VML

CONTINUOUS SECURITY: INTEGRATE SECURITY INTO YOUR DEVOPS PIPELINES

BILL DINGER
Solutions Architect | @adazlian
bill.dinger@vml.com

SECURITY LAPSES RUIN CUSTOMERS TRUST IN YOU

I am rugged and, more importantly, my code is rugged.
I recognize that software has become a foundation of our modern world.
I recognize the awesome responsibility that comes with this foundational role.
I recognize that my code will be used in ways I cannot anticipate, in ways it was not designed, and for longer than it was ever intended.
I recognize that my code will be attacked by talented and persistent adversaries who threaten our physical, economic, and national security.
I recognize these things - and I choose to be rugged.
I am rugged because I refuse to be a source of vulnerability or weakness.
I am rugged because I assure my code will support its mission.
I am rugged because my code can face these challenges and persist in spite of them.
I am rugged, not because it is easy, but because it is necessary and I am up for the challenge.

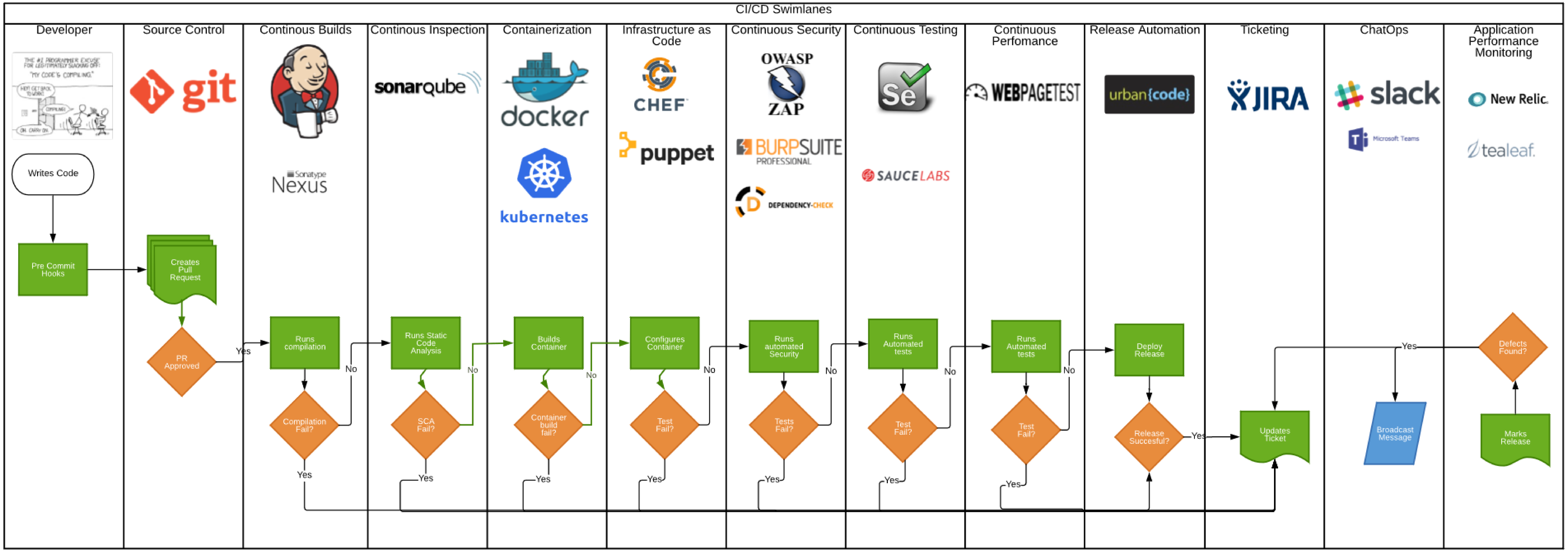
SHIFT LEFT: TEST EARLY & TEST OFTEN

“Another finding in our research is that teams that build security into their work also do better at continuous delivery […] When the tools provided actually make life easier for the engineers who use them, they will adopt them of their own free will.”

“We found that high performers were spending 50% less time remediating security issues than low performers. In other words, by building security into their daily work, as opposed to retrofitting security concerns at the end, they spent significantly less time addressing security issues.”

Accelerate – Nicole Forsgren, Jez Humble & Gene Kim

BRIDGING THE GAP



COUPLE ASSUMPTIONS...

Bitbucket

JavaScript

Jenkins

.NET

docker

Java™

OUR TOOLS

Code:

- Dependency Check
- Zed Attack proxy (ZAP)
- Open Web Testing Framework (OWTF)
- Web Testing Environments (WTE)
- Sonarqube Rules

Documentation:

- Application Security Verification Standard

OWASP DEPENDENCY CHECK

OWASP DEPENDENCY CHECK

Benefits

Protects against the OWASP #9 – using components with known security vulnerabilities automatically as part of your build and CI pipeline.

How it works

Scans dependencies and compares them against the NIST National Vulnerability Database (NVD) for any posted CVEs. Supports suppression lists, report generation, failure depending on severity, etc.

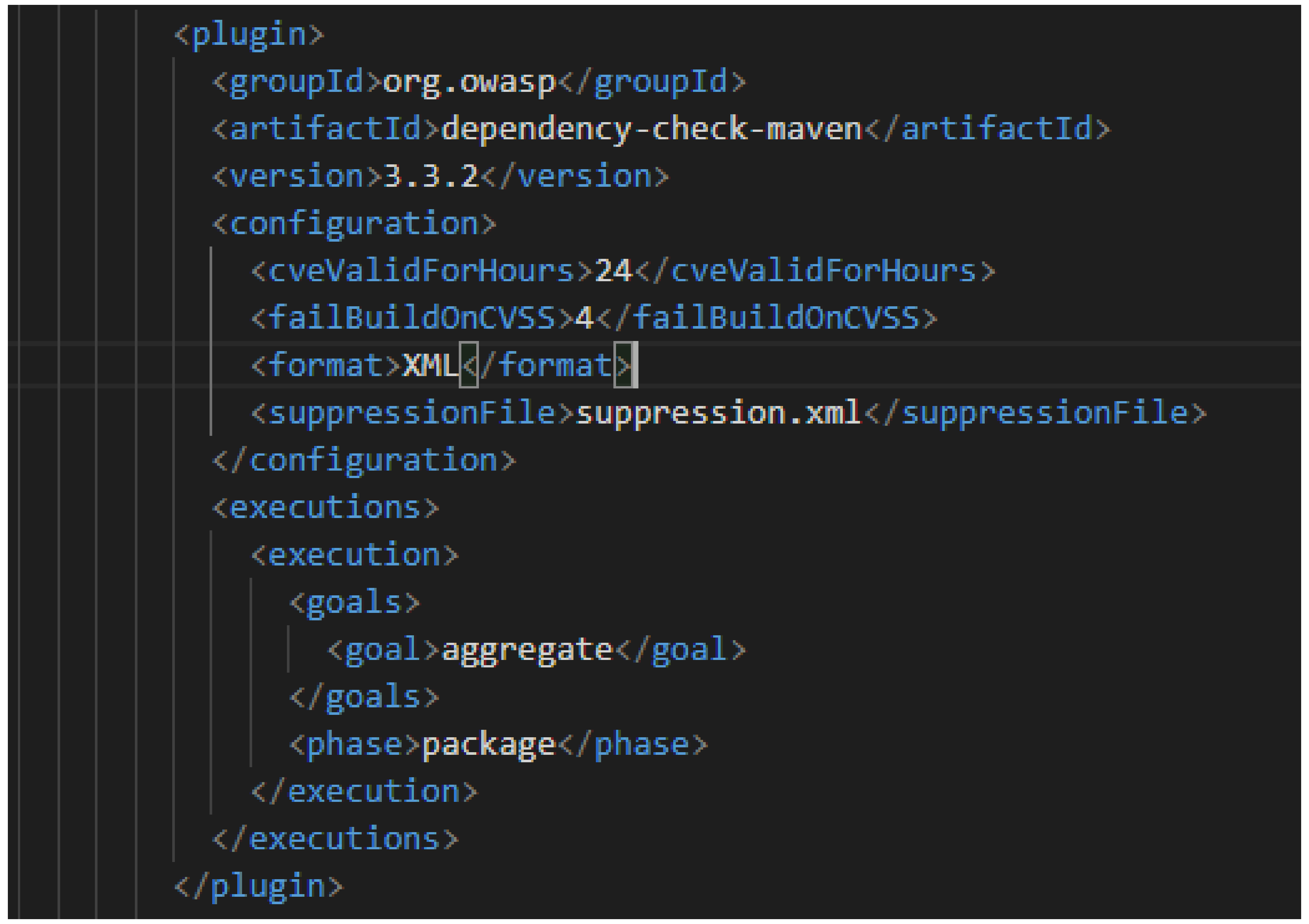
Implementations

- Java & .NET are officially supported.
- Python, Ruby, PHP, Node.JS, C/C++ experimental support.
- Command Line/Jenkins/Gradle/maven/Ant plugins and tooling all exist in various forms of maturity.

