Week 1: HTML Injection & Cross-Site Scripting (XSS)

🔹 HTML Injection

What is it?

- Happens when user input is not sanitized, allowing injection of HTML tags.
- Used to modify webpage content, insert fake forms, or redirect users.

How to find it?

- Test input fields by injecting simple HTML tags like:

  Test

  <img src="https://evil.com/logo.png">

- Check if the input is reflected on the page as HTML instead of plain text.

How to prevent it?

- Encode special characters (< > " ' &) before displaying user input.
- Use **Content Security Policy (CSP)** to restrict unauthorized HTML execution.

🔹 Cross-Site Scripting (XSS)

What is it?

- Allows attackers to inject **JavaScript** into a webpage, affecting other users.
- Used to steal cookies, modify content, or perform phishing attacks.

How to find it?

- Try injecting JavaScript payloads in input fields or URL parameters:

  <script>alert('XSS Found')</script>

  <img src=x onerror=alert('XSS')>

- Look for inputs that reflect user data without sanitization.

How to prevent it?

- Sanitize user inputs by escaping special characters.
- Use **CSP** to block unauthorized script execution.
- Set **HttpOnly** and **Secure** flags on cookies to prevent theft.