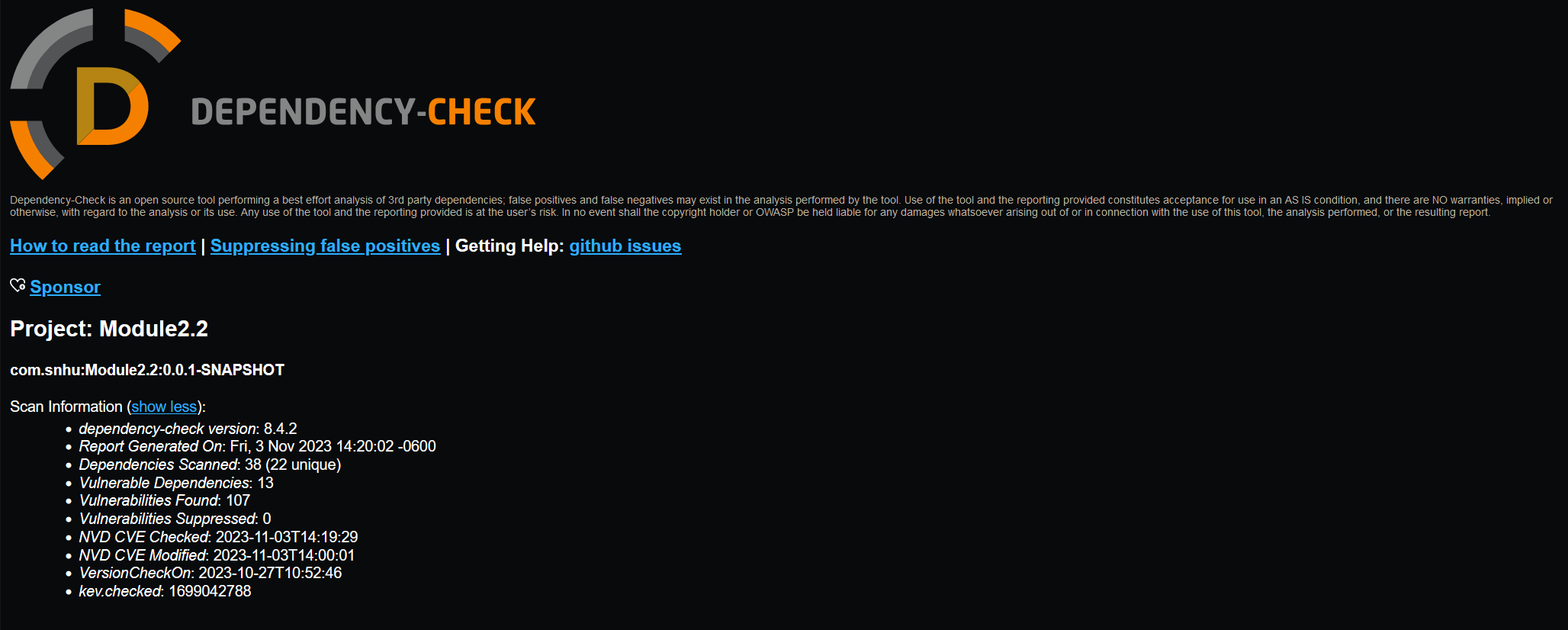
# CS 305 Module Two Coding Assignment Template

## 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



# Document Results

# Dependencies and their top vulnerability

### hibernate-validator-6.0.18.Final.jar

[**CVE-2020-10693**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-10693) **-** A flaw was found in Hibernate Validator version 6.1.2.Final. A bug in the message interpolation processor enables invalid EL expressions to be evaluated as if they were valid. This flaw allows attackers to bypass input sanitation (escaping, stripping) controls that developers may have put in place when handling user-controlled data in error messages.

### jackson-databind-2.10.2.jar

[**CVE-2020-25649**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-25649) **-** A flaw was found in FasterXML Jackson Databind, where it did not have entity expansion secured properly. This flaw allows vulnerability to XML external entity (XXE) attacks. The highest threat from this vulnerability is data integrity.

### log4j-api-2.12.1.jar

[**CVE-2020-9488**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=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. Fixed in Apache Log4j 2.12.3 and 2.13.1

### logback-core-1.2.3.jar

[**CVE-2021-42550**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=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.

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

**CVE-2021-20328** (OSSINDEX) [[CVE-2021-20328] CWE-295: Improper Certificate Validation](https://ossindex.sonatype.org/vulnerability/CVE-2021-20328?component-type=maven&component-name=org.mongodb%2Fmongo-java-driver&utm_source=dependency-check&utm_medium=integration&utm_content=8.4.2) - 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. This issue was discovered during internal testing and affects all versions of the Java driver that support CSFLE. The Java async, Scala, and reactive streams drivers are not impacted. This vulnerability does not impact driver traffic payloads with CSFLE-supported key services originating from applications residing inside the AWS, GCP, and Azure network fabrics due to compensating controls in these environments. This issue does not impact driver workloads that don’t use Field Level Encryption.

