CYBERSECURITY DAILY DIARY

DAY-26
DATE- 18 July, 2025

Topics Covered:

- XSS and input validation basics: test reflected XSS on a local vulnerable app (DVWA/Juice Shop).

What I Did Today

Reflected XSS (Cross-Site Scripting) occurs when user input is immediately echoed back in the response without proper sanitization. It's often found in search forms, error messages, or URL parameters.

- In DVWA (set to low security), I injected `<script>alert('XSS')</script>` into a search box and saw the alert pop up — confirming the vulnerability.
- Juice Shop had more realistic defenses, but some endpoints still reflected input — using payloads like `<img src=x onerror=alert(1)>` helped bypass basic filters.

Input Validation vs Output Encoding

- Input validation checks for acceptable characters and formats before processing — but it's not enough on its own.
- Output encoding ensures that special characters (like `<`, `>`, `"`) are rendered harmless in the browser — this is the real defense against XSS.

Testing Techniques

- Used Burp Suite to intercept and replay requests with modified payloads.
- Observed how different contexts (HTML, JavaScript, attributes) required different payload styles.
- Learned that reflected XSS is easier to spot than stored or DOM-based XSS — but still dangerous if exploited in phishing or redirection attacks.

Reflections

- Seeing a simple script pop up in the browser was oddly thrilling — like unlocking a hidden layer of the web.
- Realized how fragile web apps can be when they trust user input too much.
- Grateful for safe environments like DVWA and Juice Shop — they let me break things without consequences.