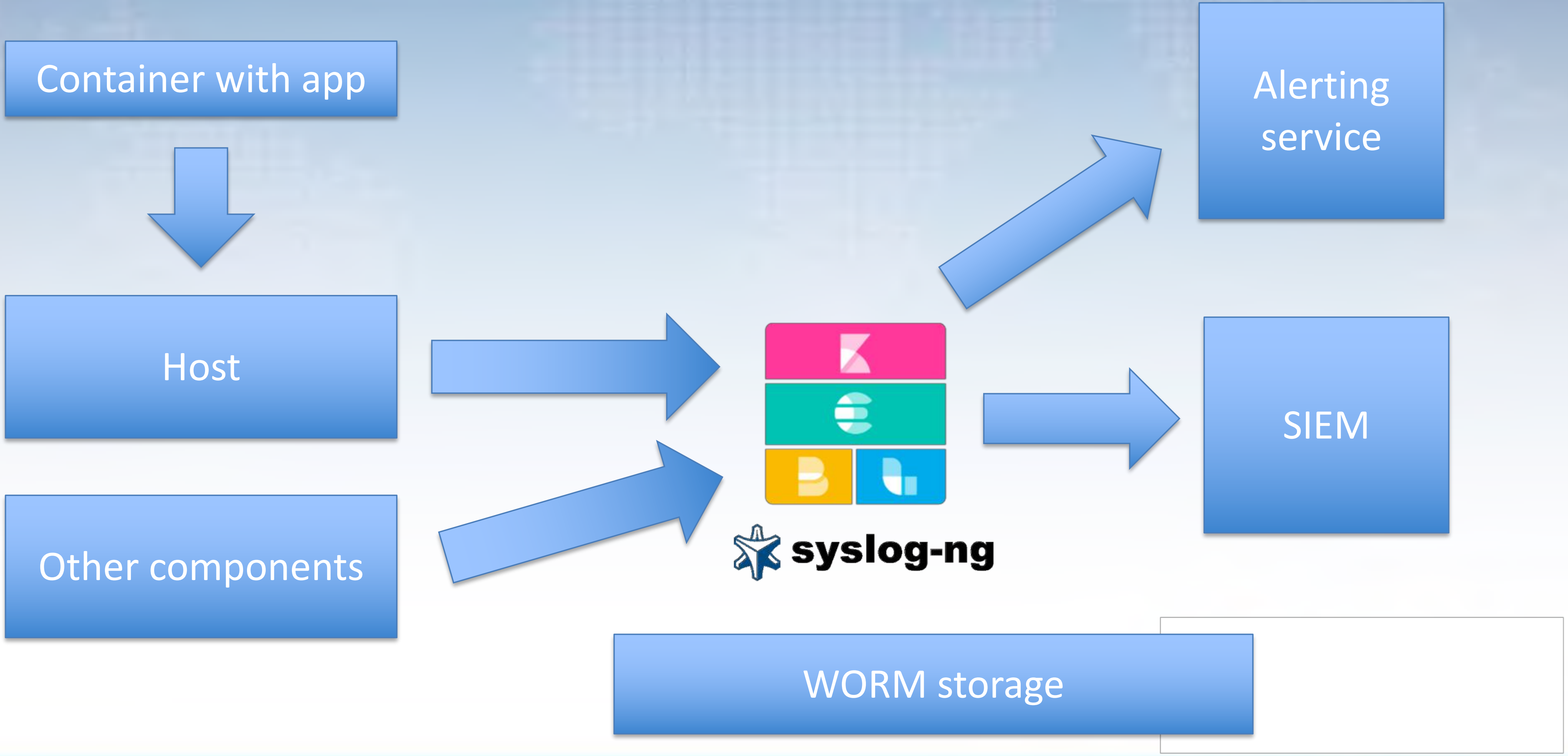
YOU CANNOT HEAR AN ATTACKER IF YOU DO NOT LISTEN.

# Monitoring & Alerting

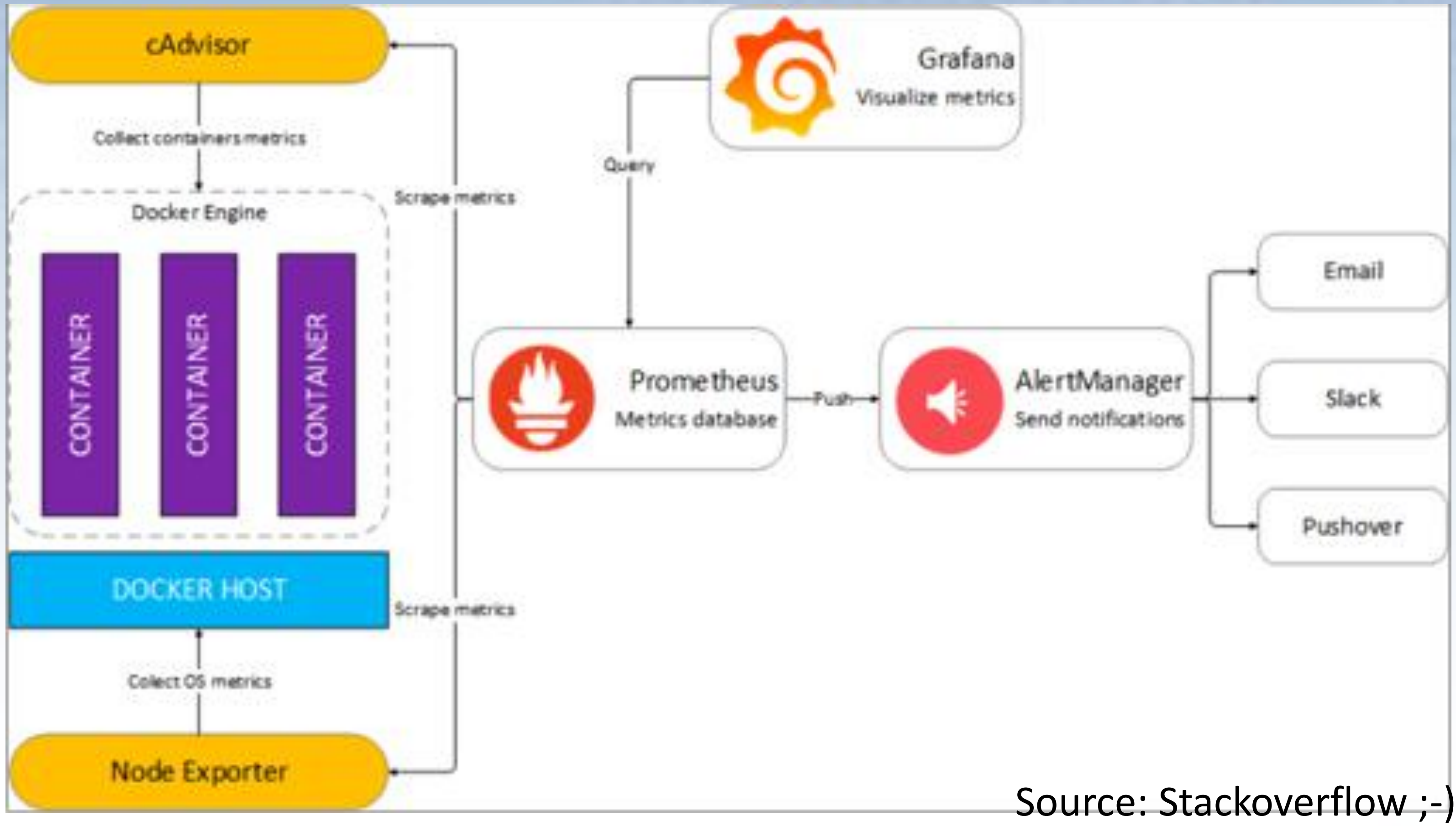
| | |
|---|---|
| Container with app | Application logs,<br>logs from container components |
| Host | AuditD logs, OS logs, SSH, etc. |
| Network components | Flow logs, component logs |

Something something cloud…

# Monitoring & Alerting



# Monitoring & Alerting



Source: Stackoverflow ;-)

