Short Notes – All Important XSS Concepts

1. What is XSS

- XSS = Cross-Site Scripting.

- Happens when untrusted input is inserted into a webpage without proper escaping.

- Allows attackers to run JS, read data, act as user, modify content.

2. Types of XSS

Reflected XSS:

- Input from URL is reflected immediately.

Stored XSS:

- Input stored in DB, logs, files, shown later in dangerous contexts.

DOM-Based XSS:

- Happens entirely in browser JS: Source → Processing → Sink.

3. How to Test XSS

- Use harmless marker like abc123XYZ.

- Find reflection points.

- Identify context.

- Check encoding/sanitization.

4. What XSS Can Be Used For

- Impersonate users

- Read data

- Capture credentials

- Modify content

5. Cookie Theft Limitations

- HttpOnly, SameSite, session binding, logout, short sessions reduce risk.

6. Password Autofill Limitations

- Autofill requires user action, domain-bound, hidden fields not autofilled.

7. XSS to Bypass CSRF

- XSS can read CSRF tokens and perform authenticated actions.

8. Dangling Markup Injection

- Breaks HTML attributes and leaks data through request-generating tags.

9. Content Security Policy (CSP)

Nonce-based CSP:

- Allows only scripts with correct nonce.

Hash-based CSP:

- Allows only exact script hashes.

Strict CSP:

- Blocks external scripts, images, frames.

Limitations:

- Images may still leak data.

- User interactions may trigger behavior.

10. Validation Strategies

Blacklist:

- Blocking keywords fails.

Whitelist:

- Allow only expected patterns; most secure.

11. Prevention Techniques

- Validate input

- Encode output per context

- Avoid innerHTML

- Use DOMPurify

- Use strict CSP

12. Differences

XSS vs CSRF:

- XSS executes JS; CSRF forces actions.

XSS vs SQL Injection:

- XSS targets users; SQLi targets DB.

13. Prevention in PHP & Java

PHP:

- Whitelist, htmlentities, JS Unicode escapes.

Java:

- Whitelist, Guava HTML encoding, JS Unicode escapes.