### snakeyaml-1.25.jar

[**CVE-2022-1471**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-1471) **-** SnakeYaml's Constructor() class does not restrict types which can be instantiated during deserialization. Deserializing yaml content provided by an attacker can lead to remote code execution. We recommend using SnakeYaml's SafeConsturctor when parsing untrusted content to restrict deserialization. We recommend upgrading to version 2.0 and beyond.

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

[**CVE-2023-20873**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20873) **-** In Spring Boot versions 3.0.0 - 3.0.5, 2.7.0 - 2.7.10, and older unsupported versions, an application that is deployed to Cloud Foundry could be susceptible to a security bypass. Users of affected versions should apply the following mitigation: 3.0.x users should upgrade to 3.0.6+. 2.7.x users should upgrade to 2.7.11+. Users of older, unsupported versions should upgrade to 3.0.6+ or 2.7.11+.

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

[**CVE-2023-20873**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2023-20873) **-** In Spring Boot versions 3.0.0 - 3.0.5, 2.7.0 - 2.7.10, and older unsupported versions, an application that is deployed to Cloud Foundry could be susceptible to a security bypass. Users of affected versions should apply the following mitigation: 3.0.x users should upgrade to 3.0.6+. 2.7.x users should upgrade to 2.7.11+. Users of older, unsupported versions should upgrade to 3.0.6+ or 2.7.11+.

### spring-core-5.2.3.RELEASE.jar

[**CVE-2022-22965**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22965) **–**

**CISA Known Exploited Vulnerability:**

* Product: VMware Spring Framework
* Name: Spring Framework JDK 9+ Remote Code Execution Vulnerability
* Date Added: 2022-04-04
* Description: Spring MVC or Spring WebFlux application running on JDK 9+ may be vulnerable to remote code execution (RCE) via data binding.
* Required Action: Apply updates per vendor instructions.
* Due Date: 2022-04-25

A Spring MVC or Spring WebFlux application running on JDK 9+ may be vulnerable to remote code execution (RCE) via data binding. The specific exploit requires the application to run on Tomcat as a WAR deployment. If the application is deployed as a Spring Boot executable jar, i.e. the default, it is not vulnerable to the exploit. However, the nature of the vulnerability is more general, and there may be other ways to exploit it.

### spring-web-5.2.3.RELEASE.jar

[**CVE-2016-1000027**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2016-1000027) **-** Pivotal Spring Framework through 5.3.16 suffers from a potential remote code execution (RCE) issue if used for Java deserialization of untrusted data. Depending on how the library is implemented within a product, this issue may or not occur, and authentication may be required. NOTE: the vendor's position is that untrusted data is not an intended use case. The product's behavior will not be changed because some users rely on deserialization of trusted data.

### spring-webmvc-5.2.3.RELEASE.jar

**[CVE-2022-22965](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2022-22965" \t "_blank) –**

**CISA Known Exploited Vulnerability:**

* Product: VMware Spring Framework
* Name: Spring Framework JDK 9+ Remote Code Execution Vulnerability
* Date Added: 2022-04-04
* Description: Spring MVC or Spring WebFlux application running on JDK 9+ may be vulnerable to remote code execution (RCE) via data binding.
* Required Action: Apply updates per vendor instructions.
* Due Date: 2022-04-25

### A Spring MVC or Spring WebFlux application running on JDK 9+ may be vulnerable to remote code execution (RCE) via data binding. The specific exploit requires the application to run on Tomcat as a WAR deployment. If the application is deployed as a Spring Boot executable jar, i.e. the default, it is not vulnerable to the exploit. However, the nature of the vulnerability is more general, and there may be other ways to exploit it.

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

[**CVE-2020-1938**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-1938) **–**

**CISA Known Exploited Vulnerability:**

* Product: Apache Tomcat
* Name: Apache Tomcat Improper Privilege Management Vulnerability
* Date Added: 2022-03-03
* Description: Apache Tomcat treats Apache JServ Protocol (AJP) connections as having higher trust than, for example, a similar HTTP connection. If such connections are available to an attacker, they can be exploited.
* Required Action: Apply updates per vendor instructions.
* Due Date: 2022-03-17