# Monitoring & Alerting: ACT UPON IT!

BeNeLux Day 2018: Lessons From The Legion (OWASP BeNeLux Remix) - Nick Drage

Nick Drage

"Lessons From The Legion"
( The OWASP BeNeLux Remix )

Lessons from the Legion
Nick Drage

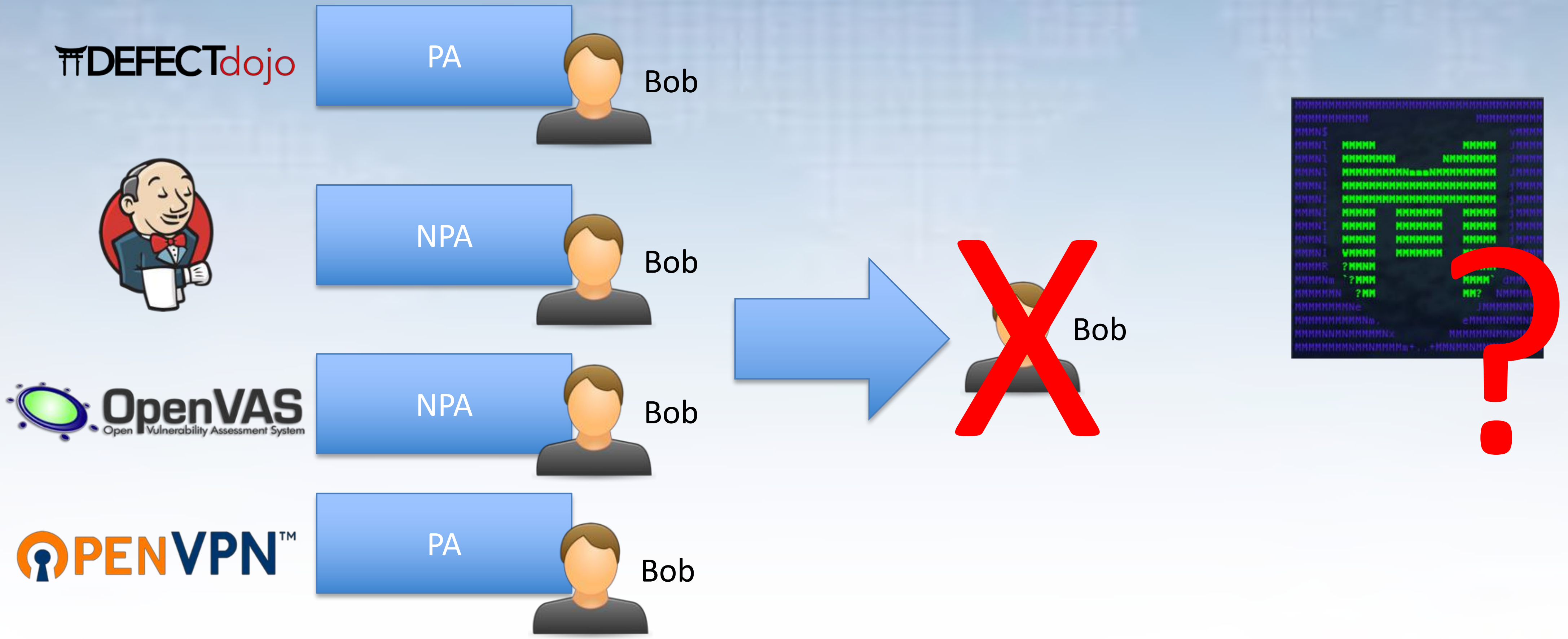
# Identity & Access Management

Developers are your production users

Ops engineers / platform teams / SRE teams are your admins

Treat them as such….

# Identity & Access Management



PA Bob

NPA Bob

NPA Bob

PA Bob

Bob

?

# Identity & Access Management

- Have a central identity store
- Integrate using Auth0, ADFS, …etc..
- RBAC
- Verify:
  - Should a user still have these roles?
  - Are all actions logged? Are the right people informed?
  - Do we get alerts when privileged actions happen?
  - Do we have segregation of duties? Is there a 4-eyes principle in place?
  - Is MFA enabled?
  - Is…… etc..

# Have its secrets managed

HashiCorp Vault

Azure Key vault

AWS KMS

AWS Cloud-HSM

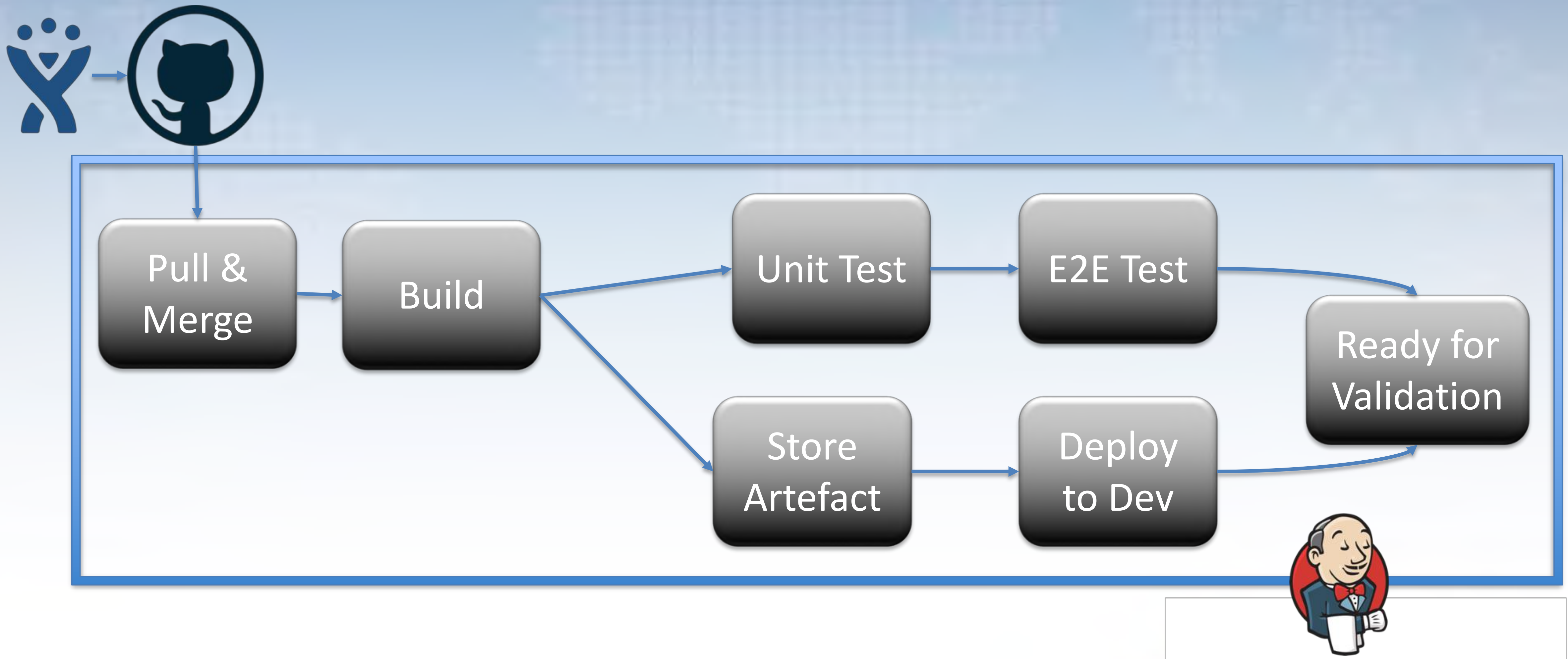
GCP KMS

- Have Audit-logging & access controls
- Don’t reuse secrets for different purposes
- Integrate with the consumer of the secret, not its environment
- Rotate secrets!

