# CS 305 Module Two Coding Assignment Template

## Instructions

Replace the bracketed text with the relevant information in your own words. If you choose to include images or supporting materials, make certain to insert them in all the relevant locations in the document.

## Run Dependency Check

A screenshot of a computer

Description automatically generated

## Document Results

After running the dependency check on the code base, 13 vulnerable dependencies were identified as follows:

1. hibernate-validator-6.0.18.Final.jar – Description: Hibernate’s Bean Validation (JSR-380) reference implementation. CVE-2020-10693 – MEDIUM severity - A bug in the message interpolation processor enables invalid EL expressions to be evaluated.
2. jackson-databind-2.10.2.jar – Description: General data-binding functionality for Jackson: works on core streaming API - 6 associated CVEs, mostly allowing attackers to cause denial of service
3. log4j-api-2.12.1.jar – CVE-2020-9488 – LOW Severity – Description: The Apachi Log4j API. CVE-2020-9488 - Improper validation of certificate with host mismatch in Apache Log4j SMTP appender. This could allow an SMTPS connection to be intercepted by a man-in-the-middle attack which could leak any log messages sent through that appender.
4. logback-core-1.2.3.jar – MEDIUM Severity – Description: logback-core module – CVE-2021-42550 - In logback version 1.2.7 and prior versions, an attacker with the required privileges to edit configurations files could craft a malicious configuration allowing to execute arbitrary code loaded from LDAP servers.
5. mongo-java-driver-2.4.jar – MEDIUM Severity – Description: Java driver for MongoDB – CVE-2021-20328 - Specific versions of the Java driver that support client-side field level encryption (CSFLE) fail to perform correct host name verification on the KMS server’s certificate. This vulnerability in combination with a privileged network position active MITM attack could result in interception of traffic between the Java driver and the KMS service rendering Field Level Encryption ineffective.
6. snakeyaml-1.25.jar – CRITICAL Severity – Description: YAML 1.1 parser and emitter for Java. 8 associated CVEs Ranging with vulnerabilities that could include DoS attacks and remote code execution
7. spring-boot-2.2.4.RELEASE.jar – CRITICAL Severity – Description – Spring boot – 3 associated CVEs that include vulnerabilities ranging from security bypass, temporary directory hijacking, and DoS attacks.
8. spring-boot-starter-web-2.2.4.RELEASE.jar – CRITICAL Severity – Starter for building web, including RESTful, applications using Spring MVC. Uses Tomcat as the default embedded container. – 3 associated CVEs that include vulnerabilities ranging from security bypass, temporary directory hijacking and DoS attacks.
9. spring-core-5.2.3.RELEASE.jar – CRITICAL Severity – Description: Spring Core. 11 associated CVEs including one CISA known exploited vulnerability. Vulnerabilities range from remote code execution, privilege escalation, protection bypassing, and denial of service attacks.
10. spring-web-5.2.3.RELEASE.jar – CRITICAL Severity – Description: Spring Web. 12 associated CVEs. Vulnerabilities range from Remote Code Execution, privilege escalation, protection bypassing, DoS attacks, and malicious input that causes the insertion of additional log entries.
11. spring-webmvc-5.2.3.RELEASE.jar – CRITICAL Severity – Description: Spring Web MVC. 11 Associated CVEs. Vulnerabilities range from code injection to privilege escalation, protection bypassing, DoS attacks, and malicious input.
12. tomcat-embed-core-9.0.30.jar – CRITICAL Severity – Description: Core Tomcat Implementation. 21 Associated CVEs. Vulnerabilities range from a known exploited vulnerability of improper priviledge management to triggering high CPU usage rendering the server unresponsive, DoS attacks, connection closures, information leaks, and exposure of sensitive information.
13. tomcat-embed-websocket-9.0.30.jar – CRITICAL Severity – Description: Core Tomcat Implementation. 22 Associated CVEs. Vulnerabilities range from improper privilege management, incorrect default permissions, high CPU usage causing unresponsiveness, DoS attacks, exposure of sensitive information, protection bypassing, URL redirection to untrusted sites, and unprotected transport of credentials.

