# CS 305 Module Two Coding Assignment

CS-305 Module 2

Professor Norman

Sean Jetté

## Run Dependency Check

A close-up of a check

Description automatically generated

## Document Results

**Dependency:** hibernate-validator-6.0.18.Final.jar

**Description:** Hibernate's Bean Validation (JSR-380) reference implementation.

**License:** http://www.apache.org/licenses/LICENSE-2.0.txt

**Vulnerabilities:** CVE-2023-1932 (OSSINDEX), CVE-2020-10693

**Highest Severity:** MEDIUM

**Dependency:** jackson-databind-2.10.2.jar  
**Description:** General data-binding functionality for Jackson: works on core streaming API  
**License:** http://www.apache.org/licenses/LICENSE-2.0.txt  
**Vulnerabilities:** CVE-2020-25649, CVE-2020-36518, CVE-2021-46877, CVE-2022-42003, CVE-2022-42004, CVE-2023-35116

**Highest Severity:** HIGH

**Dependency:** log4j-api-2.12.1.jar

**Description:** The Apache Log4j API

**License:** https://www.apache.org/licenses/LICENSE-2.0.txt

**Vulnerabilities:** CVE-2020-9488

**Highest Severity:** LOW

**Dependency:** logback-core-1.2.3.jar

**Description:** logback-core module

**License:** http://www.eclipse.org/legal/epl-v10.html, http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html

**Vulnerabilities:** CVE-2023-6378, CVE-2021-42550

**Highest Severity:** HIGH

**Dependency:** mongo-java-driver-2.4.jar

**Description:** Java Driver for MongoDB

**License:** The Apache Software License, Version 2.0: http://www.apache.org/licenses/LICENSE-2.0.txt

**Vulnerabilities:** CVE-2021-20328 (OSSINDEX)

**Highest Severity:** MEDIUM

**Dependency:** snakeyaml-1.25.jar

**Description:** YAML 1.1 parser and emitter for Java

**License:** Apache License, Version 2.0: http://www.apache.org/licenses/LICENSE-2.0.txt

**Vulnerabilities:** CVE-2022-1471, CVE-2017-18640, CVE-2022-25857, CVE-2022-38749, CVE-2022-38751, CVE-2022-38752, CVE-2022-41854, CVE-2022-38750

**Highest Severity:** CRITICAL

**Dependency:** spring-boot-2.2.4.RELEASE.jar

**Description:** Spring Boot

**License:** Apache License, Version 2.0: https://www.apache.org/licenses/LICENSE-2.0

**Vulnerabilities:** CVE-2023-20873, CVE-2022-27772, CVE-2023-20883

**Highest Severity:** CRITICAL

**Dependency:** spring-boot-starter-web-2.2.4.RELEASE.jar

**Description:** Starter for building web, including RESTful, applications using Spring MVC. Uses Tomcat as the default embedded container

**License:** Apache License, Version 2.0: https://www.apache.org/licenses/LICENSE-2.0

**Vulnerabilities:** CVE-2022-22965, CVE-2021-22118, CVE-2020-5421, CVE-2022-22950, CVE-2022-22971, CVE-2023-20861, CVE-2023-20863, CVE-2022-22968, CVE-2022-22970, CVE-2021-22060, CVE-2021-22096

**Highest Severity:** CRITICAL

**Dependency:** spring-core-5.2.3.RELEASE.jar

**Description:** Spring Core

**License:** Apache License, Version 2.0: https://www.apache.org/licenses/LICENSE-2.0

**Vulnerabilities:** CVE-2023-20873, CVE-2022-27772, CVE-2023-20883

**Highest Severity:** CRITICAL\*

**Dependency:** spring-expression-5.2.3.RELEASE.jar

**Description:** Spring Expression Language (SpEL)

**License:** Apache License, Version 2.0: http://www.apache.org/licenses/LICENSE-2.0.txt

**Vulnerabilities:** CVE-2022-1471, CVE-2017-18640, CVE-2022-25857, CVE-2022-38749, CVE-2022-38751, CVE-2022-38752, CVE-2022-41854, CVE-2022-38750

**Highest Severity:** CRITICAL\*

**Dependency:** spring-web-5.2.3.RELEASE.jar

**Description:** N/A

**License:** Apache License, Version 2.0: http://www.apache.org/licenses/LICENSE-2.0.txt

**Vulnerabilities:** CVE-2016-1000027, CVE-2022-22965, CVE-2024-38809 (OSSINDEX), CVE-2024-22243 (OSSINDEX), CVE-2024-22262 (OSSINDEX), CVE-2021-22118, CVE-2020-5421, CVE-2022-22950, CVE-2022-22971, CVE-2023-20861, CVE-2023-20863, CVE-2022-22968, CVE-2022-22970, CVE-2021-22060, CVE-2021-22096

