Web Application Security Testing Report – Task 1

Intern Name: Aditi Gupta

Tool Used: OWASP ZAP

Target Website:

http://testphp.vulnweb.com

Date: 17/07/2025

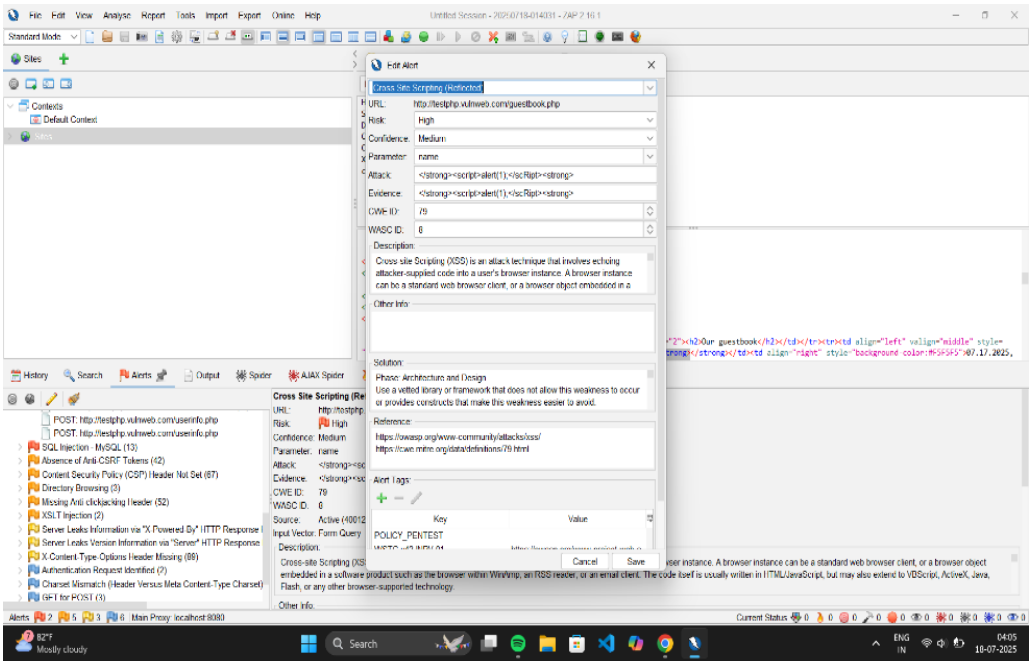
1. Cross-Site Scripting (Reflected)

- **Risk:** High
- **URL Affected:** http://testphp.vulnweb.com/guestbook.php
- **Parameter:** name
- **Evidence:**
  Payload reflected in output: <script>alert(1)</script>
- **Description:**
  Reflected Cross-Site Scripting (XSS) occurs when unsanitized user input is immediately echoed back in the browser without proper validation or encoding. This allows attackers to inject malicious JavaScript, which can hijack user sessions, steal cookies, or modify site content.
- **Mitigation:**
  - Validate and sanitize all user input
  - Use secure encoding libraries (e.g., OWASP ESAPI)
  - Set HttpOnly and Secure flags on cookies
  - Avoid reflecting raw user input in HTML output

OWASP Mapping:

🔹 Falls under **A03: Injection** in the OWASP Top 10 (2021).

XSS is a type of injection where untrusted user input is interpreted as executable script code by the browser.