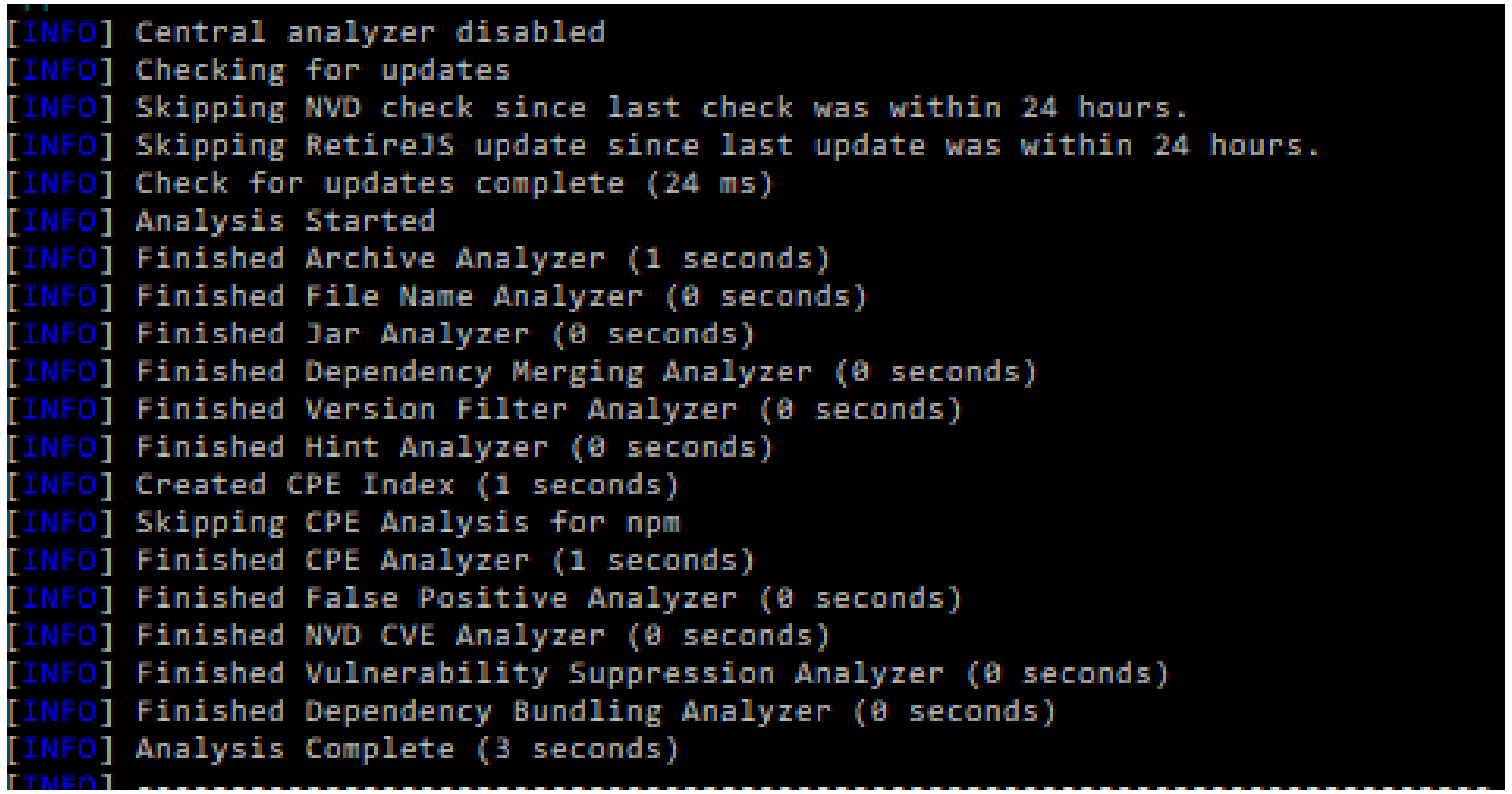
DEPENDENCY CHECK IN ACTION

Java (Maven)

```
mvn dependency-check:check
```



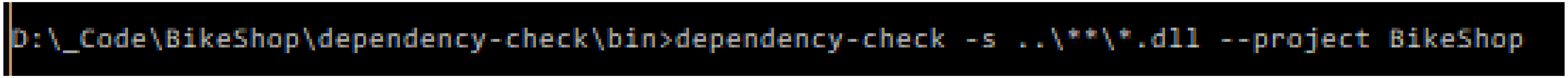
```
<plugin>
  <groupId>org.owasp</groupId>
  <artifactId>dependency-check-maven</artifactId>
  <version>3.3.2</version>
  <configuration>
    <cveValidForHours>24</cveValidForHours>
    <failBuildOnCVSS>4</failBuildOnCVSS>
    <format>XML</format>
    <suppressionFile>suppression.xml</suppressionFile>
  </configuration>
  <executions>
    <execution>
      <goals>
        <goal>aggregate</goal>
      </goals>
      <phase>package</phase>
    </execution>
  </executions>
</plugin>
```



```
[INFO] Central analyzer disabled
[INFO] Checking for updates
[INFO] Skipping NVD check since last check was within 24 hours.
[INFO] Skipping RetireJS update since last update was within 24 hours.
[INFO] Check for updates complete (24 ms)
[INFO] Analysis Started
[INFO] Finished Archive Analyzer (1 seconds)
[INFO] Finished File Name Analyzer (0 seconds)
[INFO] Finished Jar Analyzer (0 seconds)
[INFO] Finished Dependency Merging Analyzer (0 seconds)
[INFO] Finished Version Filter Analyzer (0 seconds)
[INFO] Finished Hint Analyzer (0 seconds)
[INFO] Created CPE Index (1 seconds)
[INFO] Skipping CPE Analysis for npm
[INFO] Finished CPE Analyzer (1 seconds)
[INFO] Finished False Positive Analyzer (0 seconds)
[INFO] Finished NVD CVE Analyzer (0 seconds)
[INFO] Finished Vulnerability Suppression Analyzer (0 seconds)
[INFO] Finished Dependency Bundling Analyzer (0 seconds)
[INFO] Analysis Complete (3 seconds)
```

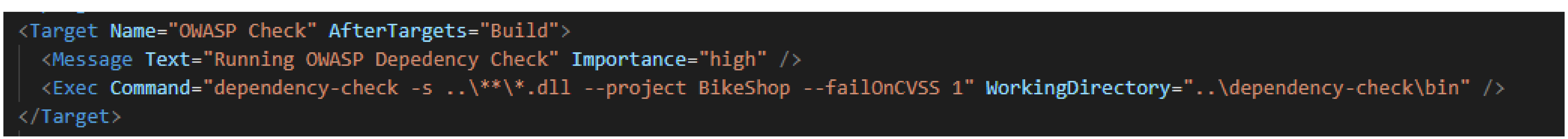
DEPENDENCY CHECK IN ACTION

Command Line



```
D:\_Code\BikeShop\dependency-check\bin>dependency-check -s ..\**\*.dll --project BikeShop
```

MSBuild Task

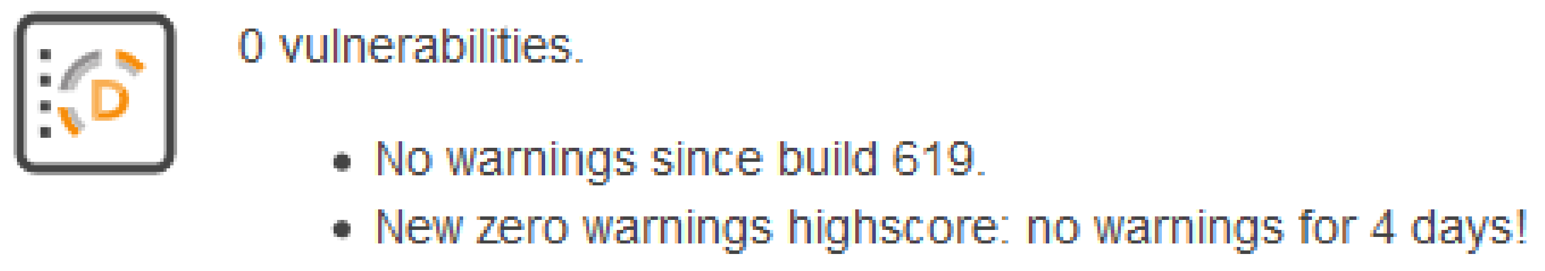


```
<Target Name="OWASP Check" AfterTargets="Build">
  <Message Text="Running OWASP Depedency Check" Importance="high" />
  <Exec Command="dependency-check -s ..\**\*.dll --project BikeShop --failOnCVSS 1" WorkingDirectory="..\dependency-check\bin" />
</Target>
```