**Highest Severity:** CRITICAL\*

**Dependency:** spring-webmvc-5.2.3.RELEASE.jar

**Description:** Spring Web MVC

**License:** Apache License, Version 2.0: https://www.apache.org/licenses/LICENSE-2.0

**Vulnerabilities:** CVE-2022-22965, CVE-2024-38816 (OSSINDEX), CVE-2021-22118, CVE-2020-5421, CVE-2022-22950, CVE-2022-22971, CVE-2023-20861, CVE-2023-20863, CVE-2022-22968, CVE-2022-22970, CVE-2021-22060, CVE-2021-22096

**Highest Severity:** CRITICAL\*

**Dependency**: tomcat-embed-core-9.0.30.jar

**Description**: Core Tomcat implementation

**License**: Apache License, Version 2.0: http://www.apache.org/licenses/LICENSE-2.0.txt

**Vulnerabilities:** CVE-2020-1938, CVE-2020-11996, CVE-2020-13934, CVE-2020-13935, CVE-2020-17527, CVE-2021-25122, CVE-2021-41079, CVE-2022-29885, CVE-2022-42252, CVE-2023-44487, CVE-2023-46589, CVE-2020-9484, CVE-2021-25329, CVE-2021-30640, CVE-2022-34305, CVE-2023-41080, CVE-2021-24122, CVE-2021-33037, CVE-2023-42795, CVE-2023-45648, CVE-2024-21733, CVE-2019-17569, CVE-2020-1935, CVE-2020-13943, CVE-2023-28708, CVE-2021-43980

**Highest Severity:** CRITICAL\*

**Dependency**: tomcat-embed-websocket-9.0.30.jar

**Description**: Core Tomcat implementation

**License**: Apache License, Version 2.0: http://www.apache.org/licenses/LICENSE-2.0.txt

**Vulnerabilities:** CVE-2020-1938, CVE-2020-8022, CVE-2020-11996, CVE-2020-13934, CVE-2020-13935, CVE-2020-17527, CVE-2021-25122, CVE-2021-41079, CVE-2022-29885, CVE-2022-42252, CVE-2023-44487, CVE-2023-46589, CVE-2020-9484, CVE-2021-25329, CVE-2021-30640, CVE-2022-34305, CVE-2023-41080, CVE-2021-24122, CVE-2021-33037, CVE-2023-42795, CVE-2023-45648, CVE-2024-21733, CVE-2019-17569, CVE-2020-1935, CVE-2020-13943, CVE-2023-28708, CVE-2021-43980

**Highest Severity:** CRITICAL\*

## Analyze Results

To identify the best solutions and address dependencies in the code base, I would need to identify all potential issues and recommended actions as per the dependency check and enforce the general guidelines. Specifically, starting with vulnerability scanning with Maven and then updating dependencies in accordance with the identified Common Vulnerabilities and Exposures (CVE) to the latest patched version (Ingram, 2021). This alone will address the many security flaws introduced by older versions such that if a CVE has a patched version of the dependency, upgrading will often resolve the vulnerability. Another best practice would be to utilize the CVE database regarding the details of each vulnerability to learn about affected versions and if there are any patches/workarounds (MITRE Corporation, 2024). Utilizing this sort of resource gives crucial information on any relevancy concerning identified vulnerabilities. As a method to learn more about CVEs and the impact they have, one can include the National Vulnerability Database provided by the U.S. Government’s NIST’s library with detailed information on the latest news and best practices (National Institute of Standards and Technology, 2024). Some dependencies might frequently update for security issues, and it would be important to ensure you use a version where significant CVEs are addressed before pushing out your product.

Key factors when using tools like the Maven dependency-check plugin is to filter out false positives. This can help to address only real vulnerabilities and spend less time on vulnerabilities that don’t actually exist (Hackerone, 2024). Moreover, trying to make edits to your code towards false positives might introduce security issues and bugs that affect the program. In order to filter these false positives out, we need to ensure we cross-reference vulnerabilities from the NVD and CVE and verify if the flagged dependency is actually used within the code. Additionally, “Suppressing these false positives is fairly easy using the HTML report. In the report next to each CPE identified (and on CVE entries) there is a suppress button” (Long, 2024).

According to the dependency check report, we have the following:

**hibernate-validator-6.0.18.Final.jar**: CVE-2023-1932 is a vulnerability in the isValid method of the SafeHtmlValidator class. An attacker can bypass validation by omitting the closing tag for certain HTML elements, which could lead to XSS attacks. Upgrading to a patched version mitigates this risk.

**jackson-databind-2.10.2.jar**: This version of Jackson Databind contains several flaws that allow XML External Entity (XXE) attacks, which could lead to denial-of-service (DoS) attacks via nested objects, arrays, or cyclic dependencies. These vulnerabilities can result in data integrity issues and resource exhaustion. Upgrading to a newer version addresses these issues.

