Automating Third-party Libraries Security Testing

Peter Mosmans
LEAD PENETRATION TESTER

@onwebsecurity https://www.onwebsecurity.com

Scenario

Maeve

“We looked at our code”

“Should we also look at third party code that we use?”

Jennifer

“Yes, absolutely”

“Let me show you how you can do that”

Module Overview

Third-party libraries scanners

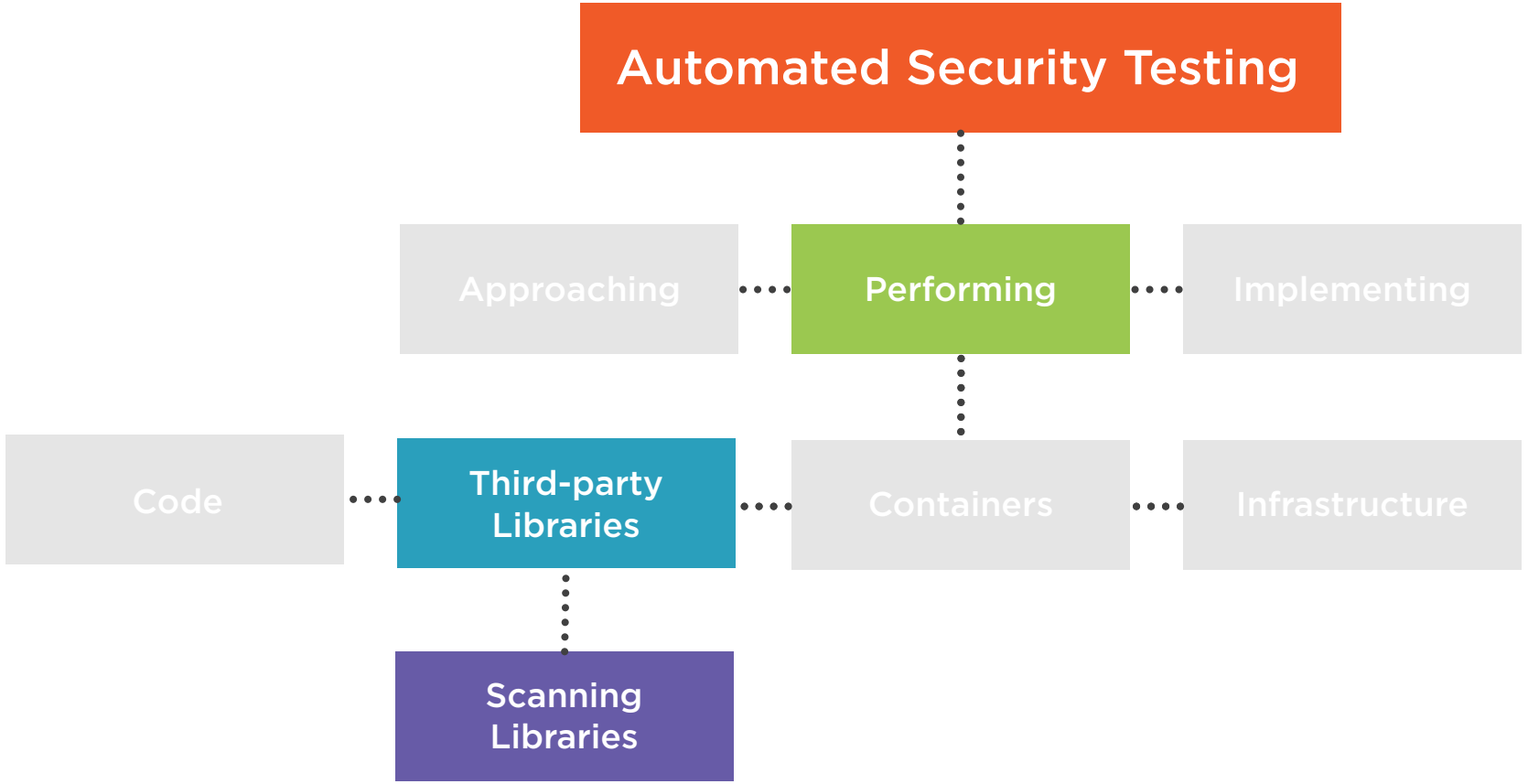
Where and when to use a scanner

Demos:

- Scanning for outdated and insecure third-party libraries
- Integrate scanner in pipeline