https://jeremylong.github.io/DependencyCheck/dependency-check-cli/arguments.html

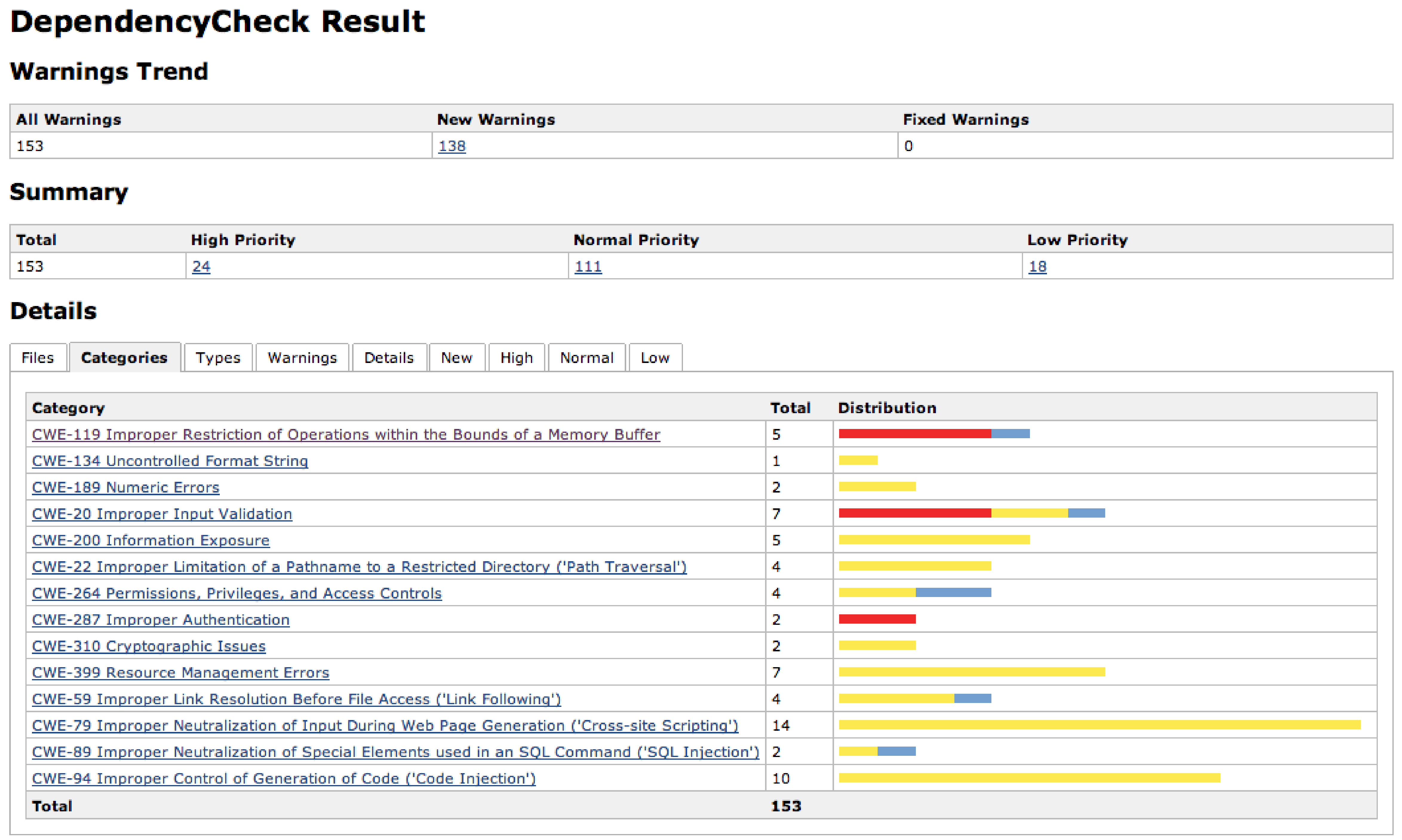
DEPENDENCY CHECK IN ACTION

Jenkins Integration



0 vulnerabilities.

- No warnings since build 619.
- New zero warnings highscore: no warnings for 4 days!



