

CS 305 Integrating the Maven Dependency Check Plug-in Tutorial

This course uses Maven, which is a package manager. You have the option of running the dependency check as a standalone application or as a part of a Maven project. This tutorial shows you how to run a dependency check as a Maven project. Follow these steps to integrate the dependency check plug-in.

1. Go to [Dependency-Check Maven](#). Refer to the example provided on the webpage and as shown below:

Example 1:

Create the dependency-check-report.html in the target directory.

```
<project>
  ...
  <build>
    ...
    <plugins>
      ...
      <plugin>
        <groupId>org.owasp</groupId>
        <artifactId>dependency-check-maven</artifactId>
        <version>5.3.0</version>
        <executions>
          <execution>
            <goals>
              <goal>check</goal>
            </goals>
          </execution>
        </executions>
      </plugin>
      ...
    </plugins>
    ...
  </build>
  ...
</project>
```

2. Identify the current version number of the dependency check. You will need this information for Step 3.



[OWASP](#) / [Dependency-Check](#) / [documentation](#) / [dependency-check](#) / Usage

Version: 5.3.0 Last Published: 2020-01-15



3. Next, upload the Maven project you would like to complete a static test for in Eclipse.
 - a. Open the pom.xml file to add the dependency check plug-in.

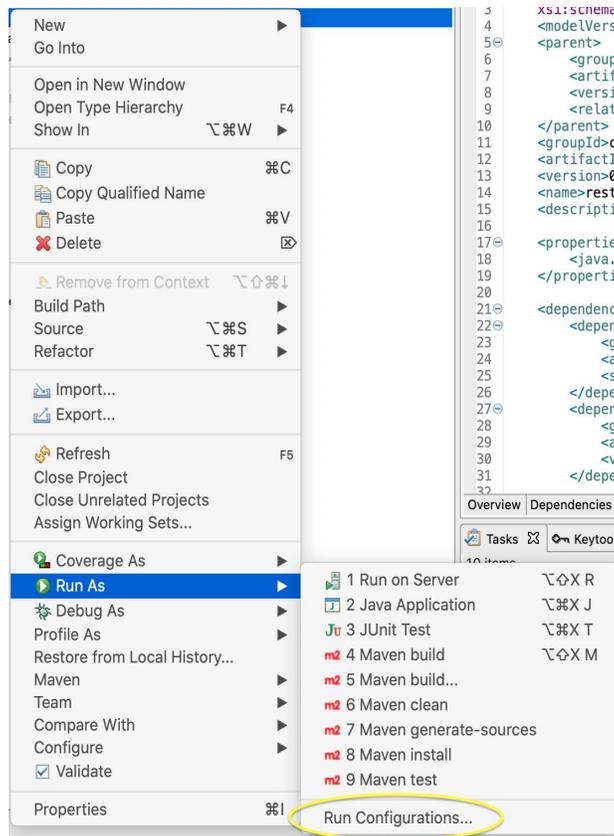
- b. Copy the following lines of code and paste the code into the pom.xml file. Be sure the version number is current based on your findings in Step 2.

```
<plug-in>
  <groupId>org.owasp</groupId>
  <artifactId>dependency-check-maven</artifactId>
  <version>5.3.0</version>
  <executions>
    <execution>
      <goals>
        <goal>check</goal>
      </goals>
    </execution>
  </executions>
</plug-in>
```

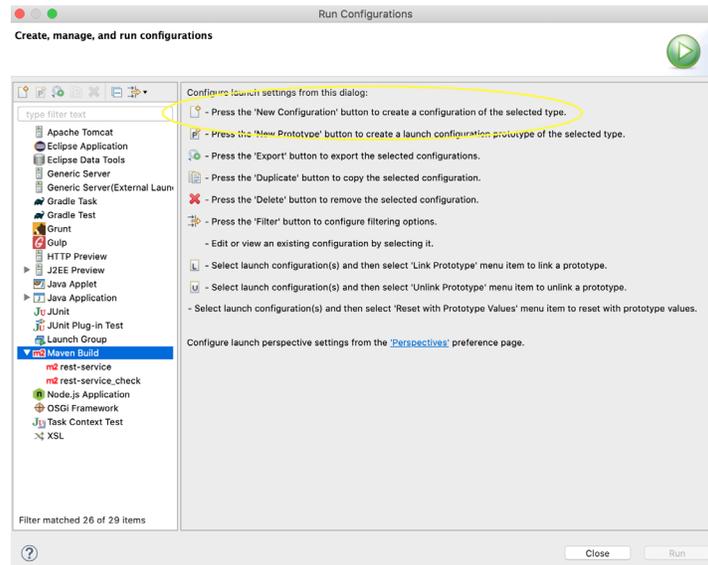
Please note: To use the dependency check plug-in, you will need to complete these steps for each code base or software application.

4. When you compile your code, Eclipse will run the Maven plug-in. Run the pom.xml file to ensure the plug-in is running effectively.

