**Cybersecurity 401**

**Module 7 - Threat Hunting**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

## **Read 36 - XSS with w3af, DVWA**

## 

**| Rodrigo Brasil 12/2023 |**

[Cross-site scripting](https://portswigger.net/web-security/cross-site-scripting)

1. Explain how a cross-site scripting attack works in non-technical terms.
   1. an XSS attack takes advantage of a website's vulnerability to sneak in and run unauthorized code in a web browser. Which can compromise personal information or manipulate the user experience on the site.
2. What are the three types of XSS attacks?
   1. Reflected XSS, where the malicious script comes from the current HTTP request.
   2. Stored XSS, where the malicious script comes from the website's database.
   3. DOM-based XSS, where the vulnerability exists in client-side code rather than server-side code.
3. If an attacker successfully exploits a XSS vulnerability, what malicious actions would they be able to perform?
   1. Impersonate or masquerade as the victim user.
   2. Carry out any action that the user is able to perform.
   3. Read any data that the user is able to access.
   4. Capture the user's login credentials.
   5. Perform virtual defacement of the web site.
   6. Inject trojan functionality into the web site.
4. What are some security controls that can be implemented to prevent XSS attacks?
   1. Filter input on arrival. At the point where user input is received, filter as strictly as possible based on what is expected or valid input.
   2. Encode data on output. At the point where user-controllable data is output in HTTP responses, encode the output to prevent it from being interpreted as active content. Depending on the output context, this might require applying combinations of HTML, URL, JavaScript, and CSS encoding.
   3. Use appropriate response headers. To prevent XSS in HTTP responses that aren't intended to contain any HTML or JavaScript, you can use the Content-Type and X-Content-Type-Options headers to ensure that browsers interpret the responses in the way you intend.
   4. Content Security Policy. As a last line of defense, you can use Content Security Policy (CSP) to reduce the severity of any XSS vulnerabilities that still occur.