DependencyCheck Result

Warnings Trend

All Warnings	New Warnings	Fixed Warnings
153	138	0

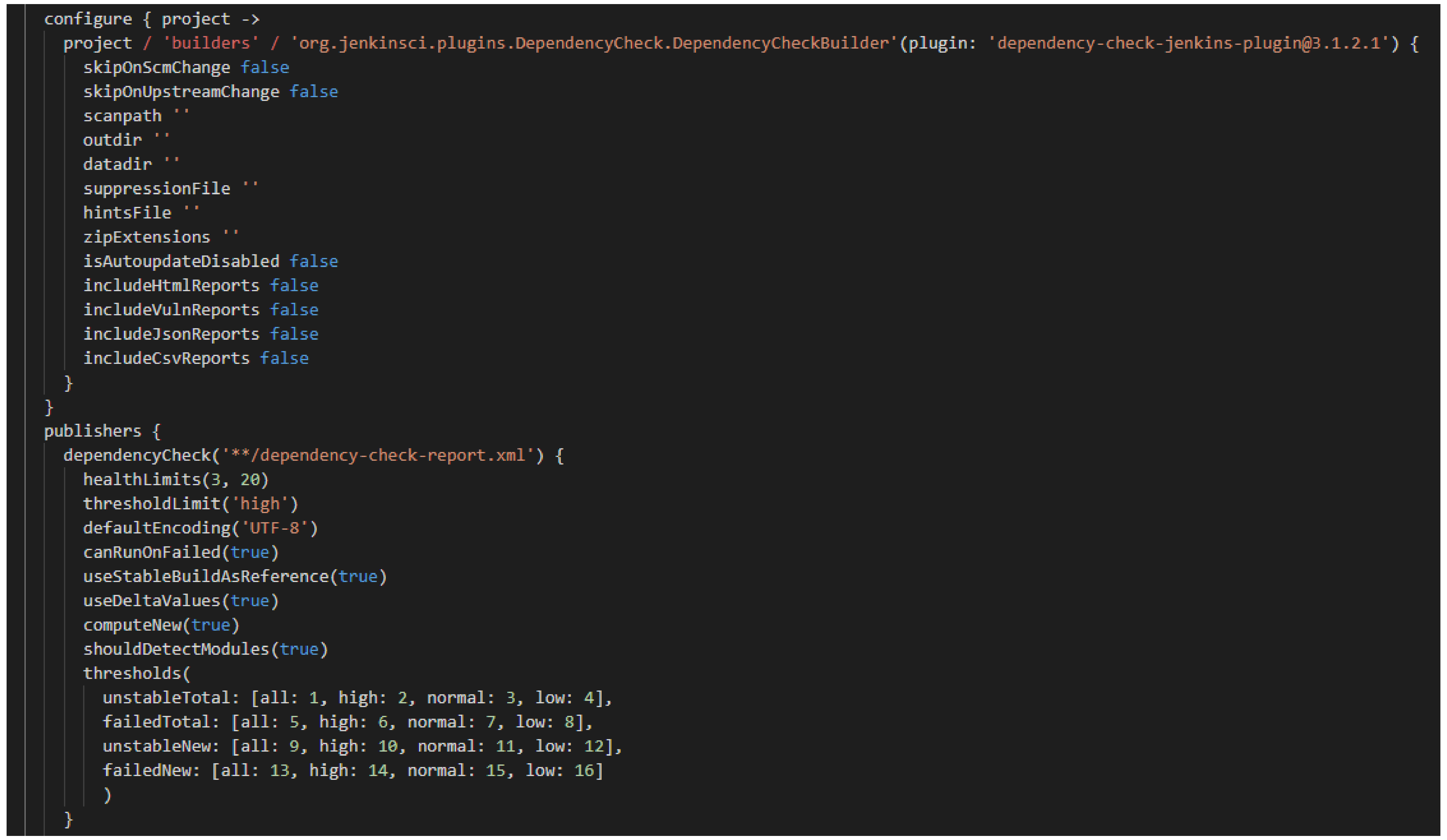
Summary

Total	High Priority	Normal Priority	Low Priority
153	24	111	18

Details

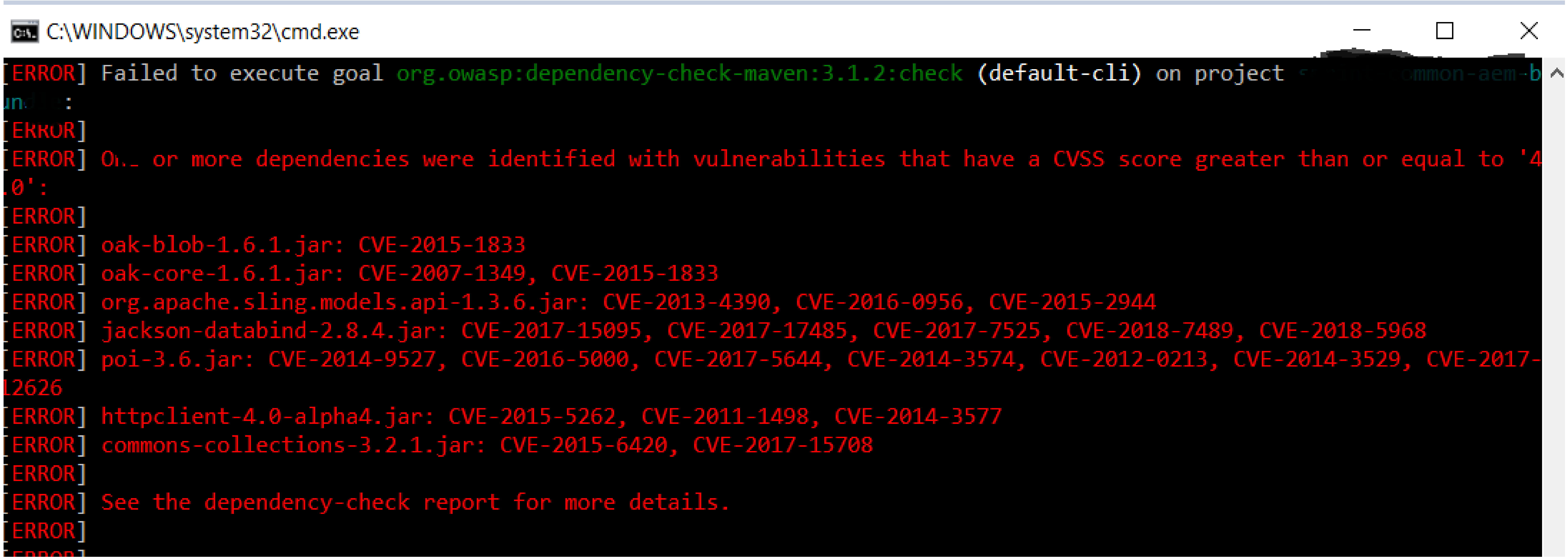
Files | **Categories** | Types | Warnings | Details | New | High | Normal | Low

Category	Total	Distribution
CWE-119 Improper Restriction of Operations within the Bounds of a Memory Buffer	5	🟥🟥🟥🟥🟦
CWE-134 Uncontrolled Format String	1	🟨
CWE-189 Numeric Errors	2	🟨🟨
CWE-20 Improper Input Validation	7	🟥🟥🟥🟥🟨🟨🟦
CWE-200 Information Exposure	5	🟨🟨🟨🟨🟨
CWE-22 Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	4	🟨🟨🟨🟨
CWE-264 Permissions, Privileges, and Access Controls	4	🟨🟨🟦🟦
CWE-287 Improper Authentication	2	🟥🟥
CWE-310 Cryptographic Issues	2	🟨🟨
CWE-399 Resource Management Errors	7	🟨🟨🟨🟨🟨🟨🟨
CWE-59 Improper Link Resolution Before File Access ('Link Following')	4	🟨🟨🟨🟦
CWE-79 Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	14	🟨🟨🟨🟨🟨🟨🟨🟨🟨🟨🟨🟨🟨🟨
CWE-89 Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	2	🟨🟦
CWE-94 Improper Control of Generation of Code ('Code Injection')	10	🟨🟨🟨🟨🟨🟨🟨🟨🟨🟨
Total	**153**	



```
configure { project ->
  project / 'builders' / 'org.jenkinsci.plugins.DependencyCheck.DependencyCheckBuilder'(plugin: 'dependency-check-jenkins-plugin@3.1.2.1') {
    skipOnScmChange false
    skipOnUpstreamChange false
    scanpath ''
    outdir ''
    datadir ''
    suppressionFile ''
    hintsFile ''
    zipExtensions ''
    isAutoupdateDisabled false
    includeHtmlReports false
    includeVulnReports false
    includeJsonReports false
    includeCsvReports false
  }
}
publishers {
  dependencyCheck('**/dependency-check-report.xml') {
    healthLimits(3, 20)
    thresholdLimit('high')
    defaultEncoding('UTF-8')
    canRunOnFailed(true)
    useStableBuildAsReference(true)
    useDeltaValues(true)
    computeNew(true)
    shouldDetectModules(true)
    thresholds(
      unstableTotal: [all: 1, high: 2, normal: 3, low: 4],
      failedTotal: [all: 5, high: 6, normal: 7, low: 8],
      unstableNew: [all: 9, high: 10, normal: 11, low: 12],
      failedNew: [all: 13, high: 14, normal: 15, low: 16]
      )
  }
```

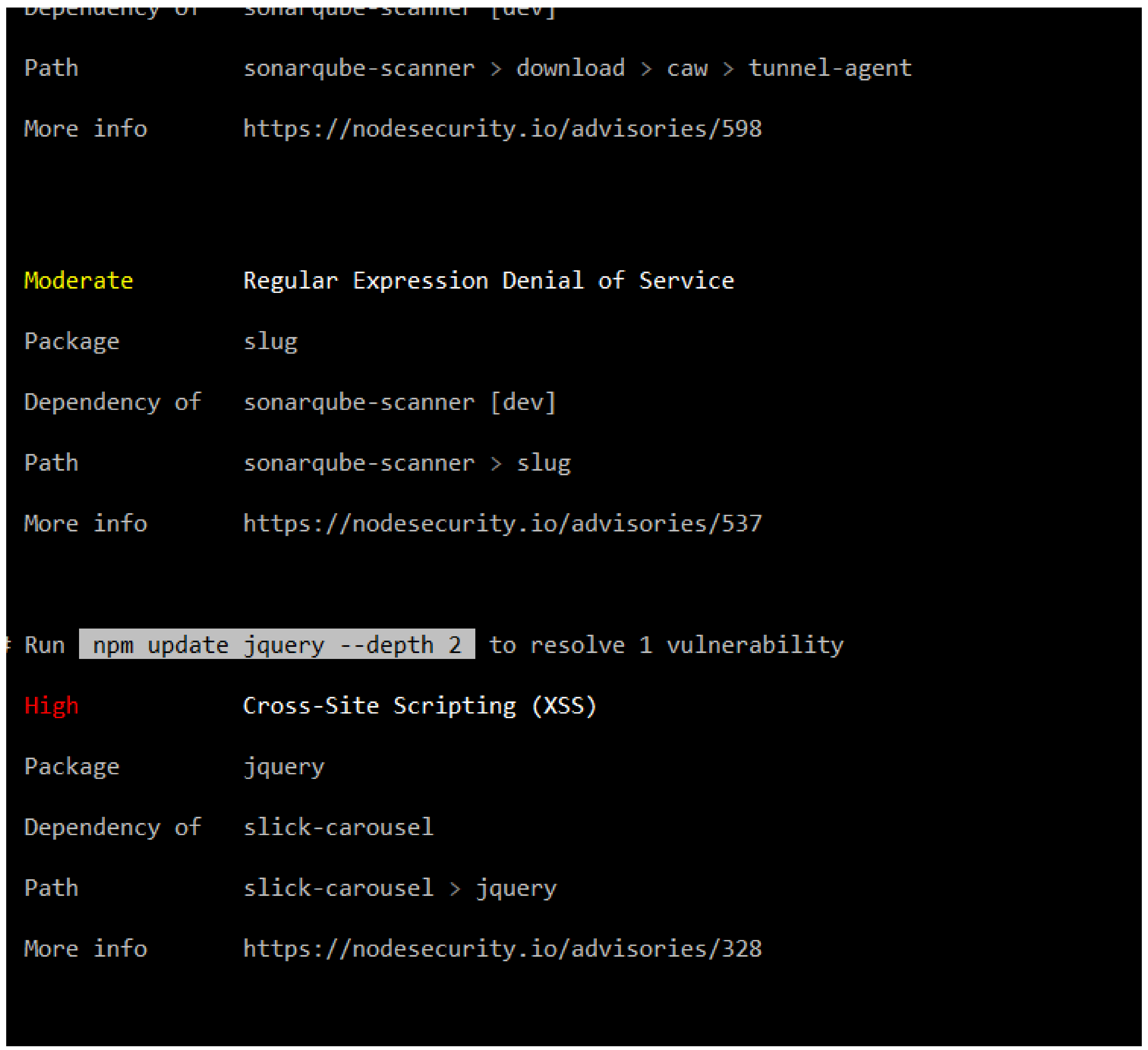
HOW BAD CAN IT BE?



