**OWASP Top 10 Part 2**

Module 9 | Activity 2



**Introduction**

In Module 9, we learned all about web vulnerabilities and the Open Web Application Security Project (OWASP). We discovered the top ten list of web vulnerabilities created by OWASP. In Activity 1, we covered the top 1-5 vulnerabilities. Activity 2 will go over the last 6-10 of the vulnerabilities: Security Misconfiguration, Cross-Site Scripting XSS, Insecure Deserialization, Using Components with Known Vulnerabilities, Insufficient Logging & Monitoring. This activity will help you grasp a better understanding of each vulnerability and put them in practice.

**Getting Started**

Similar to Activity 1 of this module, we will be using [HACKSPLAINING](https://www.hacksplaining.com/owasp) to complete this online activity. The instructions will guide you through the exercises on the website, and help emphasize the most important material. We will be completing the corresponding exercises for the web vulnerabilities below.

First, go on over to the [Link](https://www.hacksplaining.com/) and click the sign up button on the top right corner. Sign up either through email or GitHub (common hacker website).

**6. Security Misconfiguration**

The [HACKSPLAINING](https://www.hacksplaining.com/owasp) link will lead you on over to the OWASP Top Ten list.

Scroll down to “6. Security Misconfiguration”. Read the paragraph explaining it, and fill in the blanks below.

Security misconfiguration is the most commonly seen issue. This is commonly a result of insecure \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_ or \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, and \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. Not only must all operating systems, frameworks, libraries, and applications be securely configured, but they must be \_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_ in a timely fashion.

Now hop on over to this [[Link](https://www.hacksplaining.com/exercises/lax-security-settings)] to learn a little more about the misconfiguration of the LAX Security Settings. Click on the icon to start the exercise. Answer the following questions as you read through the components.

1. What is an extreme cause of security holes that can be taken advantage of? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. What must be disabled by developers once they are in production? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. What can expose sensitive files to attackers? \_\_\_\_\_\_\_\_\_\_\_\_\_
4. List what must be turned off to configure a more secure system. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
5. What is recommended to be done if you allow team members to manage production data?\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
6. Explain what Content Delivery Networks (CDNs) are.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
7. Find the default username and password for JBoss on google to get access to the site.

Feel free to click on the orange icon to learn how to help prevent human error in configuration.

Next, go on over to [[Link](https://www.hacksplaining.com/exercises/information-leakage)] to learn about Information Leakage. Answer the following questions as you read through the components.

1. What is the first thing an attacker will try to find out about a website? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. What feature must you turn off to protect your website from attackers? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. What can also expose information to hackers, and what path should you avoid to prevent this? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
4. What type of page is important for storing error messages and AJAX responses.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
5. Uncover the user’s Social Security Number within the exercising.

**7. Cross-site Scripting (XSS)**

Head back to the [HACKSPLAINING](https://www.hacksplaining.com/owasp) link to jump on over to the OWASP Top Ten list.

Scroll down to “7. Cross-site Scripting (XXS)”. Read the paragraph explaining it, and fill in the blanks below.

XSS flaws occur whenever an application includes \_\_\_\_\_\_\_\_\_\_\_\_\_ in a new web page without proper validation or escaping, or \_\_\_\_\_\_\_\_\_\_ an existing web page with user-supplied data using a browser API that can create \_\_\_\_\_\_\_ or \_\_\_\_\_\_\_\_\_\_\_\_. XSS allows attackers to execute scripts in the victim’s browser which can \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, or \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Now, go on over to [[Link](https://www.hacksplaining.com/exercises/xss-stored)] to learn about Cross-site Scripting. Answer the following questions as you read through the components.

1. What do you have to be careful about constructing when building a website with a comment section? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. What can hackers inject into a comment section? \_\_\_\_\_\_\_\_\_\_\_\_
3. Using cross-site scripting, what can hackers steal? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
4. Describe what happens when your inject a script tag into the chat.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Once you have completed the exercise, click on the orange icon to learn about how to protect against Cross-site Scripting (XXS).

1. List the five things a hacker can implement by exploiting XSS? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. Read the “PROTECTION” section and write a quick summary of how to best protect a web application for XXS attacks? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Let's continue with Cross-site Scraping (XXS) by going to the [[Link](https://www.hacksplaining.com/exercises/xss-reflected)] about Rejected XXS.

1. What is another way attackers can inject malicious JavaScript using XXS? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. What can a hacker do once he gets its victim to click on its URL link? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Last thing we are going to talk about concerning Cross-site Scripting (XXS) is DOM-Based XXS. Click on the [[Link](https://www.hacksplaining.com/exercises/xss-dom)]

1. What is the vulnerability that is common when using URI fragments? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**8. Insecure Deserialization**

Go back on over to the [HACKSPLAINING](https://www.hacksplaining.com/owasp) OWASP Top Ten list.

Scroll down to “8. Insecure Deserialization”. Read the paragraph explaining it, and fill in the blanks below.

Insecure deserialization often leads to \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. Even if deserialization flaws do not result in remote code execution, they can be used to perform attacks, including \_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, and \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Once again we are going to click on the [[Link](https://www.hacksplaining.com/exercises/privilege-escalation)] to learn about Privilege Escalation.

1. What is privilege escalation? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. What is horizontal escalation? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. What is vertical escalation? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
4. Click on the orange click to read about how to protect against privilege escalation. Write a quick summary on what you read to protect web applications. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**9. Using Components with Known Vulnerabilities**

Just like before, we will be using the [HACKSPLAINING](https://www.hacksplaining.com/owasp) OWASP Top Ten list to fill in the blank below.

Scroll down to “9. Privilege Escalation”. Read the paragraph explaining it to fill in the blanks.

Components, such as \_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_, and other \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious \_\_\_\_\_\_\_\_\_\_\_ or \_\_\_\_\_\_\_\_\_\_\_\_\_. Applications and APIs using components with known vulnerabilities may undermine application \_\_\_\_\_\_\_\_\_\_\_\_ and enable various \_\_\_\_\_\_\_\_ and \_\_\_\_\_\_\_\_.

You know the deal, use the [[Link](https://www.hacksplaining.com/exercises/toxic-dependencies)] to do the Toxic Dependencies exercise.

1. What is Apache Struts? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. What are the vulnerabilities of Apache Struts 2? \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. What is Ruby on Rails? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ What are the vulnerabilities of Rails? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
4. XCodeGhost is designed to do what? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
5. What vulnerabilities were found in 2014 and 2020? \_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**10. Insufficient Logging & Monitoring**

One last time, go to the [HACKSPLAINING](https://www.hacksplaining.com/owasp) OWASP Top Ten list and fill in the blanks below.

Scroll down to “10. Insufficient Logging & Monitoring”. Read the paragraph explaining it, to fill in the blanks below.

Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, and \_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_, or \_\_\_\_\_\_\_\_\_ data. Most breach studies show time to detect a breach is over \_\_\_\_\_\_ days, typically detected by \_\_\_\_\_\_\_\_\_ parties rather than \_\_\_\_\_\_\_\_\_\_\_\_\_\_ processes or monitoring.

Use this final [[Link](https://www.hacksplaining.com/exercises/logging-and-monitoring)] of the activity to complete the Logging and Monitoring exercise.

1. What is logging? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. Each logging statement should have what data? \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. What should you not write in a log file? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
4. What does monitoring measure? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
5. What are other uses of monitoring?\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
6. Why do you need a response plan? What does it include?\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

YOU ARE DONE!!!