When using the Apache JServ Protocol (AJP), care must be taken when trusting incoming connections to Apache Tomcat. Tomcat treats AJP connections as having higher trust than, for example, a similar HTTP connection. If such connections are available to an attacker, they can be exploited in ways that may be surprising. In Apache Tomcat 9.0.0.M1 to 9.0.0.30, 8.5.0 to 8.5.50 and 7.0.0 to 7.0.99, Tomcat shipped with an AJP Connector enabled by default that listened on all configured IP addresses. It was expected (and recommended in the security guide) that this Connector would be disabled if not required. This vulnerability report identified a mechanism that allowed: - returning arbitrary files from anywhere in the web application - processing any file in the web application as a JSP Further, if the web application allowed file upload and stored those files within the web application (or the attacker was able to control the content of the web application by some other means) then this, along with the ability to process a file as a JSP, made remote code execution possible. It is important to note that mitigation is only required if an AJP port is accessible to untrusted users. Users wishing to take a defence-in-depth approach and block the vector that permits returning arbitrary files and execution as JSP may upgrade to Apache Tomcat 9.0.31, 8.5.51 or 7.0.100 or later. A number of changes were made to the default AJP Connector configuration in 9.0.31 to harden the default configuration. It is likely that users upgrading to 9.0.31, 8.5.51 or 7.0.100 or later will need to make small changes to their configurations.

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

[**CVE-2020-1938**](https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-1938) **–**

**CISA Known Exploited Vulnerability:**

* Product: Apache Tomcat
* Name: Apache Tomcat Improper Privilege Management Vulnerability
* Date Added: 2022-03-03
* Description: Apache Tomcat treats Apache JServ Protocol (AJP) connections as having higher trust than, for example, a similar HTTP connection. If such connections are available to an attacker, they can be exploited.
* Required Action: Apply updates per vendor instructions.
* Due Date: 2022-03-17

When using the Apache JServ Protocol (AJP), care must be taken when trusting incoming connections to Apache Tomcat. Tomcat treats AJP connections as having higher trust than, for example, a similar HTTP connection. If such connections are available to an attacker, they can be exploited in ways that may be surprising. In Apache Tomcat 9.0.0.M1 to 9.0.0.30, 8.5.0 to 8.5.50 and 7.0.0 to 7.0.99, Tomcat shipped with an AJP Connector enabled by default that listened on all configured IP addresses. It was expected (and recommended in the security guide) that this Connector would be disabled if not required. This vulnerability report identified a mechanism that allowed: - returning arbitrary files from anywhere in the web application - processing any file in the web application as a JSP Further, if the web application allowed file upload and stored those files within the web application (or the attacker was able to control the content of the web application by some other means) then this, along with the ability to process a file as a JSP, made remote code execution possible. It is important to note that mitigation is only required if an AJP port is accessible to untrusted users. Users wishing to take a defence-in-depth approach and block the vector that permits returning arbitrary files and execution as JSP may upgrade to Apache Tomcat 9.0.31, 8.5.51 or 7.0.100 or later. A number of changes were made to the default AJP Connector configuration in 9.0.31 to harden the default configuration. It is likely that users upgrading to 9.0.31, 8.5.51 or 7.0.100 or later will need to make small changes to their configurations.

# Analyze Results

All solutions were found in the vulnerability link provided above and in the links for each vulnerability in the dependency report.

### hibernate-validator-6.0.18. Final.jar

According to <https://lists.apache.org/thread/cyq0gvcpf74p4w25wv4ssqph2q7b4cfn>:

“The following commit(s) were added to refs/heads/master by this push:

new 8bd2e48 PLUTO-791 Upgrade to hibernate-validator-6.0.20.Final due to CVE-2020-10693 and CVE-2019-10219

8bd2e48 is described below

commit 8bd2e48511f82d60bd1c93b4092c2be1b67e633b

Author: Neil Griffin <ne...@gmail.com>

AuthorDate: Wed Jul 14 16:57:19 2021 -0400

PLUTO-791 Upgrade to hibernate-validator-6.0.20.Final due to CVE-2020-10693 and CVE-2019-10219”

Upgrading to the 6.0.20 version is the accepted solution for this vulnerability.

### jackson-databind-2.10.2.jar

Solution can be found here: <https://github.com/FasterXML/jackson-databind/issues/2589>

In short, I believe this issue is fixed in version 2.12.0. Upgrading to that version should resolve the vulnerability.

### How to find potential solutions

Similar to how it was approached it above, each of the solutions can be found on the NVD link provided with each vulnerability. From there, look through the links provided to find one that was tagged as a patch. Sometimes the solution is provided in the CVE description, in that case following the NVD link might not be necessary.

### Filtering false positives

Filtering false positives will simply improve the accuracy of the tool. The next time you run it it will no longer show “bad” or “irrelevant” information. This will also save a lot of time for the developer as there is less data for them to sort through that results in dead-ends.