Summary

Automated Security Testing Overview

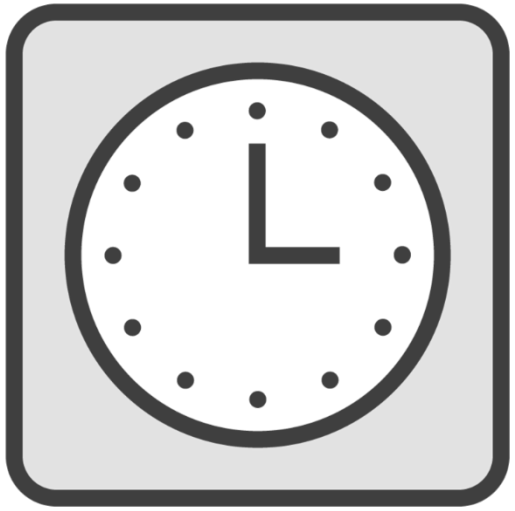


Third-party Libraries Scanners

More Information

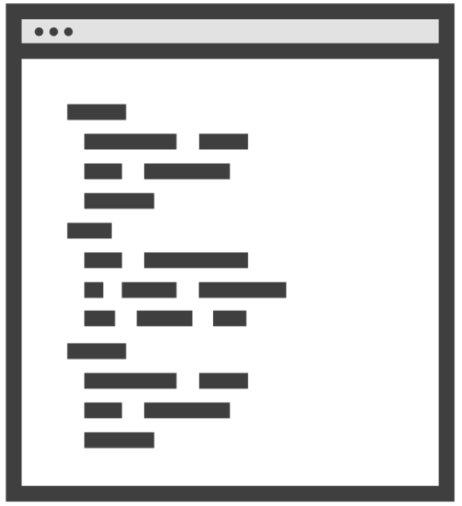
https://www.pluralsight.com/courses/secure-coding-using-components-known-vulnerabilities

Security of an Application



Degrades over time

Is dependent on other parts



Changes irrespective of the code itself

How Third-party Libraries Scanners Work

Characteristics of a Good Scanner

Uses multiple up-to-date sources for vulnerability reports

Understands multiple frameworks

Can parse manifests or software Bill of Materials

What Can A Scanner for Outdated Libraries Do?