## Analyze Results

Solutions for the identified dependencies are as follows:

1. Hibernate-validator-6.0.18.Final.jar - CVE-2020-10693 – Its Weakness type (CWE) is Input validation. Fix: Upgrade to version org.hibernate-validator:6.0.20.Final,6.1.5.Final
2. Jackson-databind-2.10.2.jar
   1. CVE-2023-35166 - Its weakness type is (CWE-611) Allocation of resources without limits or throttling. Fix: Upgrade to version com.fasterxml.jackson.core:Jackson-databind:2.12.6,2.13.1
   2. CVE-2021-46877 - Its weakness type (CWE-770) is allocation of resources without limits or throttling. Fix: Upgrade to version com.fasterxml.jackson.core:jackson-databind:2.12.6,2.13.1
   3. CVE-2022-42004 - Its weakness type (CWE-502) is deserialization of untrusted data. Fix: Upgrade to version com.fasterxml.jackson.core:jackson-databind:2.13.4
   4. CVE-2022-42003 - Its weakness type is (CWE-502) Deserialization of Untrusted Data. Fix: Upgrade to version com.fasterxml.jackson.core:jackson-databind:2.12.7.1,2.13.4.1
   5. CVE-2020-36518 - Its weakness type (CWE-787) is out-of-bounds write. Fix: Upgrade to version com.fasterxml.jackson.core:jackson-databind:2.12.6.1,2.13.2.1
   6. CVE-2020-25649 - Its weakness type (CWE-611) is improper restriction of XML external entity reference. Fix: Upgrade to version com.fasterxml.jackson.core:jackson-databind:2.6.7.4,2.9.10.7,2.10.5.1,2.11.0.rc1.
3. log4j-api-2.12.1.jar - Fixed in Apache Log4j 2.12.3 and 2.13.1 – CVE-2020-9488 - Its weakness type (CWE-295) is improper certificate validation. Fix: Upgrade to version ch.qos.reload4j:reload4j:1.2.18.3
4. logback-core-1.2.3.jar – Its weakness type (CWE-502) is deserialization of untrusted data. Fix: Upgrade to version ch.qos.logback:logback-classic:1.2.9;ch.qos.logback:logback-core:1.2.9
5. mongo-java-driver-2.4.jar – CVE-2021-20328 – Its weakness type (CWE-295) is improper certificate validation. Fix: Upgrade to version org.mongodb:mongodb-driver-sync:3.12.8,3.11.3,4.1.2,4.2.1,4.0.6;org.mongodb:mongodb-driver-legacy:3.12.8,3.11.3,4.1.2,4.2.1,4.0.6;org.mongodb:mongodb-driver:3.12.8,3.11.3;org.mongodb:mongo-java-driver:3.12.8,3.11.3
6. snakeyaml-1.25.jar - The following is a list of associated CVEs with different weakness types and their assaociated fixes.
   1. CVE-2022-1471 – Weakness type (CWE-502) Deserialization of Untrusted Data. Fix: Upgrade to version org.yaml:snakeyaml:2.0.
   2. CVE-2022-25857 - Weakness type (CWE-776) Improper restriction of recursive entity references in DTDs (‘XML Entity Expansion’).
   3. CVE-2022-38749 – Weakness type (CWE-787) Out of Bounds. Fix: Upgrade to version org.yaml:snakeyaml:1.31

There were five additional CVEs that had the same weakness types and their fixes were also similar.

1. spring-boot-2.2.4.RELEASE.jar – 3 Associated CVEs
   1. CVE-2023-20873 -no information regarding the weakness(NVD-CWE) - Recommended fix: Upgrade to version org.springframework.boot:spring-boot-actuator-autoconfigure:2.7.11,3.0.6
   2. CVE-2022-27772 – Its weakness type (CWE-668) Exposure of resources to wrong sphere. Fix: Upgrade to version org.springframework.boot:spring-boot-actuator-autoconfigure:2.7.11,3.0.6
   3. CVE-2023-20883 – Its weakness type (CWE-400) Uncontrolled Resource Consumption. Fix: Upgrade to version org.springframework.boot:spring-boot-actuator-autoconfigure:2.7.11,3.0.6
2. spring-boot-starter-web-2.2.4.RELEASE.jar – 3 Associated CVEs. – The CVEs, their weakness types, and their fixes are all the same as the previously mentioned dependency.
3. spring-core-5.2.3.RELEASE.jar – The following is a list of associated CVEs with different weakness types and their associated fixes.
   1. CVE-2022-22965 – Its weakness type (CWE-94) is improper control of generation of code (‘Code Injection’). Fix: update to Spring framework 5.3.18 and 5.2.20 or greater.
   2. CVE-2021-22118 – Its weakness type (CWE-668) is exposure of resource to wrong sphere. Fix: Upgrade to version org.springframework:spring-web:5.2.15,5.3.7
   3. CVE-2020-5421 – no info regarding weakness type. Fix: Upgrade to version org.springframework:spring-web:4.3.29,5.0.19,5.1.18,5.2.9
   4. CVE-2022-22950 – Weakness type (CWE-770) allocation of resources without limits or throttling. Fix: Upgrade to version org.springframework:spring-expression:5.2.20,5.3.17
   5. CVE-2023-20863 – Weakness type (CWE-917) is Improper Neutralization of Special Elements used in an Expression Language Statement ('Expression Language Injection'). Fix: Upgrade to version org.springframework:spring-expression - 5.2.24.RELEASE,5.3.27,6.0.8
   6. CVE-2022-22968 – Weakness type (CWE-178) is improper handling of case sensitivity. Fix: Upgrade to version org.springframework:spring-expression - 5.2.24.RELEASE,5.3.27,6.0.8

There were five additional CVEs that had the same weakness types and their fixes were also similar.