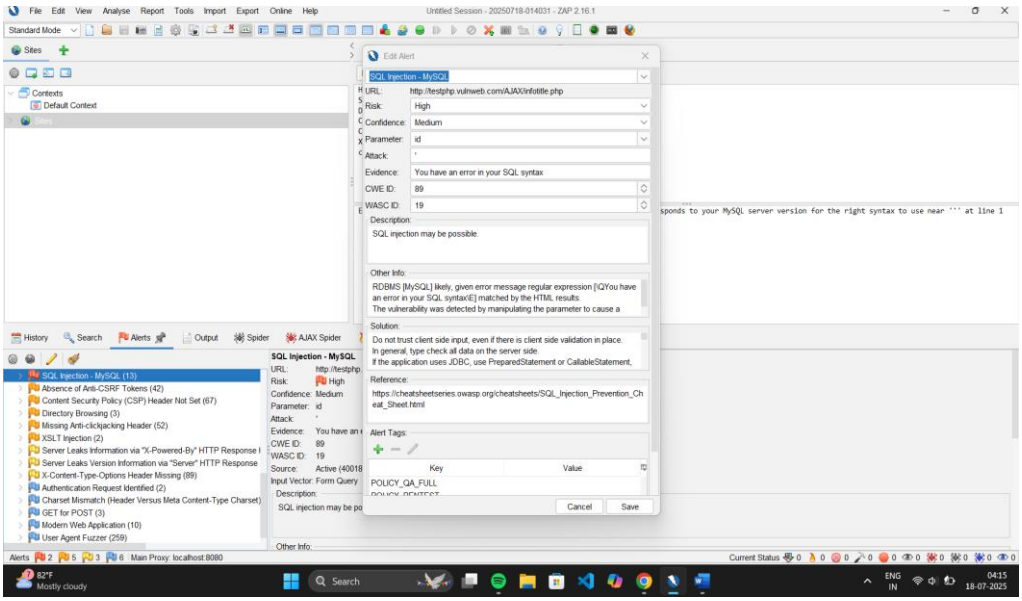
2. SQL Injection – MySQL

- **Risk:** High
- **URL Affected:** http://testphp.vulnweb.com/AJAX/infotitle.php
- **Parameter:** id
- **Evidence:**
  Error response: *"You have an error in your SQL syntax..."*
- **Description:**
  SQL Injection occurs when input from the user is directly embedded into SQL queries without proper sanitization. This can allow attackers to manipulate queries, access unauthorized data, or compromise the entire database.
- **Mitigation:**
  - Use **parameterized queries** (e.g., PreparedStatements in JDBC)
  - Avoid using string concatenation for query building
  - Apply strict **input validation** (whitelisting approach)
  - Grant the **least privileges** to database users

OWASP Mapping:

- Falls under **A03: Injection** in the OWASP Top 10 (2021).

SQLi is a classic injection vulnerability affecting database queries and data integrity.