C:\WINDOWS\system32\cmd.exe

```
[ERROR] Failed to execute goal org.owasp:dependency-check-maven:3.1.2:check (default-cli) on project [redacted]
[redacted]:
[ERROR]
[ERROR] One or more dependencies were identified with vulnerabilities that have a CVSS score greater than or equal to '4
.0':
[ERROR]
[ERROR] oak-blob-1.6.1.jar: CVE-2015-1833
[ERROR] oak-core-1.6.1.jar: CVE-2007-1349, CVE-2015-1833
[ERROR] org.apache.sling.models.api-1.3.6.jar: CVE-2013-4390, CVE-2016-0956, CVE-2015-2944
[ERROR] jackson-databind-2.8.4.jar: CVE-2017-15095, CVE-2017-17485, CVE-2017-7525, CVE-2018-7489, CVE-2018-5968
[ERROR] poi-3.6.jar: CVE-2014-9527, CVE-2016-5000, CVE-2017-5644, CVE-2014-3574, CVE-2012-0213, CVE-2014-3529, CVE-2017-
12626
[ERROR] httpclient-4.0-alpha4.jar: CVE-2015-5262, CVE-2011-1498, CVE-2014-3577
[ERROR] commons-collections-3.2.1.jar: CVE-2015-6420, CVE-2017-15708
[ERROR]
[ERROR] See the dependency-check report for more details.
[ERROR]
```

SPECIAL BONUS TOOL: NPM AUDIT!



```
Dependency of   sonarqube-scanner [dev]

Path            sonarqube-scanner > download > caw > tunnel-agent

More info       https://nodesecurity.io/advisories/598



Moderate        Regular Expression Denial of Service

Package         slug

Dependency of   sonarqube-scanner [dev]

Path            sonarqube-scanner > slug

More info       https://nodesecurity.io/advisories/537


Run  npm update jquery --depth 2  to resolve 1 vulnerability

High            Cross-Site Scripting (XSS)

Package         jquery

Dependency of   slick-carousel

Path            slick-carousel > jquery

More info       https://nodesecurity.io/advisories/328
```

OWASP SONARQUBE RULES

OWASP SONARQUBE RULES

Benefits

A number of “suggested” rulesets to add to your static code analysis to catch common security issues before they can ever be merged in.

How it works

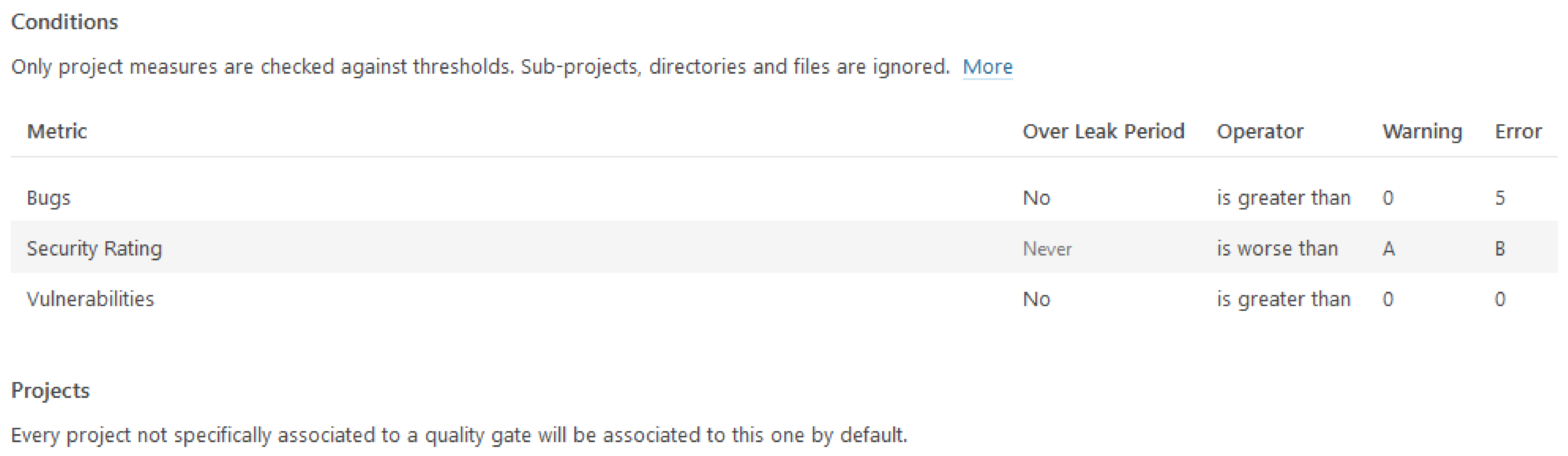
Integrates into Sonarqube natively as a ruleset. Performs static code analysis to see if the code violates any of the rules and flags as necessary. If running SonarQube as part of your build pipeline (and if you aren’t…) it’ll raise violations as critical allowing you to fail builds or fail merge checks.

Implementations

- Java/Groovy
- Sonarqube itself ships with a number of rulesets for Javascript/Typescript/Ruby/Python/PHP/Swift/C#/etc

SONARQUBE RULES IN ACTION

Quality Gates, in line reporting, corrective action



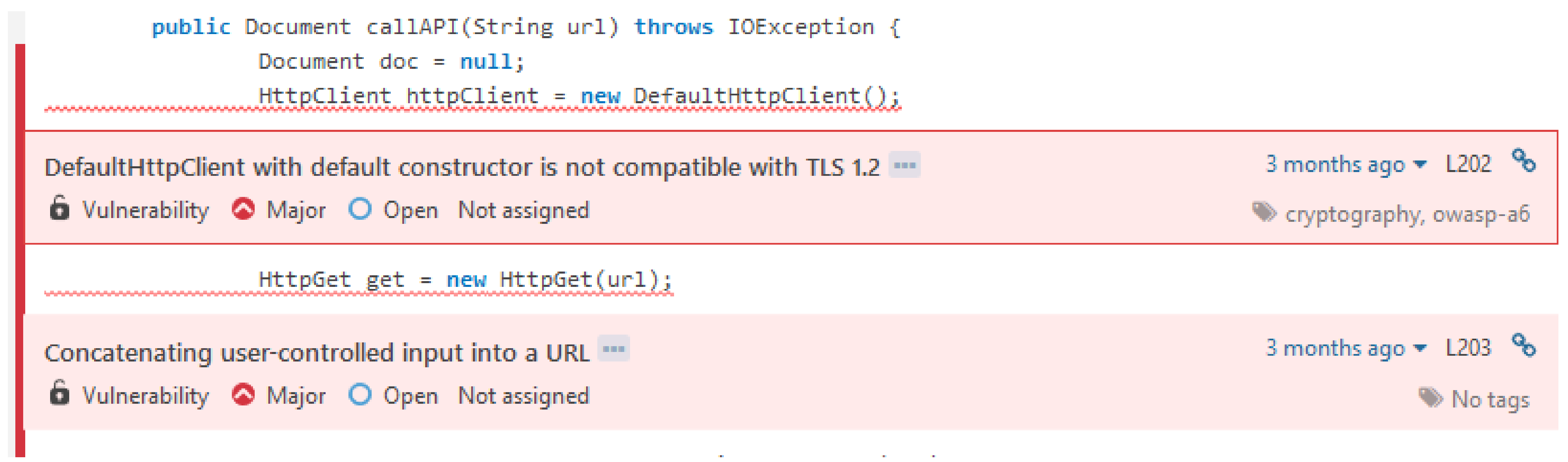
Conditions

Only project measures are checked against thresholds. Sub-projects, directories and files are ignored. More

Metric	Over Leak Period	Operator	Warning	Error
Bugs	No	is greater than	0	5
Security Rating	Never	is worse than	A	B
Vulnerabilities	No	is greater than	0	0

Projects

Every project not specifically associated to a quality gate will be associated to this one by default.

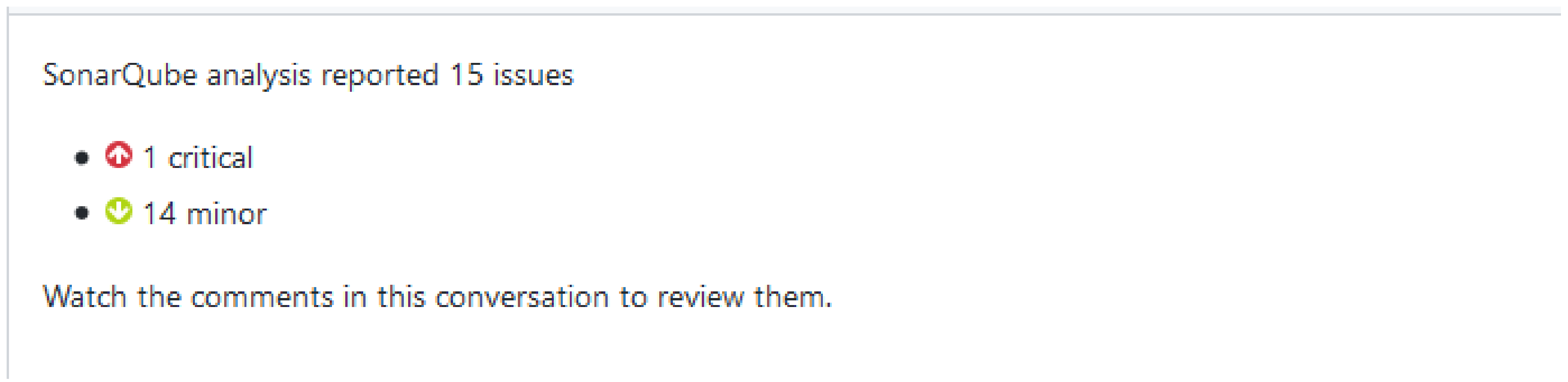


```
        public Document callAPI(String url) throws IOException {
                Document doc = null;
                HttpClient httpClient = new DefaultHttpClient();
```

DefaultHttpClient with default constructor is not compatible with TLS 1.2 ··· 3 months ago ▾ L202
Vulnerability · Major · Open · Not assigned · cryptography, owasp-a6

```
                HttpGet get = new HttpGet(url);
```

Concatenating user-controlled input into a URL ··· 3 months ago ▾ L203
Vulnerability · Major · Open · Not assigned · No tags



SonarQube analysis reported 15 issues

- 1 critical
- 14 minor

Watch the comments in this conversation to review them.