1. spring-web-5.2.3.RELEASE.jar - The following is a list of associated CVEs with different weakness types and their associated fixes.
   1. CVE-2016-1000027 – Weakness type (CWE-502) is deserialization of untrusted data. Fix: Upgrade to version org.springframework:spring-web:6.0.0
   2. CVE-2022-22965 - Its weakness type (CWE-94) is improper control of generation of code (‘Code Injection’). Fix: update to Spring framework 5.3.18 and 5.2.20 or greater.
   3. CVE-2021-22118 – Its weakness type (CWE-668) is exposure of resource to wrong sphere. Fix: Upgrade to version org.springframework:spring-web:5.2.15,5.3.7
   4. CVE – 2022-22950 – Its weakness type (CWE-770) is allocation of resources without limits or throttling. Fix: Upgrade to version org.springframework:spring-expression:5.2.20,5.3.17
   5. CVE-2023-20863 – Weakness type (CWE-917) is Improper Neutralization of Special Elements used in an Expression Language Statement ('Expression Language Injection'). Fix: Upgrade to version org.springframework:spring-expression - 5.2.24.RELEASE,5.3.27,6.0.8

This list continues on with many of the same types of weaknesses and associated CVEs from previously discussed dependencies. Again, there were a total of 12 CVE associated with this dependency, all of their fixes were similar requiring an update.

1. spring-webmvc-5.2.3.RELEASE.jar – This dependency was also quite similar to previously mentioned spring related dependencies. Their central solution to all identified CVEs was to update the corresponding version or framework.
2. tomcat-embed-core-9.0.30. - The following is a list of associated CVEs with different weakness types and their associated fixes.
   1. CVE-2020-1938 – Weakness type (CWE-20 & CWE-269) is input validation and improper privilege management. Fix: Upgrade to version 7.0.100,8.5.51,9.0.31
   2. CVE-2020-13934 – Weakness type (CWE-401 & CWE-476) Missing release of memory after effective lifetime and NULL pointer dereference. Fix: Upgrade to version org.apache.tomcat:tomcat-coyote:8.5.57,9.0.37,10.0.0-M7
   3. CVE-2020-13935 – Weakness type (CWE-835) is loop with unreachable exit condition (‘Infinite Loop’). Fix: Upgrade to version org.apache.tomcat:tomcat-websocket:7.0.105,8.5.57,9.0.37,10.0.0-M7;org.apache.tomcat.embed:tomcat-embed-websocket:7.0.105,8.5.57,9.0.37,10.0.0-M7
   4. CVE-2020-17527 – Weakness type (CWE-200) is exposure of sensitive information to an unauthorized actor. Fix: Upgrade to version org.apache.tomcat:tomcat-websocket:7.0.105,8.5.57,9.0.37,10.0.0-M7;org.apache.tomcat.embed:tomcat-embed-websocket:7.0.105,8.5.57,9.0.37,10.0.0-M7

This list continues with similar vulnerabilities and various weakness types. All show common fixes as performing updates.

1. tomcat-embed-websocket-9.0.30.jar –

This dependency was similar to the prior dependency. CVEs, their weakness types were matching all providing the same similar solutions: upgrade version.

Additionally, a person should filter false positives from the dependency-check tool as a means of knowing which CVEs actually apply or are exploitable in the environment we are working. Checking to see if the CPE (Common Platform Enumeration identifier) is correct as well as looking at the evidence count will help when determining whether any particular CVE is or is not a false positive (OWASP, 2023). Filtering false positives will also help identify errors quicker as they relate to the work being done. False positives can be suppressed so that they do not continue to show up in future reports.

**References**

NIST - National Institute of Standards and technology. (n.d.). National Vulnerability Database.

<https://nvd.nist.gov/>

*Open Source Vulnerability Database: MEND*. Mend.io. (2022, May 25).

https://www.mend.io/vulnerability-database/

OWASP. (2023). How To Read the Reports. dependency-check – How To Read The Reports.

https://jeremylong.github.io/DependencyCheck/general/thereport.html