# A SECURITY PIPELINE

Wielding the power of security tooling

# The pipeline (App)



Pull & Merge

Build

Unit Test

E2E Test

Store Artefact

Deploy to Dev

Ready for Validation

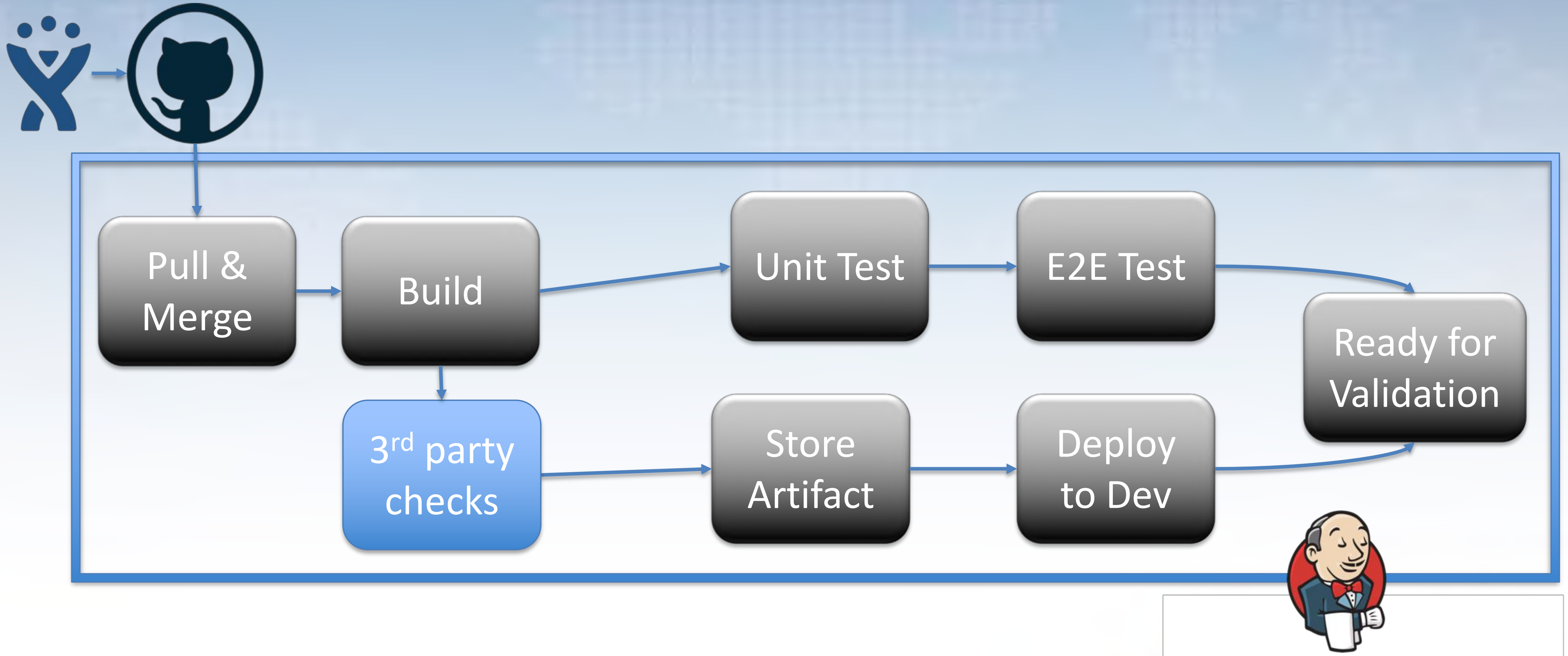
# The pipeline (App) – start with dependencies

Add dependency & license checkers on top of quality tooling.

↓

Get feedback FAST!

# The pipeline (App) – start with dependencies



Pull & Merge

Build

3rd party checks

Unit Test

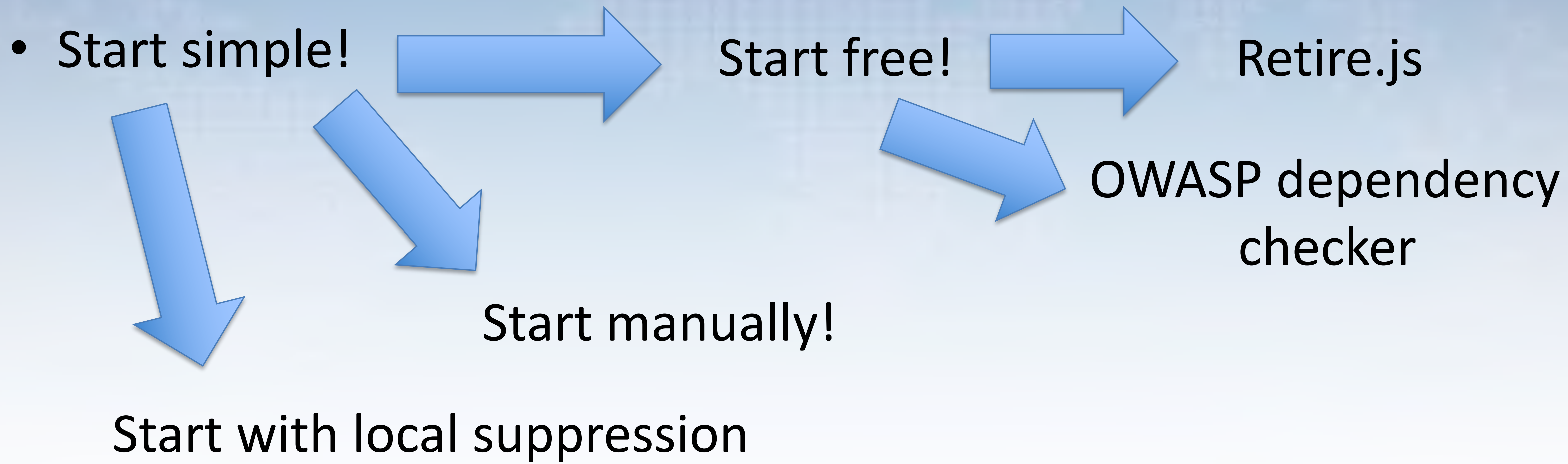
E2E Test

Store Artifact

Deploy to Dev

Ready for Validation

# The pipeline (App) – start with dependencies



- Start simple!

Start free!

Retire.js

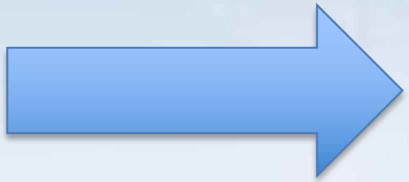
OWASP dependency checker

Start manually!

Start with local suppression

# The pipeline (App) – start with dependencies

- Start simple!
- Automate!
  - Having duplicates?
  - Having false positives?
  - Don’t fail / notify on findings until first cleanup
  - Cache your vulnerability feeds



Vulnerarибility

# The pipeline (App) – start with dependencies

- Start simple!
- Automate!
- Still in need of commercial tool? Test before buy!

# The pipeline (App) – Static Code analysis

- So what about application sources?
- Enter static code analysis
- Quality tooling ⬄ Security tooling