ZAP ATTACK PROXY

ZAP ATTACK PROXY

Benefits

Automatically hits your site/server probing for vulnerabilities.

How it works

Scans your machine with a configurable set of rules looking for common vulnerabilities and outputs a report. Looks for all the common OWASP top 10 violations as well as misconfigured machines, etc.

Implementations

Has a large, customizable rule set as well as integration points with Jenkins and prebuilt docker images as well as an official Jenkins plugin.

ZAP OPTIONS

GUI – Java based, uses SWING

CLI – Command line interface

REST - API

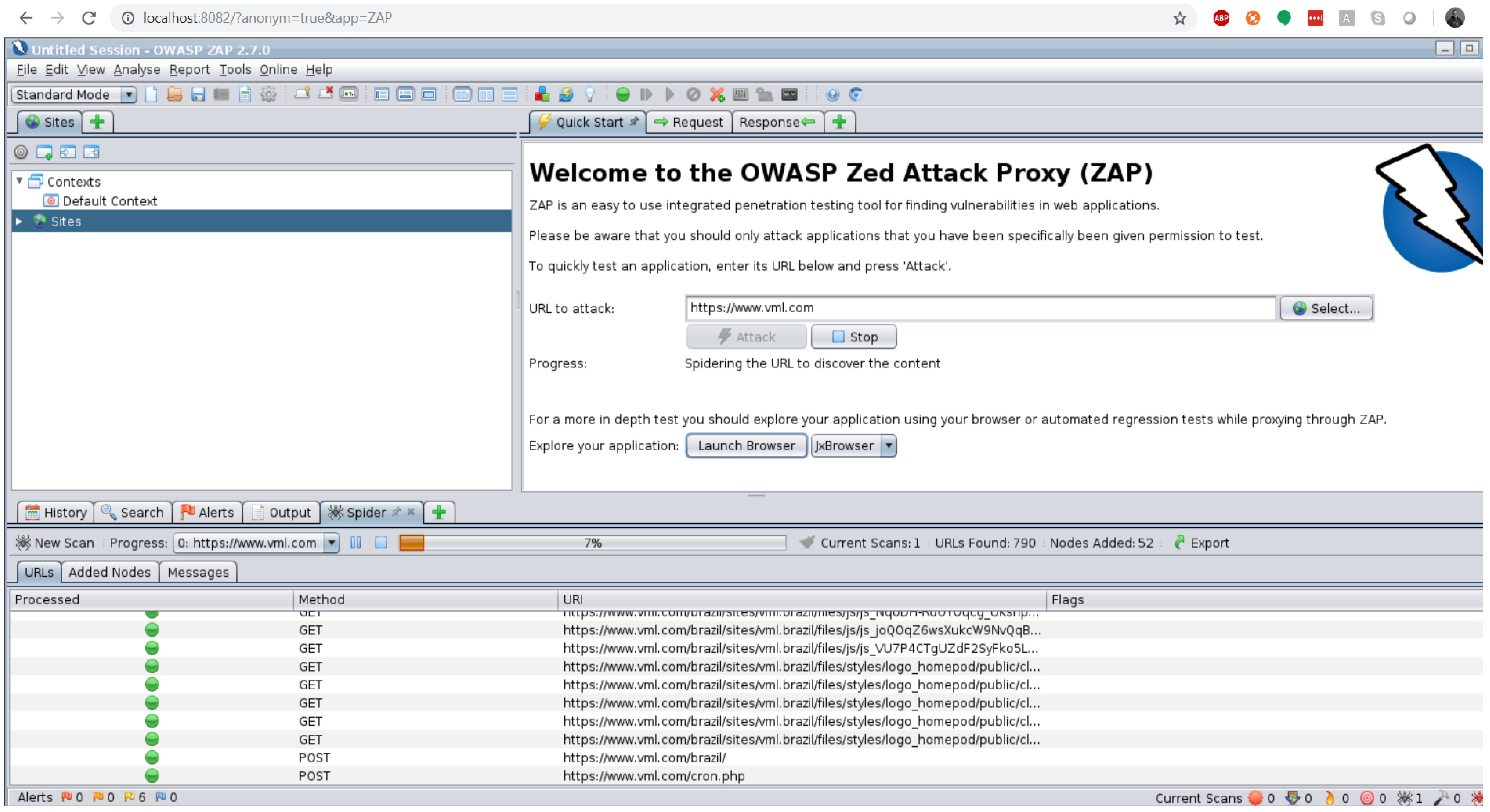
Jenkins Plugin

ZAP IN ACTION

GUI

There is a GUI available to run a "traditional" scan but this Isn't very scriptable.

Great for exploratory testing and figuring out your options.


localhost:8082/?anonym=true&app=ZAP
Untitled Session - OWASP ZAP 2.7.0
File Edit View Analyse Report Tools Online Help
Standard Mode
Sites
Quick Start
Request
Response
Contexts
Default Context
Sites

Welcome to the OWASP Zed Attack Proxy (ZAP)

ZAP is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications.

Please be aware that you should only attack applications that you have been specifically been given permission to test.

To quickly test an application, enter its URL below and press 'Attack'.

URL to attack: https://www.vml.com [Select...]

[Attack] [Stop]

Progress: Spidering the URL to discover the content

For a more in depth test you should explore your application using your browser or automated regression tests while proxying through ZAP.

Explore your application: [Launch Browser] [JxBrowser]

History | Search | Alerts | Output | Spider

New Scan Progress: 0: https://www.vml.com 7% Current Scans: 1 URLs Found: 790 Nodes Added: 52 Export

URLs | Added Nodes | Messages

Processed	Method	URI	Flags
●	GET	https://www.vml.com/brazil/sites/vml.brazil/files/js/js_NqoDH-RdOYOqcg_OKsnp...	
●	GET	https://www.vml.com/brazil/sites/vml.brazil/files/js/js_joQ0qZ6wsXukcW9NvQqB...	
●	GET	https://www.vml.com/brazil/sites/vml.brazil/files/js/js_VU7P4CTgUZdF2SyFko5L...	
●	GET	https://www.vml.com/brazil/sites/vml.brazil/files/styles/logo_homepod/public/cl...	
●	GET	https://www.vml.com/brazil/sites/vml.brazil/files/styles/logo_homepod/public/cl...	
●	GET	https://www.vml.com/brazil/sites/vml.brazil/files/styles/logo_homepod/public/cl...	
●	GET	https://www.vml.com/brazil/sites/vml.brazil/files/styles/logo_homepod/public/cl...	
●	GET	https://www.vml.com/brazil/sites/vml.brazil/files/styles/logo_homepod/public/cl...	
●	POST	https://www.vml.com/brazil/	
●	POST	https://www.vml.com/cron.php	

