VAPT

What is OWASP?

→ OWASP (Open Web Application Security Project)
→ A non-profit community that provides free resources for web security.

Most famous project = *OWASP TOP 10*
→ It lists 10 most common & critical vulnerabilities found in web applications.

Why OWASP TOP 10 is Important?

- Helps developers to write secure code
- Must-know for Bug Bounty Hunters
- Followed in Industry Pentesting
- Base for Certifications (OSCP, CEH, etc.)

OWASP TOP 10 (2021 Latest Version)

---

1. Broken Access Control (A01)

What it means?

→ When a user can access data or functionality which they should not have permission for.

Real Life Example:

- User changing URL:

/profile?user=1 → Changing to user=2

- Accessing Admin Panel directly:

/admin

How to Test Practically:

- IDOR Testing (Parameter Tampering)

- Force Browsing Hidden Pages
- Changing HTTP Methods (GET → POST / DELETE)

---

2. Cryptographic Failures (A02)

What it means?

→ Failure to protect sensitive data like Passwords, Credit Card info, Session IDs.

Real Life Example:

- Plain Text Password Storage
- Weak Encryption used
- Sensitive data over HTTP

How to Test Practically:

- Inspect Cookies (Burp Suite → Proxy → HTTP History)
- Check for JWT tokens
- Use jwt.io to decode tokens
- Check for HTTPS implementation

---

3. Injection (A03)

What it means?

→ When untrusted data gets executed as code.

Types of Injection:

- SQL Injection
- XSS (Cross Site Scripting)
- Command Injection

- LDAP Injection

Real Life Example:

```
' OR '1'='1 --
```

```
<script>alert(1)</script>
```

```
;ls -la
```

How to Test Practically:

- Input test payloads in:
  - Search Bars
  - Login Forms
  - URL Parameters
- Use Burp Intruder to automate payloads
- Look for errors or unexpected output

---

4. Insecure Design (A04)

What it means?

→ Flaws in business logic or application design.

Real Life Example:

- Payment Bypass
- Infinite API Usage without rate limiting
- Changing Price of Product from 1000 to 1

How to Test Practically:

- Try Business Logic Abuse

- Modify requests in Burp
- Try to skip payment steps
- Test workflows manually

---

5. Security Misconfiguration (A05)

What it means?

→ Default settings left in applications or servers.

Real Life Example:

- Exposed Admin Panels
- Exposed .git folders
- Verbose Error Messages
- phpinfo page exposed

How to Test Practically:

- Check Robots.txt
- Check for Debug Pages
- Access .env, config.php
- Use:

/admin

/phpinfo.php

/.git/config

/.env

---

6. Vulnerable & Outdated Components (A06)

What it means?

→ Using old libraries or software with known CVEs.

How to Test Practically:

- Check HTTP Response Headers:

Server: Apache/2.2.14 (2009)

- Check Libraries used in website
- Search CVEs at: https://cvedetails.com

---

7. Identification & Authentication Failures (A07)

What it means?

→ Weak authentication mechanisms.

Real Life Example:

- No account lockout
- Weak Password Policies
- Session Fixation
- User Enumeration on Login Error

How to Test Practically:

- Bruteforce using Burp Intruder
- Change Session ID after login/logout
- Observe error messages

---

8. Software & Data Integrity Failures (A08)

What it means?

→ No validation of data or software updates.

Real Life Example:

- Downloading files from untrusted source
- No checksum verification
- Tampered JavaScript files

How to Test Practically:

- Check CDN JS Files
- Try modifying JS files locally
- Check File Upload functionality

---

9. Security Logging & Monitoring Failures (A09)

What it means?

→ Failure to detect & respond to attacks.

Real Life Example:

- No logs for failed logins
- No alerts for sensitive events
- Silent SQLi attempts

How to Test Practically:

- Try Multiple Wrong Logins
- Try SQLi/XSS → Check if blocked/logged
- Check Password Reset without logs

---

10. Server-Side Request Forgery (SSRF) (A10)

What it means?

→ Forcing server to make requests on attacker’s behalf.

Real Life Example:

URL parameter like:

url=http://internal-ip:8080

url=http://169.254.169.254/latest/meta-data/

How to Test Practically:

- Look for URL Fetch features
- Test with:

file:///etc/passwd

http://localhost

http://burpcollaborator.net/abc

Bonus Practical Tips:

- Use Burp Suite's:
  - Intruder → For Automation
  - Repeater → Manual Testing
  - Decoder → JWT / Hash Decode
  - Scanner (Pro) → Auto Scan
- Use OWASP ZAP → Alternative to Burp
- Always check:
  - Request & Response
  - Headers
  - Cookies

- Parameters
- Hidden Fields

Final Pro Tip for Teaching:

OWASP Issue	**Mindset for Students**
Access Control	Think like an Insider
Crypto Failure	Think about Data Safety
Injection	Think about Dirty Inputs
Insecure Design	Think beyond Login-Logout
Misconfig	Think like Admin mistake
Outdated Components	Think Version Check
Auth Failure	Think Password Policy
Integrity Failure	Think File Tampering
Logging Failure	Think Silent Attacks
SSRF	Think Server Acting for You