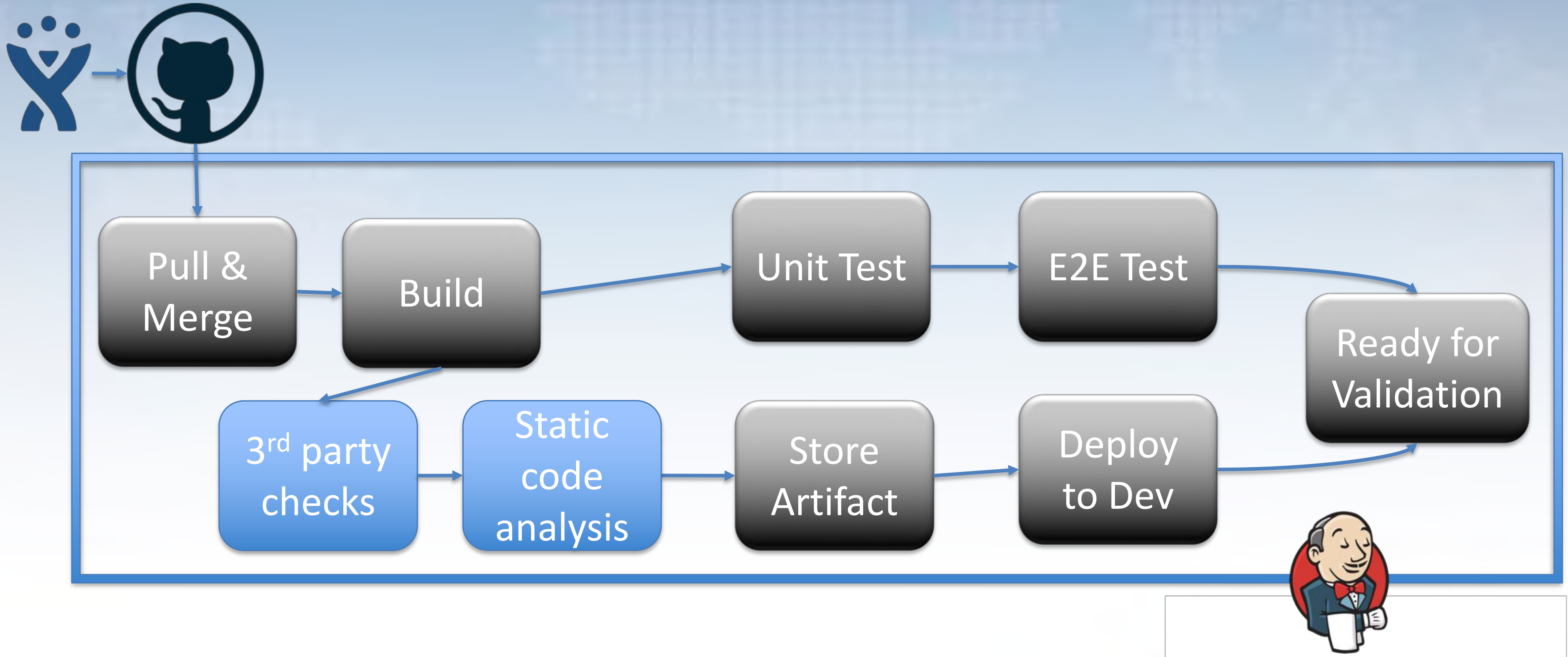
- See “DevSecOps: How to Use DevOps to Make You More Secure” by Zane yesterday!

CHECKMARX

FORTIFY®

VERACODE

# The pipeline (App) – Static Code analysis



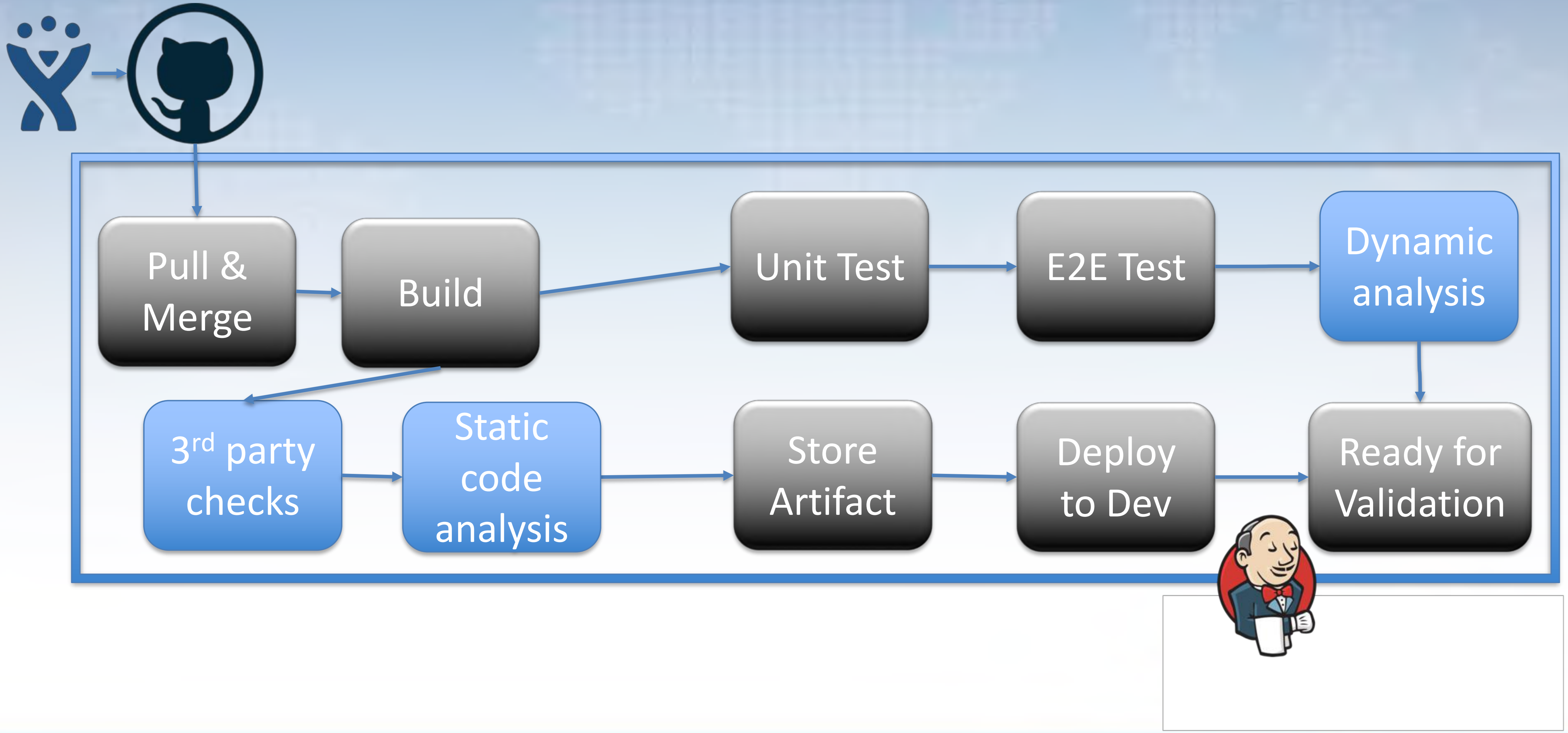
# The pipeline (App) – Dynamic analysis

Static code analysis does not find everything…

… And is often very expensive.

Let’s do some dynamic analysis!