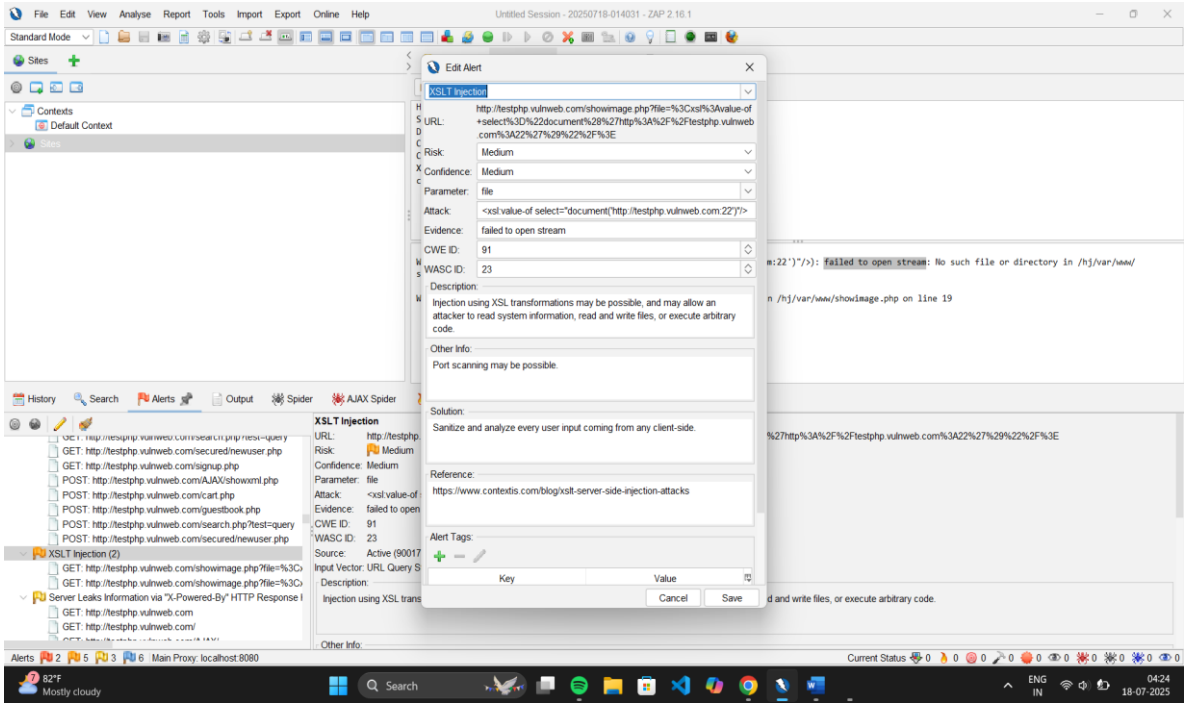
3. XSLT Injection

- **Risk:** Medium
- **URL Affected:** http://testphp.vulnweb.com/showimage.php?file=...
- **Parameter:** file
- **Evidence:**
  Error shown: *"failed to open stream: No such file or directory"*
- **Description:**
  XSLT Injection happens when attackers are able to inject malicious XSLT code into a system that processes user-supplied XML. This may allow attackers to read server-side files, make internal HTTP requests, or execute arbitrary logic via XSL transformations.
- **Mitigation:**
  - Sanitize and validate **all XML input**
  - Disable external entity processing in XML parsers
  - Avoid dynamic file referencing in XSLT unless necessary
  - Use secure parser configurations

OWASP Mapping:

- Falls under **A03: Injection** in the OWASP Top 10 (2021).

XSLT injection is a specialized form of injection targeting XML/XSL processing.


File Edit View Analyse Report Tools Import Export Online Help
Untitled Session - 20250718-014031 - ZAP 2.16.1
Standard Mode
Sites
Contexts
Default Context
Edit Alert
XSLT Injection
URL: http://testphp.vulnweb.com/showimage.php?file=%3Cxsl%3Avalue-of+select%3D%22document%28%27http%3A%2F%2Ftestphp.vulnweb.com%3A22%27%29%22%2F%3E
Risk: Medium
Confidence: Medium
Parameter: file
Attack: <xsl:value-of select="document('http://testphp.vulnweb.com:22')"/>
Evidence: failed to open stream
CWE ID: 91
WASC ID: 23
Description:
Injection using XSL transformations may be possible, and may allow an attacker to read system information, read and write files, or execute arbitrary code.
Other Info:
Port scanning may be possible.
Solution:
Sanitize and analyze every user input coming from any client-side.
Reference:
https://www.contextis.com/blog/xslt-server-side-injection-attacks
Alert Tags:
Key
Value
Cancel
Save
History Search Alerts Output Spider AJAX Spider
GET: http://testphp.vulnweb.com/secured/newuser.php
GET: http://testphp.vulnweb.com/signup.php
POST: http://testphp.vulnweb.com/AJAX/showxml.php
POST: http://testphp.vulnweb.com/cart.php
POST: http://testphp.vulnweb.com/guestbook.php
POST: http://testphp.vulnweb.com/search.php?test=query
POST: http://testphp.vulnweb.com/secured/newuser.php
XSLT Injection (2)
Server Leaks Information via "X-Powered-By" HTTP Response
GET: http://testphp.vulnweb.com
GET: http://testphp.vulnweb.com/
Alerts 2 5 3 6 Main Proxy: localhost:8080
Current Status
82°F
Mostly cloudy
Search
ENG
IN
04:24
18-07-2025