# 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

AI-generated content may be incorrect.

## Document Results

**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>

**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>

**log4j-api-2.12.1.jar**

Description: The Apache Log4j API

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

**logback-classic-1.2.3.jar**

Description: logback-classic module

License: [http://www.eclipse.org/legal/epl-v10.html, http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html](http://www.eclipse.org/legal/epl-v10.html,%20http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html)

**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](http://www.eclipse.org/legal/epl-v10.html,%20http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html)

**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>

**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>

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

Description: Spring Boot

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

**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>

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

Description: Spring Core

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

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

Description: Spring Expression Language (SpEL)

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

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

Description: Spring Web

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

**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>

**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>

**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>

## Analyze Results

False positives distract from real vulnerabilities, therefore removing them is necessary so that time and resources are not wasted assessing them. Additionally, modifying code that was previously secure and functioning properly in an attempt to fix false positives can create new, real security vulnerabilities.

In this report, I didn’t identify any false positives. Every vulnerability ID aligns with the corresponding listed dependency; the "safe" dependencies have already been filtered out.

* **hibernate-validator-6.0.18.Final.jar**
* A flaw in the message interpolation processor allows invalid EL expressions to be processed as if they were valid. This vulnerability enables attackers to circumvent input sanitation measures, such as escaping or stripping, that developers may have implemented when managing user-controlled data in error messages.
* Upgrading to latest version is recommended.
* **jackson-databind-2.10.2.jar**
* Vulnerabilities found in FasterXML jackson-databind, leading to denial of service or resource exhaustion. These include issues with cyclic dependencies, deeply nested objects or arrays, and improper handling of JsonNode JDK serialization, which can cause excessive memory use or StackOverflow exceptions. Additionally, there’s a flaw related to XML external entity (XXE) attacks that can impact data integrity.
* Upgrading to latest version is recommended to mitigate risks.
* **log4j-api-2.12.1.jar**
* Vulnerabilities found in Apache Log4j2 across various versions, leading to remote code execution (RCE), denial of service, information leaks, and security bypasses. These issues include improper handling of JNDI lookups, allowing attackers to execute arbitrary code or leak data through crafted inputs, especially with LDAP endpoints or non-default configurations. Some versions were also vulnerable to uncontrolled recursion in self-referential lookups, causing denial of service. Additionally, there was improper certificate validation in the SMTP appender, making it susceptible to man-in-the-middle attacks.
* Upgrading to latest version is recommended to mitigate risks.

### logback-classic-1.2.3.jar

### A serialization vulnerability in the logback receiver component of version 1.4.11 allows attackers to trigger a Denial-of-Service (DoS) attack by sending poisoned data. Additionally, in versions 1.2.7 and earlier, an attacker with access to configuration files could craft a malicious configuration to execute arbitrary code loaded from LDAP servers.

* Upgrading to latest version is recommended to mitigate risks.

### logback-core-1.2.3.jar

### A serialization vulnerability in the logback receiver component of version 1.4.11 allows attackers to trigger a Denial-of-Service (DoS) attack by sending poisoned data. Additionally, in versions 1.2.7 and earlier, an attacker with access to configuration files could craft a malicious configuration to execute arbitrary code loaded from LDAP servers.

* Upgrading to latest version is recommended to mitigate risks.

### mongo-java-driver-2.4.jar

### Certain versions of the Java driver that support client-side field level encryption (CSFLE) have a vulnerability where they don't properly verify the KMS server's certificate. This could allow an attacker in a privileged network position to intercept traffic between the Java driver and the KMS service, undermining the encryption. However, this issue doesn't affect the Java async, Scala, or reactive streams drivers, and it doesn't impact applications running within AWS, GCP, or Azure due to compensating security controls. The vulnerability only affects driver workloads using Field Level Encryption.

* Upgrading to latest version is recommended to mitigate risks.