- a. On the menu bar, click **Run, Run As, and Run Configurations.**

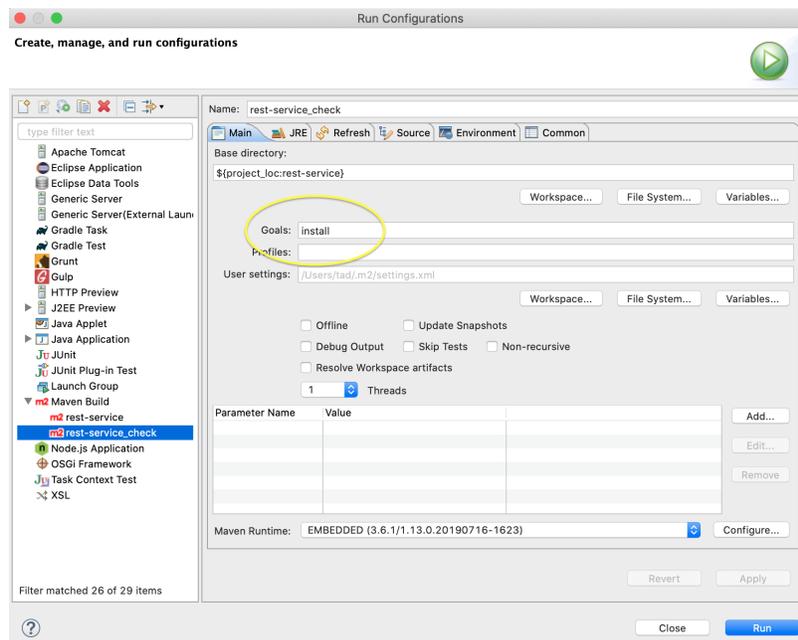


- b. In the Run Configurations window in Eclipse, double-click on **Maven Build**.



- i. Choose the Base directory if it is not already present by clicking **Workspace**, then select the appropriate directory.

- c. Set goals to “install”, then click **Run**.



- d. Be sure to observe the **Console** for dependency check execution. The first time you do this, it will require more time to download the Common Vulnerabilities and Exposures (CVE) from the National Vulnerabilities Database (NVD).

```

rest-service_check [Maven Build] /Library/Java/JavaVirtualMachines/jdk1.8.0_161.jdk/Contents/Home/bin/java (Feb 16, 2020, 11:53:27 AM)
[INFO] Using standard serializer [org.apache.commons.jcs.utils.serialization.StandardSerializer@47248a48] for auxiliary [jcs.auxi
[INFO] Region [CENTRAL] Cache file root directory: /Users/tad/.m2/repository/org/owasp/dependency-check-data/4.0/cache
[INFO] Region [CENTRAL] Set maxKeySize to: '1000000'
[INFO] Region [CENTRAL] Indexed Disk Cache is alive.
[INFO] Parsed regions [POM, NODEAUDIT, CENTRAL]
[INFO] Finished configuration in 113 ms.
[INFO] Checking for updates
[INFO] Download Started for NVD CVE - Modified
[INFO] Download Complete for NVD CVE - Modified (816 ms)
[INFO] Processing Started for NVD CVE - Modified
[INFO] Processing Complete for NVD CVE - Modified (2753 ms)
[INFO] Begin database maintenance
[INFO] Updated the CPE ecosystem on 12 NVD records
[INFO] Cleaned up 4 orphaned NVD records
[INFO] End database maintenance (4257 ms)
[INFO] Begin database defrag
[INFO] End database defrag (10557 ms)
[INFO] Check for updates complete (24961 ms)
[INFO]

Dependency-Check is an open source tool performing a best effort analysis of 3rd party dependencies; false positives and false ne

[INFO] Analysis Started
[INFO] Finished Archive Analyzer (1 seconds)
[INFO] Finished File Name Analyzer (0 seconds)
[INFO] Finished Jar Analyzer (1 seconds)
[INFO] Finished Dependency Merging Analyzer (0 seconds)
[INFO] Finished Version Filter Analyzer (0 seconds)
[INFO] Finished Hint Analyzer (0 seconds)
    
```

- e. When the Run is complete, check under the **Target director** of the project to see the report: **dependency-check-report.html**.
 - i. A sample report is available: [Dependency-Check Report](#). The output report should look similar to the example below.

Project: DependencyCheck

Scan Information ([show all](#)):

- *dependency-check* version: 1.4.4-SNAPSHOT
- *Report Generated On*: Oct 9, 2016 at 07:04:35 EDT
- *Dependencies Scanned*: 306 (289 unique)
- *Vulnerable Dependencies*: 36
- *Vulnerabilities Found*: 289
- *Vulnerabilities Suppressed*: 0
- ...

Display: [Showing Vulnerable Dependencies \(click to show all\)](#)

Dependency	CPE	GAV	Highest Severity	CVE Count	CPE Confidence	Evidence Count
ghostscript/configure.ac	cpe:/a:ghostscript:ghostscript:8.62		High	5	HIGHEST	4
axis-1.4.jar	cpe:/a:apache:axis:1.4	axis:axis:1.4	Medium	2	HIGHEST	17
axis2-kernel-1.4.1.jar	cpe:/a:apache:axis2:1.4.1	org.apache.axis2:axis2-kernel:1.4.1	High	6	HIGHEST	16
ffmpeg/ffmpeg_version.cmake	cpe:/a:ffmpeg:ffmpeg:55.18.102		High	3	LOW	3
cmake/opencvDetectPython.cmake	cpe:/a:python:python:-		High	11	LOW	1
commons-fileupload-1.2.1.jar	cpe:/a:apache:commons_fileupload:1.2.1	commons-fileupload:commons-fileupload:1.2.1	High	3	HIGHEST	23
commons-httpclient-3.1.jar	cpe:/a:apache:commons-httpclient:3.1 cpe:/a:apache:httpclient:3.1	commons-httpclient:commons-httpclient:3.1	Medium	2	LOW	20