# The pipeline (App) – Dynamic analysis



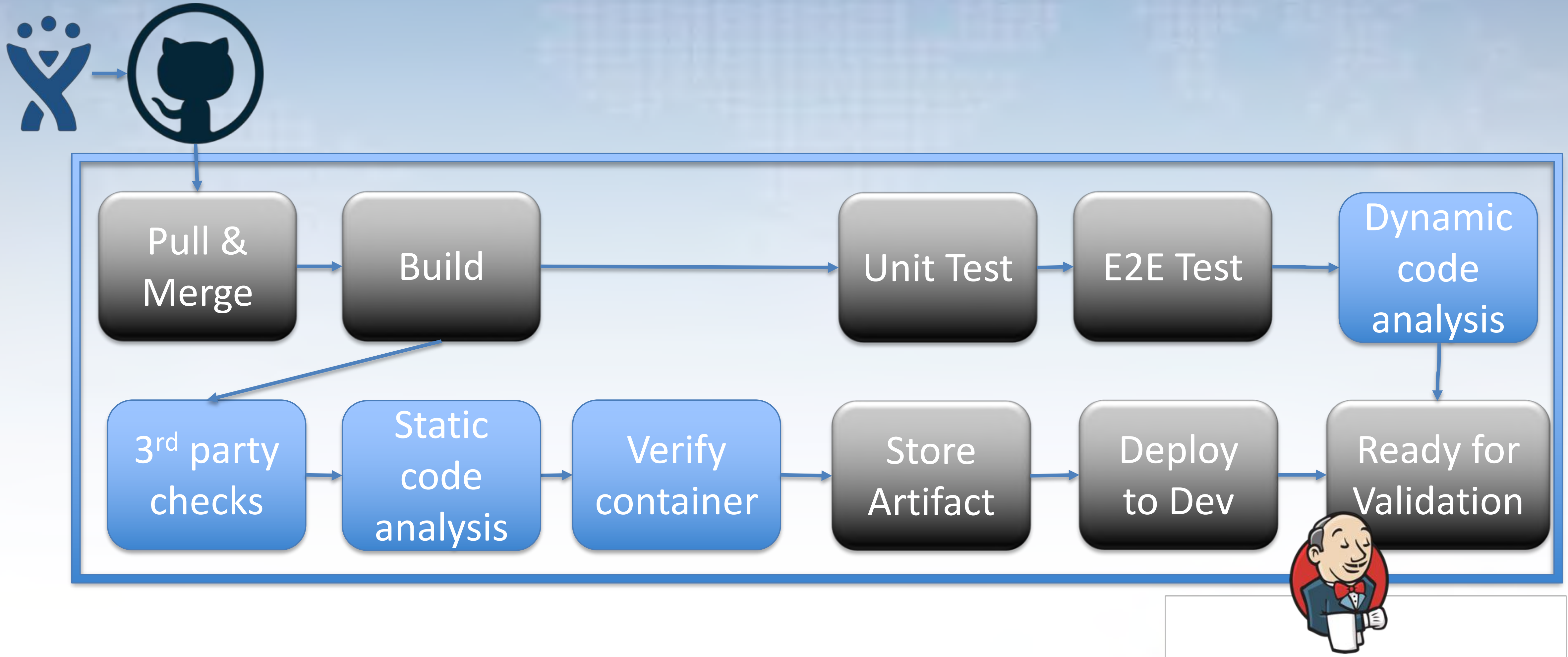
# The pipeline (App) – Dynamic analysis

- Start simple: free & manual.
  - Is it effective? Then automate!
- Take the OWASP Zed Attack Proxy (ZAP) for instance:
  - Know your app: authenticate!
  - Spin up a new container with ZAP per scan
  - Use a vulnerability manager to deduplicate findings
  - Scan small deltas as it may/will run out of memory.

# The pipeline (App) – Container verification

So how do you know, whether you have a “hardened” container?

# The pipeline (App) – Container verification



Pull & Merge

Build

Unit Test

E2E Test

Dynamic code analysis

3rd party checks

Static code analysis

Verify container

Store Artifact

Deploy to Dev

Ready for Validation

# The pipeline (App) – Container verification

- Does your container have vulnerabilities? (per layer)

- clair
- anchore
- Docker Hub
- Azure container registry

- Does your container have vulnerabilities? (as a unit)

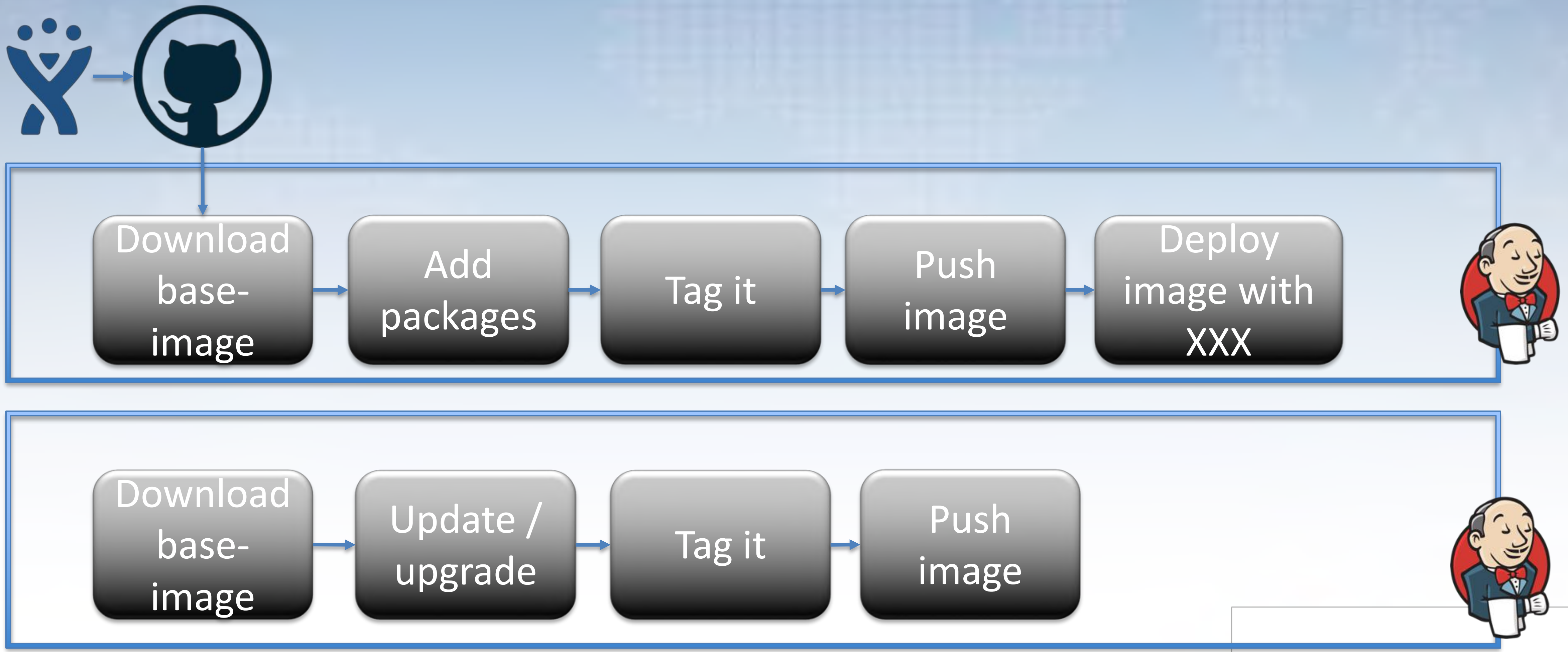
- Nessus N™
- OpenVAS Open Vulnerability Assessment System

- Does your container follow best practices & policies? (as a unit)

- OpenSCAP
- Lynis
- INSPEC

OWASP Open Web Application Security Project

# The pipeline (infra)



Download base-image → Add packages → Tag it → Push image → Deploy image with XXX

Download base-image → Update / upgrade → Tag it → Push image

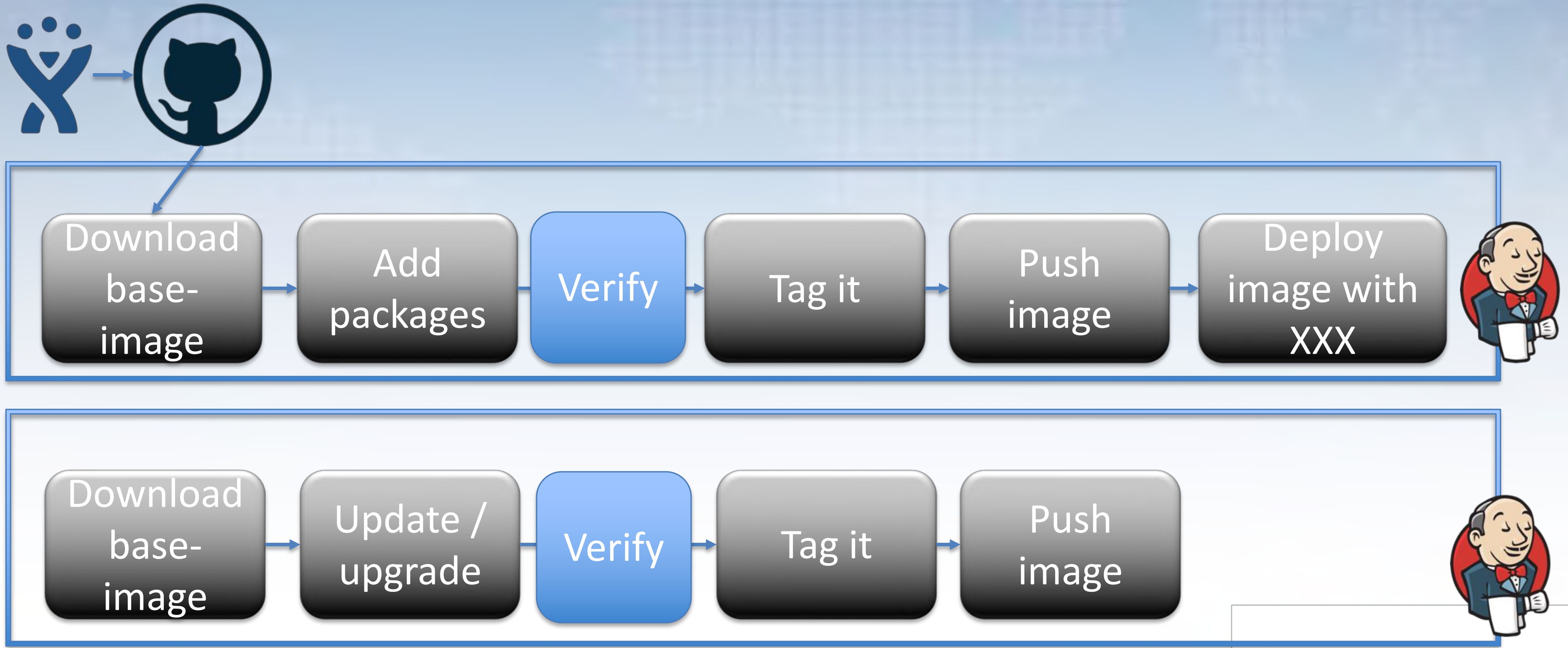
# The pipeline (infra)