### snakeyaml-1.25.jar

### SnakeYAML versions prior to 1.31 are vulnerable to several security issues. These include the ability for attackers to trigger remote code execution by deserializing untrusted YAML content through the Constructor() class, which doesn't restrict types. Additionally, SnakeYAML versions before 1.31 are prone to Denial of Service (DoS) attacks, where malicious input can cause a stack overflow and crash the parser. Versions prior to 1.26 also have an issue with the Alias feature, allowing entity expansion during loading.

### To mitigate these risks, it’s recommended to use SafeConstructor for untrusted content and upgrade to SnakeYAML 2.0 or higher.

### spring-boot-2.2.4.RELEASE.jar

### Spring Boot versions 3.0.0 - 3.0.6, 2.7.0 - 2.7.11, 2.6.0 - 2.6.14, 2.5.0 - 2.5.14, and older unsupported versions are vulnerable to a denial-of-service (DoS) attack when used with a reverse proxy cache in Spring MVC. Additionally, versions 3.0.0 - 3.0.5 and 2.7.0 - 2.7.10 are susceptible to a security bypass when deployed to Cloud Foundry. Versions prior to 2.2.11.RELEASE are vulnerable to temporary directory hijacking.

### To mitigate these risks, users should upgrade to Spring Boot 3.0.6 or higher (for 3.0.x), 2.7.11 or higher (for 2.7.x), and apply the latest updates for unsupported older versions to patch the vulnerabilities.

### spring-boot-starter-web-2.2.4.RELEASE.jar,

### Spring Boot versions 3.0.0 - 3.0.6, 2.7.0 - 2.7.11, 2.6.0 - 2.6.14, 2.5.0 - 2.5.14, and older unsupported versions are vulnerable to a denial-of-service (DoS) attack when Spring MVC is used with a reverse proxy cache. Additionally, versions 3.0.0 - 3.0.5, 2.7.0 - 2.7.10, and older unsupported versions are at risk of a security bypass when deployed to Cloud Foundry. Versions before 2.2.11.RELEASE also have a vulnerability that allows temporary directory hijacking.

### To mitigate these risks, users should upgrade to Spring Boot 3.0.6 or higher for 3.0.x users, 2.7.11 or higher for 2.7.x users, or apply the latest patches for unsupported older versions.

### spring-core-5.2.3.RELEASE.jar, spring-expression-5.2.3.RELEASE.jar, spring-web-5.2.3.RELEASE.jar, spring-webmvc-5.2.3.RELEASE.jar

### Several versions of the Spring Framework are vulnerable to various security issues, including denial-of-service (DoS) attacks caused by specially crafted SpEL expressions (up to 5.3.27 and 6.0.8), STOMP over WebSocket vulnerabilities (prior to 5.3.20 and 5.2.22), and DoS risks in file upload handling (prior to 5.3.20 and 5.2.22). Other vulnerabilities include incorrect protections against case-sensitive field names in DataBinder (up to 5.3.18), remote code execution (RCE) risks in applications running on JDK 9+ (up to 5.3.16), and insertion of malicious log entries (up to 5.3.13). Additionally, there are risks of bypassing protections against RFD attacks (5.2.8–5.2.22) and potential RCE due to untrusted data deserialization (up to 5.3.16). WebFlux applications prior to 5.3.7 are also vulnerable to privilege escalation through temporary storage directory manipulation.

### To mitigate these risks, users should upgrade to the latest Spring Framework versions.

### tomcat-embed-core-9.0.30.jar, tomcat-embed-websocket-9.0.30.jar

### Several versions of Apache Tomcat are affected by various vulnerabilities, including potential denial-of-service (DoS) attacks, information leakage, request smuggling, cross-site scripting (XSS), and remote code execution risks. Key issues include resource allocation without limits (OutOfMemoryError), improper input validation of HTTP trailer headers leading to request smuggling, incomplete cleanup causing information leaks, and URL redirection vulnerabilities. Other concerns include improper validation of incoming TLS packets, incorrect parsing of HTTP headers, and security bypasses in authentication mechanisms.

### To mitigate these risks, users should upgrade to the latest Apache Tomcat versions. Additionally, users should review configurations and apply relevant security patches for their specific deployment versions.