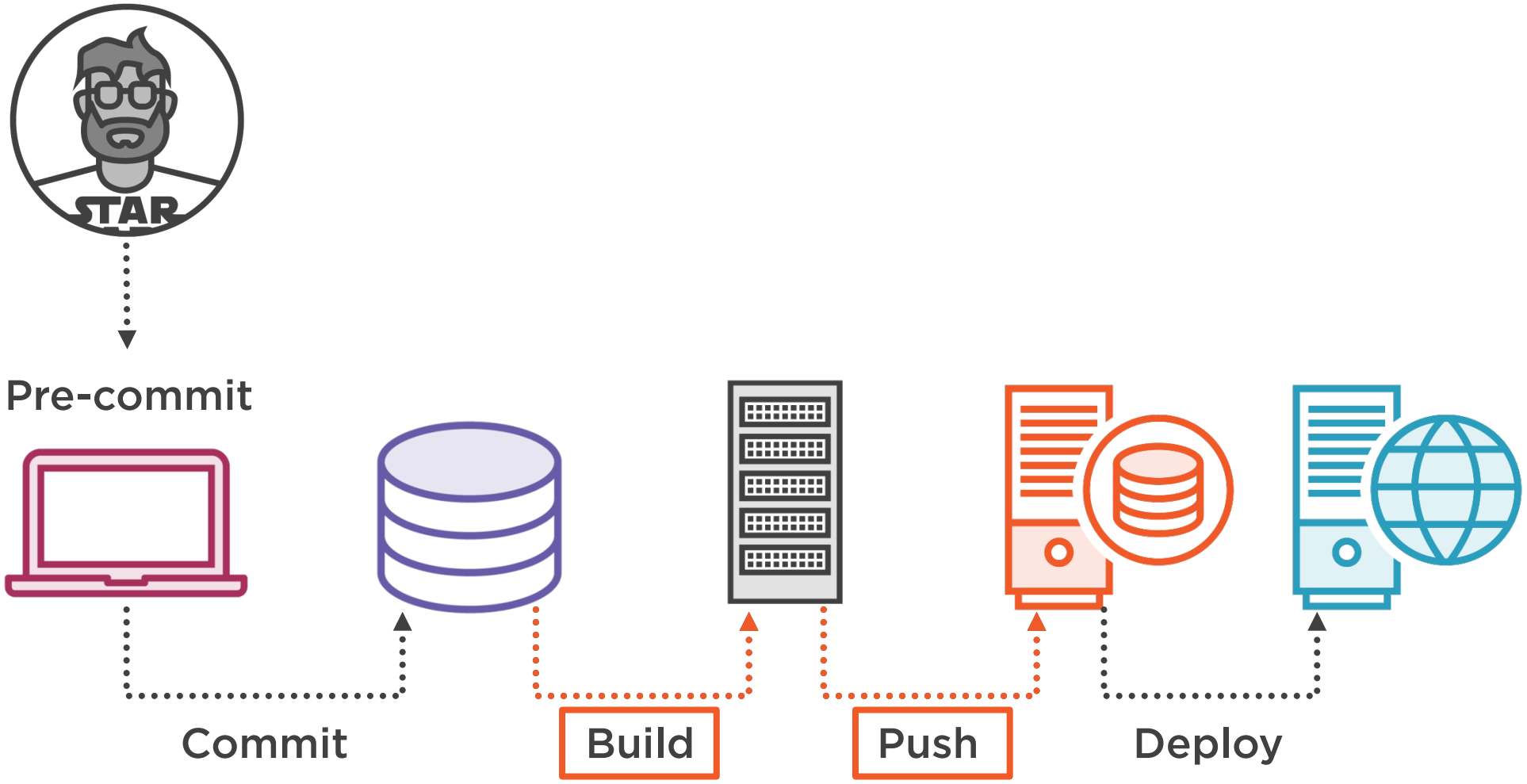
Generates overview of components in use

- Including version numbers

Generates alerts for outdated, insecure libraries

Can serve as an “asynchronous” quality gate

Where to Use Scanners



When to Scan

- Scans usually take a long time
- Results vary independently of code
- Recommended to perform periodic scans
- Scans during builds might “muddy” results

Tool That Will Be Demo-ed

OWASP Dependency-Check

- Attempts to detect published vulnerabilities in used libraries

Demo

Using OWASP Dependency-Check

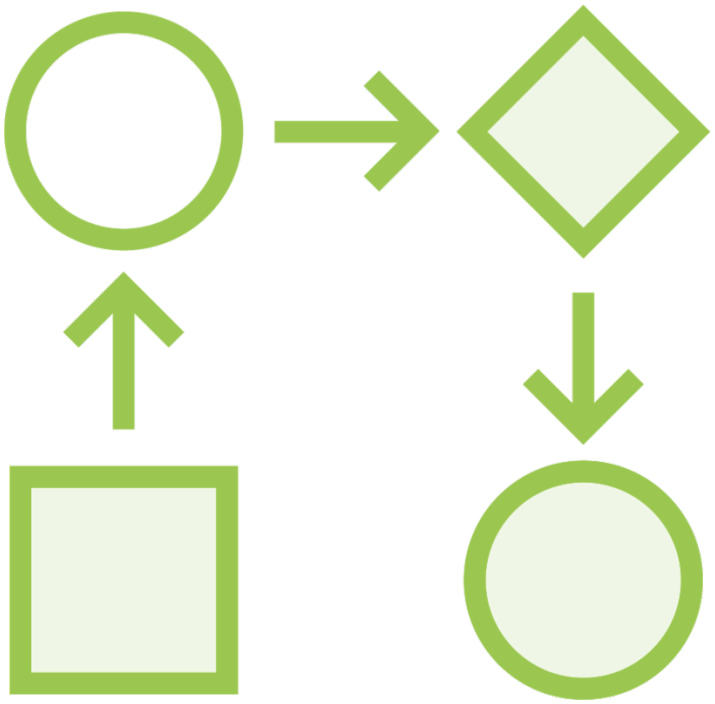
- Use Dependency-Check on command-line

Demo

Using OWASP Dependency-Check

- Use Dependency-Check in build pipeline

Workflow for Third-party Libraries Scanners



Schedule scanner

- Regularly audit the list of libraries
  - Does it recognize all libraries?

Ensure that alerts are generated for insecure libraries

Plan to perform updates when necessary

Third Party Libraries Scanner

Alerts about insecure and outdated libraries

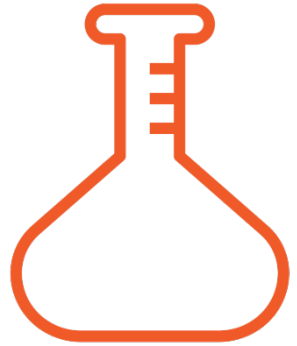
Generates overview of libraries in use

Advantage

Be actively alerted about insecure and outdated libraries

Compatibility

Depending on framework and language



Trialability

Moderately easy to employ in Continuous Integration pipelines

Follow-up can be time consuming

More Information

https://github.com/jeremylong/DependencyCheck

Summary

Security degrades over time

Run ‘asynchronous’ scheduled scans

Plan in time for updates and upgrades

Next Up

Maeve

"Very useful"

"I wonder though about the security of our containers.."

"Great observation!

"Let's test those in the next module"

Jennifer