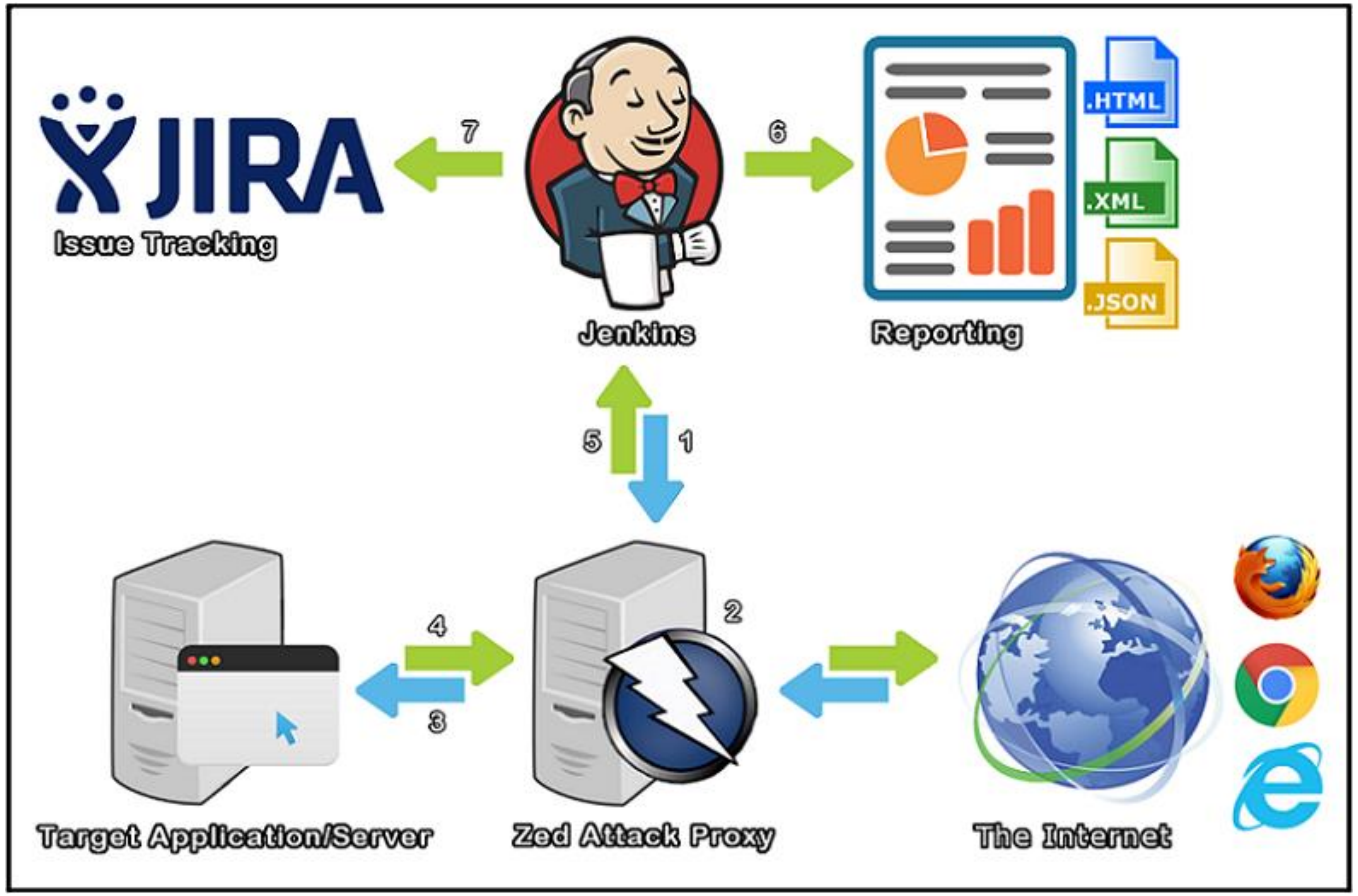
Alerts 0 0 6 0 Current Scans 0 0 0 0 1 0

ZAP IN ACTION: JENKINS PLUGIN

Also an official Jenkins plugin.

Has Integration with JIRA.

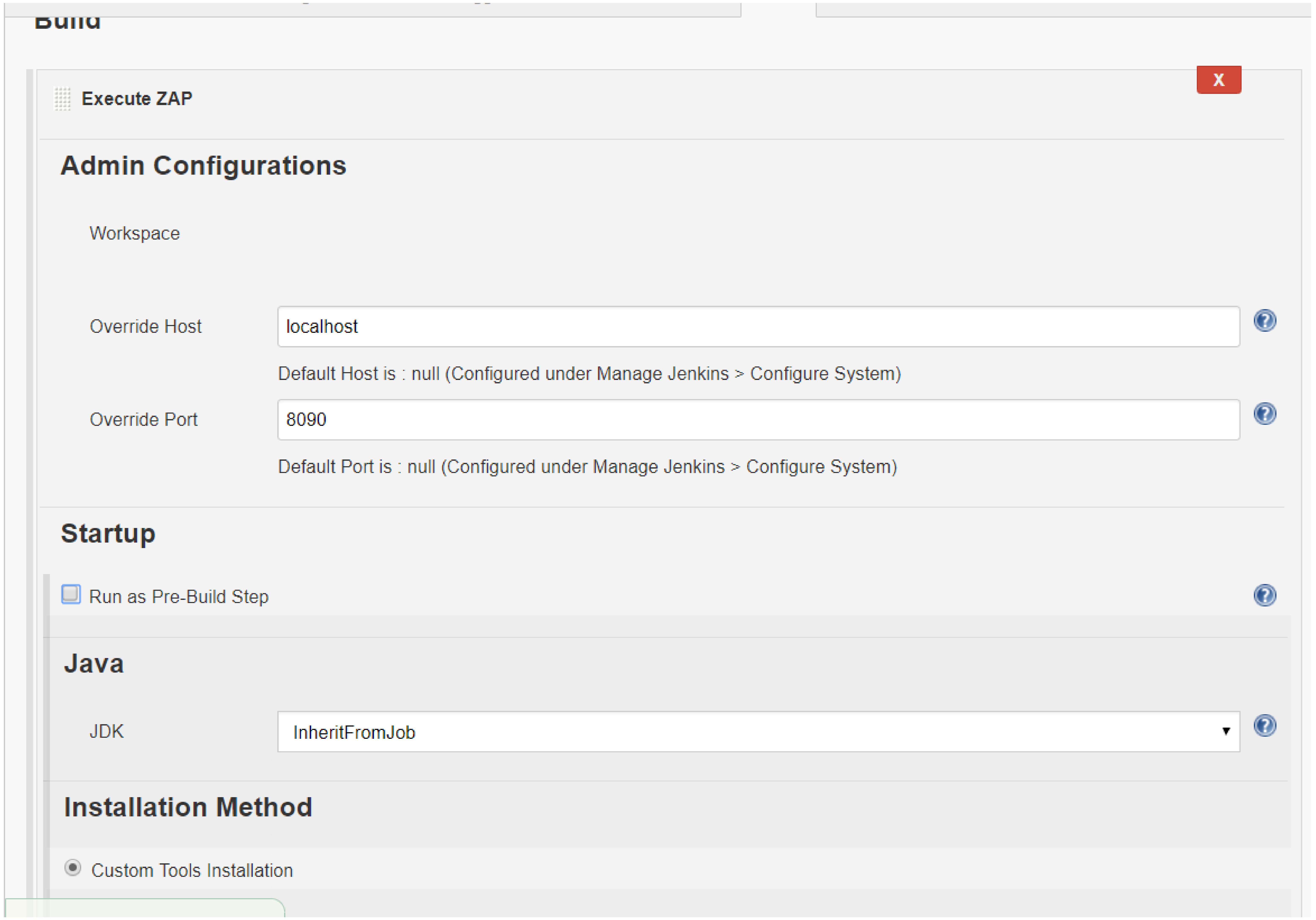
Requires full install of OWASP ZAP.



ZAP IN ACTION: JENKINS PLUGIN

Drawbacks

Great integration, but requires owasp zap server to be running on same container as Jenkins OR have Jenkins slave installed on your owasp container.



Build

Execute ZAP

Admin Configurations

Workspace

Override Host: localhost

Default Host is : null (Configured under Manage Jenkins > Configure System)

Override Port: 8090

Default Port is : null (Configured under Manage Jenkins > Configure System)

Startup

☐ Run as Pre-Build Step

Java

JDK: InheritFromJob

Installation Method

● Custom Tools Installation

ZAP IN ACTION: ZAP CLI

Quick Scan

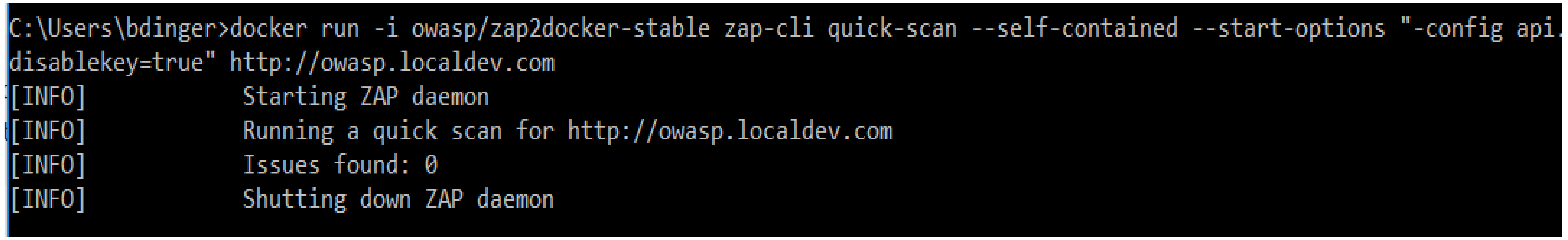
Runs a very basic scan, limited to one minute

Baseline

Completely safe, usually very quick.