- An image is actually similar to a container…

- Does your image have vulnerabilities?
  - Nessus™
  - OpenVAS Open Vulnerability Assessment System

- Does your image follow best practices & policies?
  - OpenSCAP
  - Lynis
  - INSPEC

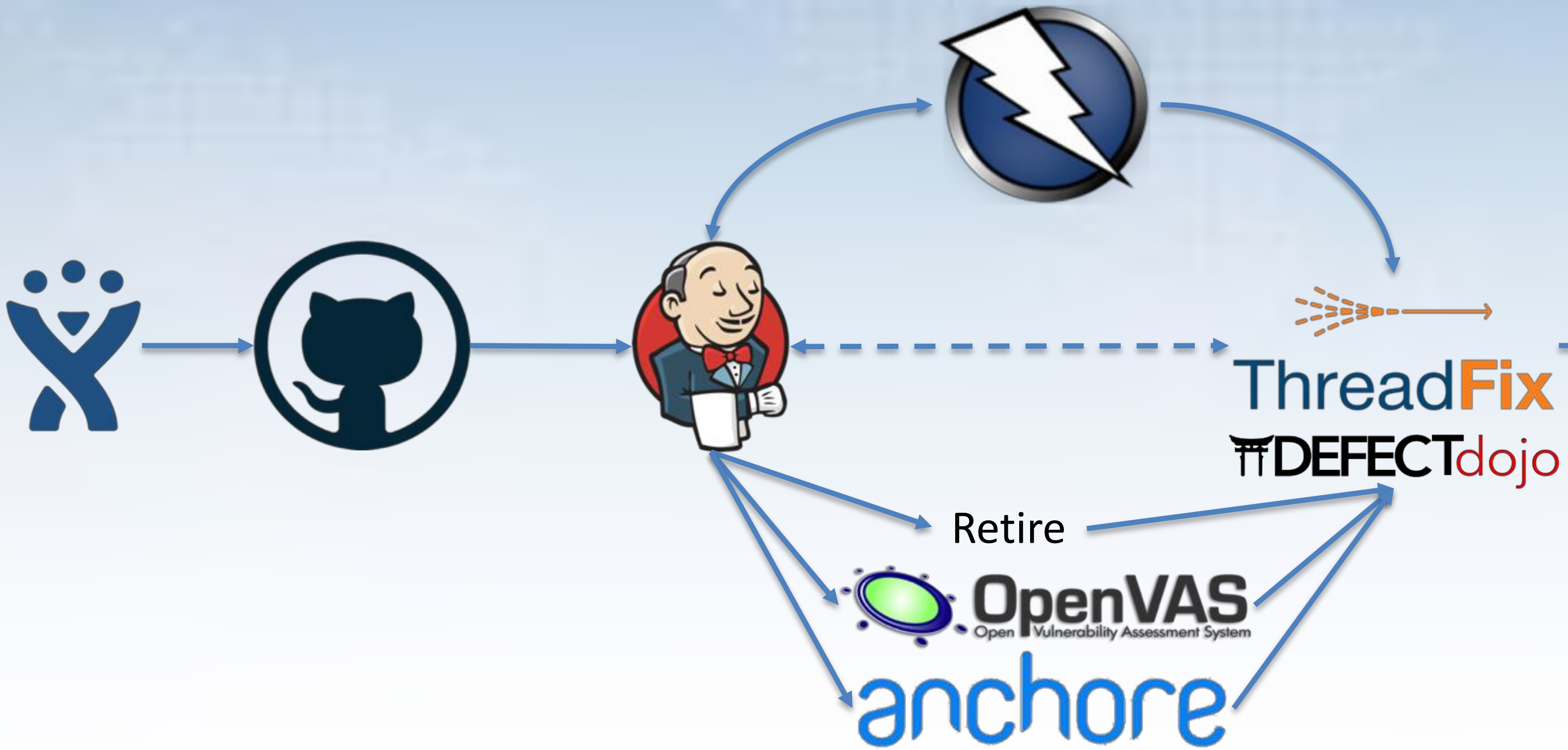
# The pipeline (infra)



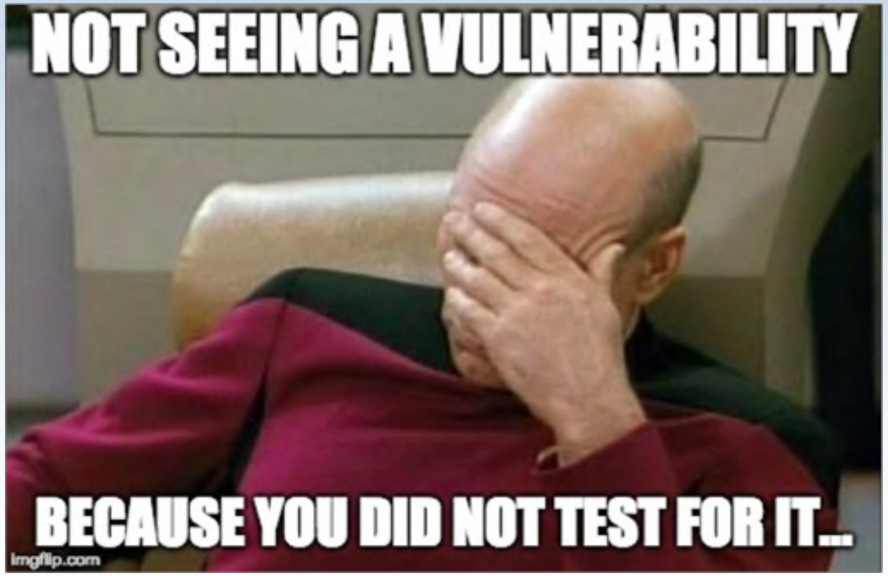
Download base-image → Add packages → Verify → Tag it → Push image → Deploy image with XXX