**log4j-api-2.12.1.jar**: Apache Log4j's SMTP implementation does not properly validate certificates, which can allow man-in-the-middle (MITM) attacks on Simple Mail Transfer Protocol (SMTP) connections, leaking sensitive log messages. This vulnerability is fixed in versions 2.12.3 and 2.13.1.

**logback-core-1.2.3.jar**: The Logback receiver component has a serialization vulnerability that could allow denial-of-service (DoS) attacks or even arbitrary code execution via LDAP if malicious configuration files are used. Upgrading to a secure version eliminates these risks.

**mongo-java-driver-2.4.jar**: This version of the MongoDB Java driver has a vulnerability where it fails to properly verify the KMS server’s certificate hostname when using Client-Side Field Level Encryption. This flaw, when combined with a privileged network position, exposes traffic to MITM attacks, affecting the encryption process. Upgrading the dependency resolves this vulnerability.

**snakeyaml-1.25.jar**: SnakeYAML's Constructor() class allows unrestricted instantiation of types during deserialization, which could lead to remote code execution. Additionally, the lack of depth limits on collections during parsing allows for stack overflow Denial-of-Service (DoS) attacks when processing deeply nested YAML from untrusted sources. Upgrading to a newer version fixes these issues.

**spring-boot-2.2.4.RELEASE.jar**: This version is vulnerable to security bypasses in Cloud Foundry deployments and has risks related to temporary directory hijacking. It also has Denial-of-Service (DoS) risks when reverse proxy caching is involved. Upgrading to versions 3.0.6+ or 2.7.11+ addresses these vulnerabilities.

**spring-boot-starter-web-2.2.4.RELEASE.jar**: Similar to the spring-boot dependency, this version is also vulnerable to security bypasses in Cloud Foundry, temporary directory hijacking, and DoS risks from reverse proxy caching. Upgrading to 3.0.6+ or 2.7.11+ is recommended to fix these issues.

**spring-core-5.2.3.RELEASE.jar**: Vulnerable to remote code execution (RCE) in JDK 9+ when deployed on Tomcat with WAR files. There is also a risk of privilege escalation in WebFlux and DoS via crafted SpEL expressions. Upgrading to a patched version mitigates these threats.

**spring-expression-5.2.3.RELEASE.jar**: This version also contains an RCE vulnerability in JDK 9+ with WAR on Tomcat, as well as a risk of DoS from crafted SpEL expressions. Upgrading to 5.3.27+ resolves these issues.

**spring-web-5.2.3.RELEASE.jar**: This version includes a vulnerability in the Spring Web module that can lead to potential information leaks and unauthorized access due to improper configuration handling. Upgrading to 5.3.27+ resolves this issue by tightening security configurations.

**spring-webmvc-5.2.3.RELEASE.jar**: This version contains multiple vulnerabilities, including an RCE vulnerability in Spring MVC or Spring WebFlux applications running on JDK 9+ and deployed on Tomcat, as well as path traversal issues when serving static resources via WebMvc.fn or WebFlux.fn. Additionally, there are DoS risks from malicious SpEL expressions and multipart file uploads. Upgrading to Spring 5.3.27+ or 5.2.24+ resolves most of these issues.

**tomcat-embed-core-9.0.30.jar**: This version includes a vulnerability in Apache Tomcat related to improper privilege management when using the Apache JServ Protocol (AJP), potentially allowing remote code execution. Upgrading to 9.0.31 or later addresses this issue by hardening the default AJP Connector configuration and preventing unauthorized access.

**tomcat-embed-websocket-9.0.30.jar**: Apache Tomcat versions 9.0.30 and earlier have a vulnerability in the WebSocket implementation that allows remote attackers to trigger DoS attacks by causing excessive memory consumption through WebSocket handshake requests. This issue can lead to server resource exhaustion. Upgrading to a newer version mitigates this problem.

**Resources:**

Hackerone. (2024). *OWASP Dependency-Check: How it works, Benefits & Pros/Cons*. https://www.hackerone.com/knowledge-center/owasp-dependency-check-how-it-works-benefits-and-proscons

Ingram, D. (2021, July 28). Best practices for dependency management. *Google Cloud Blog*. https://cloud.google.com/blog/topics/developers-practitioners/best-practices-dependency-management

Long, J. (2024, October 30). *Suppressing False Positives – dependency-check-maven*. https://jeremylong.github.io/DependencyCheck/general/suppression.html

MITRE Corporation. (2024). *CVE Program Mission*. CVE. https://www.cve.org/

National Institute of Standards and Technology. (2024, August 27). *National Vulnerability Database*. National Institute of Standards and Technology - NVD. https://nvd.nist.gov/