Active Scan

Not safe, potentially legal / production impacting.



```
C:\Users\bdinger>docker run -i owasp/zap2docker-stable zap-cli quick-scan --self-contained --start-options "-config api.
disablekey=true" http://owasp.localdev.com
[INFO]            Starting ZAP daemon
[INFO]            Running a quick scan for http://owasp.localdev.com
[INFO]            Issues found: 0
[INFO]            Shutting down ZAP daemon
```

ZAP IN ACTION: ZAP-API

Invoke VIA API

ZAP api provides a rich feature set.

Best use case is in docker container setups.

Suggestion is to chain to your Jenkins instance and invoke

ZED ATTACK PROXY: TIPS

Run as the last stage of your pipeline

ZAP requires that the code actually be running before it can be tested like automated QA tests.

Or as a schedule task

In case the complexity of a build being deployed, then being failed back, is too risky for your environment you can instead run it on a set schedule and use the results of that to inform deployment decisions.

Lots of Resources

Zap consumes major resources running and will easily bring a machine to its knees. Provision it well and (suggested) to use a docker container and scale/destroy as needed. Don’t run it on same box as your Jenkins machine!

OPEN WEB TESTING FRAMEWORK

OPEN WEB TESTING FRAMEWORK

Benefits

An active pen testers framework.

How it works

Provides a prebuilt environment with a number of configurable plugins. Designed to help your active penetration testers run their security tests against your site. Note this differs from ZAP (Zed Attack Proxy) which is *solely* an automated attack proxy.

Implementations

Currently implemented and distributed as alive cd/ docker image / homebrew package

WEB TESTING ENVIRONMENTS

WEB TESTING ENVIRONMENT

Benefits

A prebuilt image (or series of packages) with common AppSec tools built into them.

How it works

OWASP supplies a few images and packages with a number of popular security testing tools – like nmap – already built in. This allows you to grab a stock, standard prebuilt image that can be spun up and either further automated or simply handed off to your security testing team.

Implementations

Besides Debian packages also includes Vmware, Virtuablebox and parallels images as well as a docker image (soon)

WTE: IN ACTION

Automated with run commands

Stable docker image that can run nmap, etc script commands via docker or Jenkins.

Spun up to evaluate environments

Start up and provision as part of pipeline to be handed off to penetration testing team

APPLICATION SECURITY VERIFICATION STANDARDS

ASVS

Benefits

A guidebook to securing your application stack.

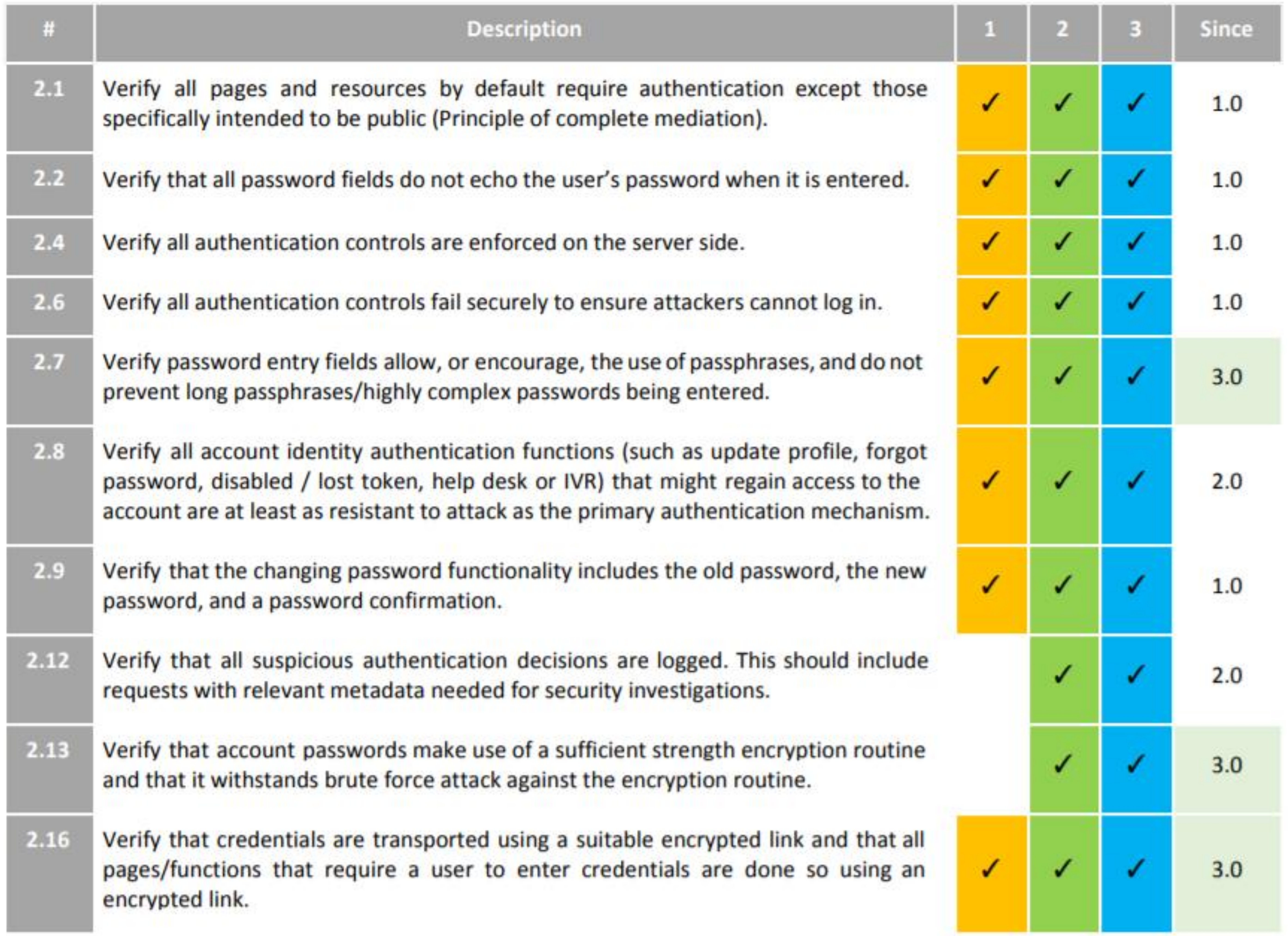
How it works

A series of checklists and procedures covering the whole gamut of security to ensure you're critical infrastructure (like your devops pipelines) is fully secure.

Implementations

Documentation available on OWASPs site

ASVS EXAMPLE AUDIT



#	Description	1	2	3	Since
2.1	Verify all pages and resources by default require authentication except those specifically intended to be public (Principle of complete mediation).	✓	✓	✓	1.0
2.2	Verify that all password fields do not echo the user's password when it is entered.	✓	✓	✓	1.0
2.4	Verify all authentication controls are enforced on the server side.	✓	✓	✓	1.0
2.6	Verify all authentication controls fail securely to ensure attackers cannot log in.	✓	✓	✓	1.0
2.7	Verify password entry fields allow, or encourage, the use of passphrases, and do not prevent long passphrases/highly complex passwords being entered.	✓	✓	✓	3.0
2.8	Verify all account identity authentication functions (such as update profile, forgot password, disabled / lost token, help desk or IVR) that might regain access to the account are at least as resistant to attack as the primary authentication mechanism.	✓	✓	✓	2.0
2.9	Verify that the changing password functionality includes the old password, the new password, and a password confirmation.	✓	✓	✓	1.0
2.12	Verify that all suspicious authentication decisions are logged. This should include requests with relevant metadata needed for security investigations.		✓	✓	2.0
2.13	Verify that account passwords make use of a sufficient strength encryption routine and that it withstands brute force attack against the encryption routine.		✓	✓	3.0
2.16	Verify that credentials are transported using a suitable encrypted link and that all pages/functions that require a user to enter credentials are done so using an encrypted link.	✓	✓	✓	3.0

REMEMBER: IF YOUR CI/CD STACK IS COMPROMISED YOU'VE LOST EFFECTIVE CONTROL OF ALL APPS

GO FORTH AND BUILD RUGGED SOFTWARE

RESOURCES

OWASP Top 10 Project:
https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

OWASP Cheat Sheet Series:
https://www.owasp.org/index.php/OWASP_Cheat_Sheet_Series

OWASP Docker Hub:
https://hub.docker.com/u/owasp/

Continuous Deployment with Docker & Jenkins:
http://vfarcic.github.io/jenkins-swarm/index.html

OWASP Dependency Check:
https://jeremylong.github.io/DependencyCheck/index.html

Docker & OWASP Setup:
https://blog.mozilla.org/fxtesteng/2016/05/11/docker-owasp-zap-part-one/

OWASP Zap automated scan list based on Swagger definitions
https://www.nearform.com/blog/zed-attack-proxy-in-a-ci-pipeline/

THANK YOU.

VML