Download base-image → Update / upgrade → Verify → Tag it → Push image

# Putting it all together now

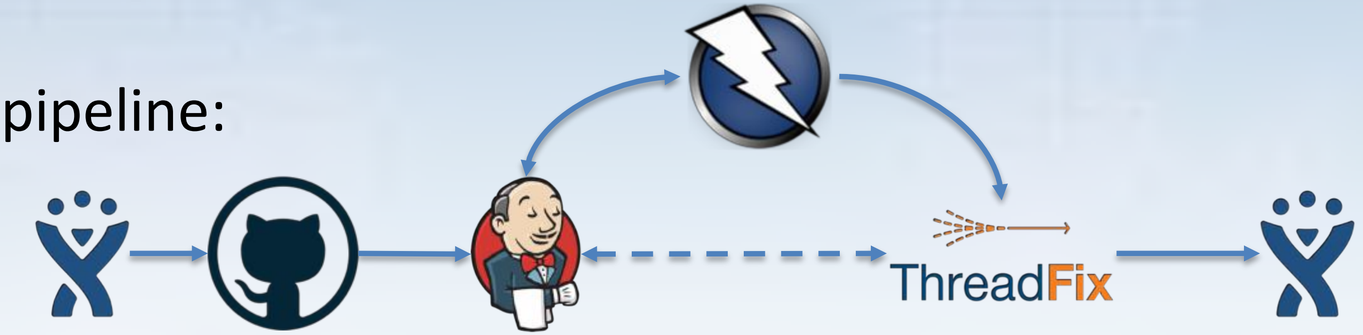


WAITING FOR
THE APPSECPIPELINE TO FINISH
imgflip.com

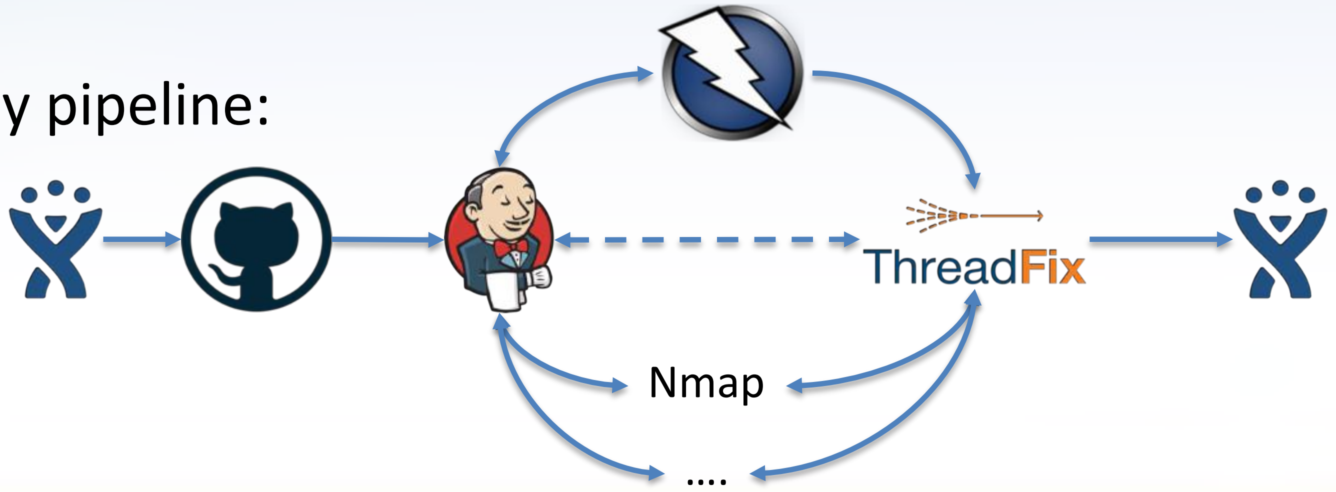

NOT SEEING A VULNERABILITY
BECAUSE YOU DID NOT TEST FOR IT...
imgflip.com

# Security pipelines

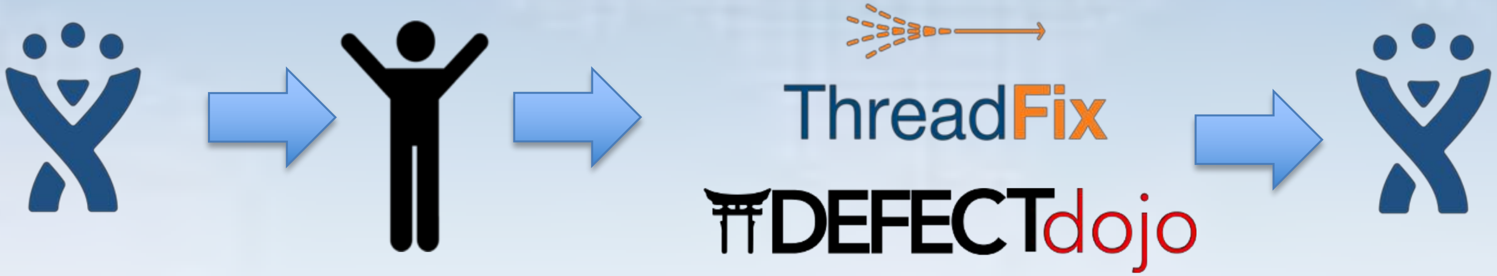
Base-pipeline:



Security pipeline:



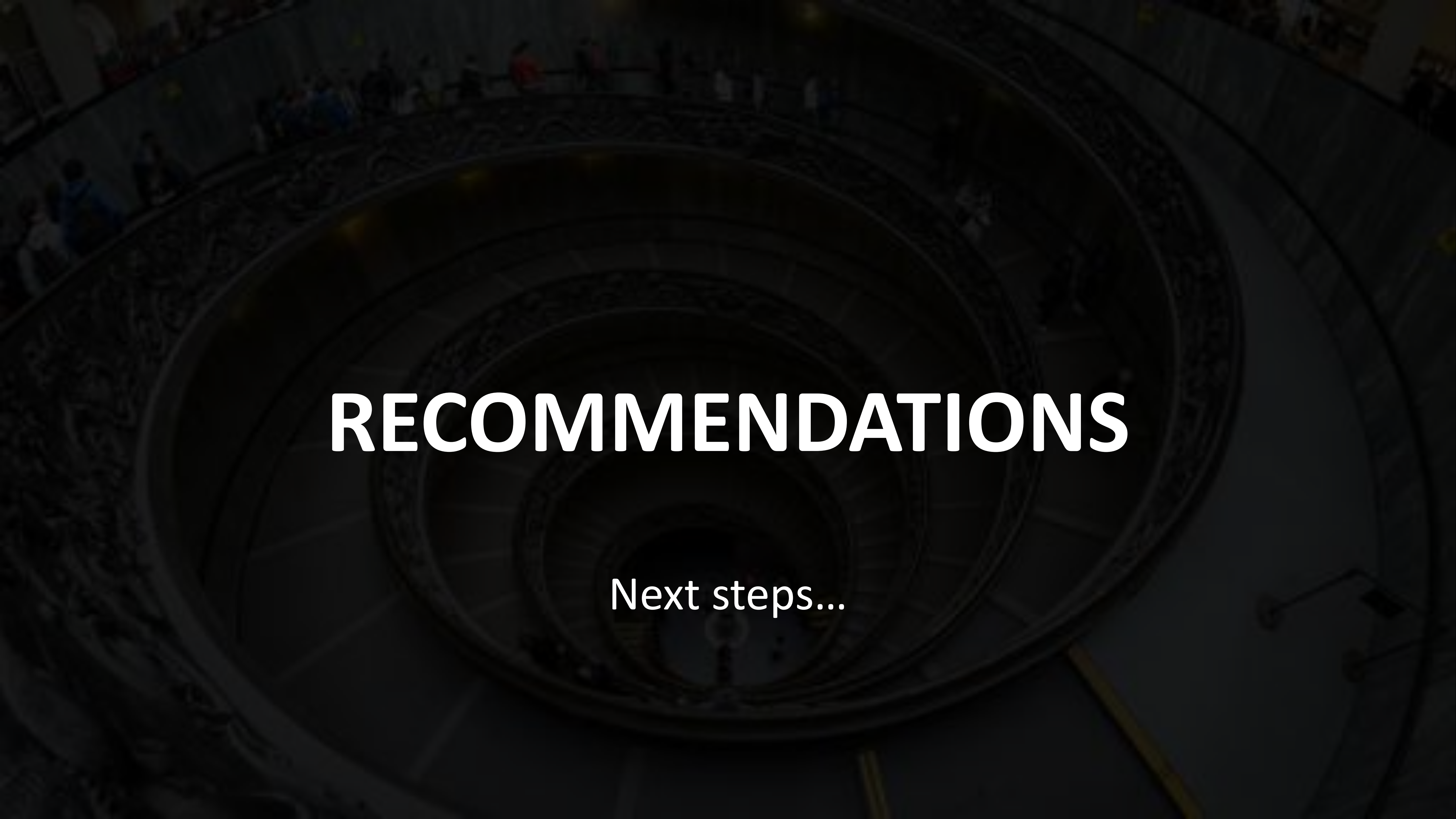
# Manual pentests...



- Optimize pipeline given findings
- A scan does not find everything

# Security testing – add your own

- What about possible weaknesses that are not in scope of the tools?
- Start threatmodelling & create evil user stories.
- Create automated tests based on the threatmodel



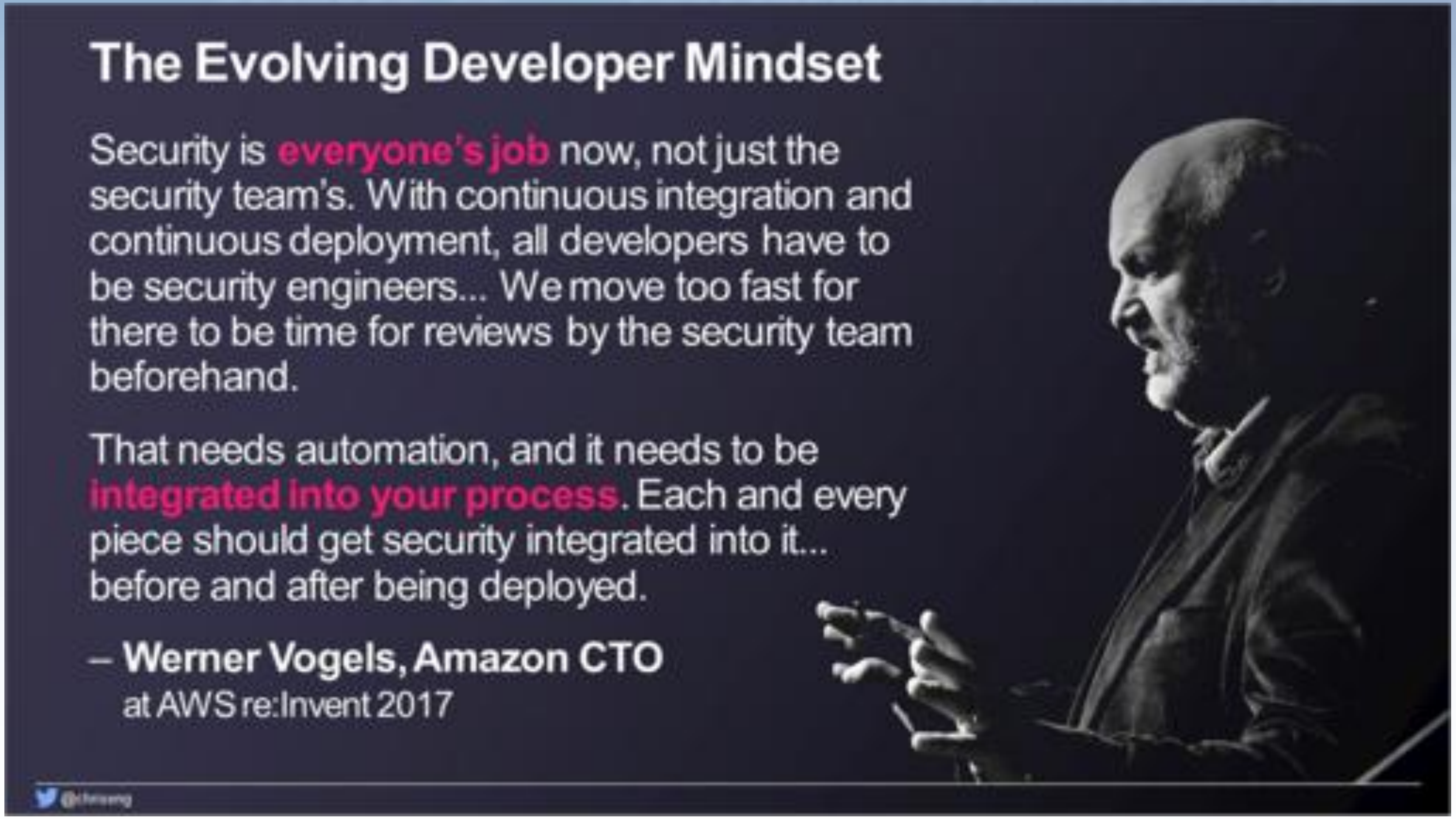
# RECOMMENDATIONS

Next steps…

# Recomendations

- Start small: begin with your infrastructure hardening
- Automate security!
- Want to take a next step? Take RISK BASED decisions!
- Protect your pipeline production-like
- Prepare to learn from your mistakes
- Never stop
- Measure the effectiveness & cost

# QUESTIONS?



## The Evolving Developer Mindset

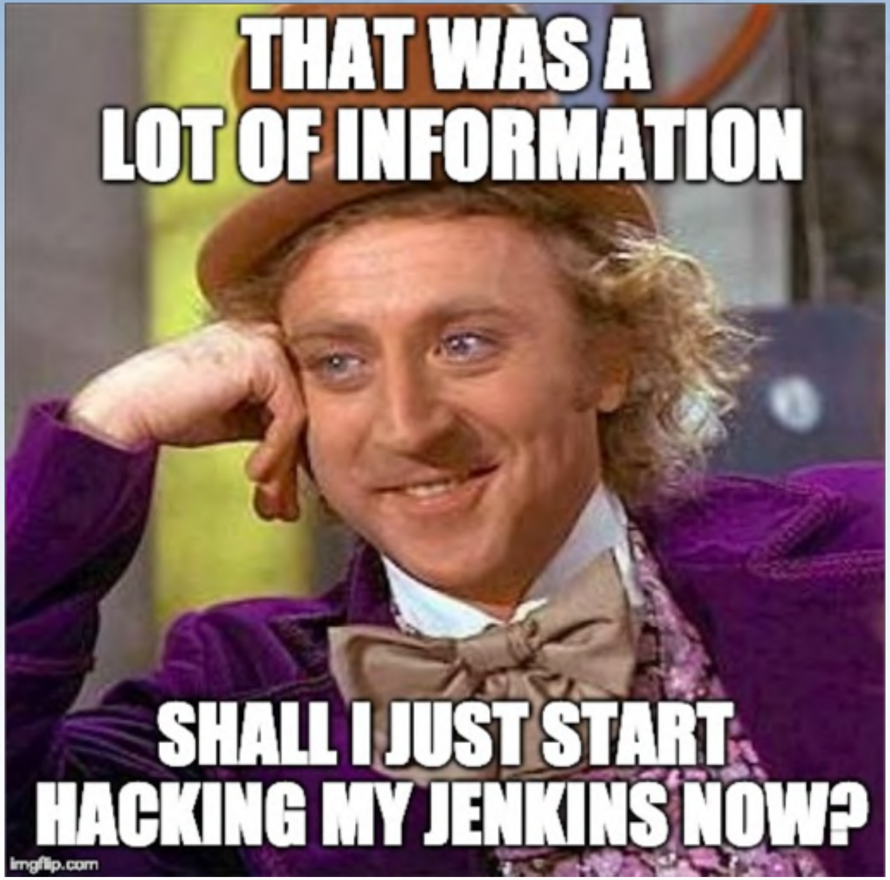
Security is everyone's job now, not just the security team's. With continuous integration and continuous deployment, all developers have to be security engineers... We move too fast for there to be time for reviews by the security team beforehand.

That needs automation, and it needs to be integrated into your process. Each and every piece should get security integrated into it... before and after being deployed.

– **Werner Vogels, Amazon CTO**
at AWS re:Invent 2017

@chriseng

# THANK YOU!



@commjoenie

jwillemsen@xebia.com

# Appendices

- Dependency checkers

# Dependency checkers

- OWASP Dependency checker (Wrappers for many)
- OSS Index
- Gemnasium
- SRC CRL
- Javascript: RetireJS, NSP, Snyk
- Others: BlackDuck, WhiteSource, Nexus, Protecode
- Tools are often supported by Gitlab/Github/etc.