@hackmarfacles | https://github.com/grafdetectechnologies | https://in.linkedin.com/company/hackingarfacles

OWASP Testing Checklist

Information Gathering

- **WSTG-INFO-01** Conduct Search Engine Discovery Reconnaissance for Information Leakage
  - Identify what sensitive design and configuration information of the application, system, or organization is exposed directly (on the organization's website or indirectly via third party services)
  - Tools: Bing; Shodan; Google; Common Crawl; OWASP Top 10 - NA; CWE - NA
- **WSTG-INFO-02** Fingerprint Web Server
  - Determine the version and type of a running web server to enable further discovery of any known vulnerabilities
  - Tools: Netcraft; Nikto; Nmap; Zenmap; OWASP Top 10; CWE 710
- **WSTG-INFO-03** Review Webserver Metafiles for Information Leakage
  - Identify hidden or obfuscated paths and functionality through the analysis of metafiles files
  - Tools: Browser; curl; wget; Burp Suite; ZAP; OWASP Top 10; CWE 200
- **WSTG-INFO-04** Enumerate Applications on Webserver
  - Enumerate the applications within the scope that exist on a web server
  - Tools: Nmap; Nessus Vulnerability Scanner; Nikto; OWASP Top 10 - NA; CWE - NA
- **WSTG-INFO-05** Review Webpage Content for Information Leakage
  - Review webpage comments, metadata, and hidden fields to find any information leakage
  - Tools: Eyeballs; Curl; Zaproxy; Burp Suite; Wireshark; Google Maps API Scanner; OWASP Top 10; CWE 200; CWE 540
- **WSTG-INFO-06** Map Application Architecture
  - Understand the architecture of the application and the technologies in use.
  - Tools: Browser; Curl; wget; Nmap; OWASP Top 10; CWE - NA
- **WSTG-INFO-07** Fingerprint Web Application
  - Identify the web application and version to determine known vulnerabilities and the appropriate exploits.
  - Tools: Whatweb; Wmap; Wmfingerprint; Nmap; Nikto; Metasploit; CMSExtract; OWASP Top 10 - NA; CWE - NA
- **WSTG-INFO-08** Fingerprint Web Application Frameworks
  - Fingerprint the application being used by the web applications.
  - Tools: Wmfingerprint; Wmap; OWASP Top 10 - NA; CWE 1104
- **WSTG-INFO-09** Map Execution Paths Through Application
  - Map the target application and understand the principal workflows.
  - Tools: ZAP; Burp Suite; Curl; Spreadsheet software; Diagramming software; OWASP Top 10 - NA; CWE - NA
- **WSTG-INFO-10** Identify Application Entry Points
  - Identify possible entry and injection points through request and response analysis.
  - Tools: OWASP Zed Attack Proxy (ZAP); Burp Suite; Fiddler; OWASP Top 10 - NA; CWE - NA

Configuration and Deployment Management Testing

- **WSTG-CONF-01** Test Network Infrastructure Configuration
  - Review the application configurations set across the network and validate that they are not vulnerable.
  - Tools: Nessus; Nmap; OWASP Top 10; CWE 284; CWE 1188; CWE 1214
- **WSTG-CONF-02** Test Application Platform Configuration
  - Ensure that defaults and known files have been removed. Validate that no debugging code or extensions are left in the production environment.
  - Tools: Nessus; Nikto; OWASP Top 10; CWE 11; CWE 13; CWE 215; CWE 489; CWE 548; CWE 556; CWE 16
- **WSTG-CONF-03** Test File Extensions Handling for Sensitive Information
  - Dirbuster,omnihex file extensions, or extensions that might contain raw data (bq, scripts, raw data, credentials, etc.)
  - Tools: wget; curl; google for 'web mirroring tools'; OWASP Top 10; CWE 200; CWE 425; CWE 552
- **WSTG-CONF-04** Review Old Backup and Unreferenced Files for Sensitive Information
  - Find and analyze unreferenced files that might contain sensitive information.
  - Tools: Nessus; Nikto2; Wget for Windows; Sam Spade; Nmo; OWASP Top 10; CWE 200; CWE 538
- **WSTG-CONF-05** Enumerate Infrastructure and Application Admin Interfaces
  - Identify hidden operational interfaces and functionality.
  - Tools: OWASP ZAP - Forced Browse; DIRB; HTTPRINT; netpeeker; OWASP Top 10; CWE 284; CWE 434
- **WSTG-CONF-06** Test HTTP Methods
  - Enumerate supported HTTP methods. Test for access control bypass. Test HTTP method override techniques.
  - Tools: Ncat; cURL; Nmap http-methods NSE script; OWASP Top 10; CWE 650; CWE 748
- **WSTG-CONF-07** Test HTTP Strict Transport Security
  - Review the HSTS header and its validity.
  - Tools: Burp Proxy; ZAP; curl; OWASP Top 10; CWE 523
- **WSTG-CONF-08** Test RIA Cross Domain Policy
  - Analyze the permissions allowed from the policy files crossdomain.xml/clientaccesspolicy.xml and allowed access from.
  - Tools: Burp Proxy; ZAP; Nikto2; OWASP Top 10; CWE 942; CWE
- **WSTG-CONF-09** Test File Permission
  - Review and identify any rogue file permissions
  - Tools: Windows AccessChk; Windows AccessChk; Linux/unix command line; OWASP Top 10; CWE 552; CWE 732; CWE
- **WSTG-CONF-10** Test for Subdomain Takeover
  - Enumerate all possible domains (previous and current). Identify forgotten or misconfigured domains.
  - Tools: dig; host; nslookup; dialmycalls; subfinder; dnsscan; OWASP Top 10 - NA; CWE 672; CWE
- **WSTG-CONF-11** Test Cloud Storage
  - Assess that the access control configuration for the storage services is properly in place.
  - Tools: AWS CLI; OWASP Top 10; CWE 200; CWE
- **WSTG-CONF-12** Test for Content Security Policy
  - Review the Content-Security-Policy header or meta element to identify misconfigurations.
  - Tools: Google CSP Evaluator; CSP Auditor; CSP Generator; OWASP Top 10; CWE-1021; CWE
- **WSTG-CONF-13** Test for Path Confusion
  - Make sure application pains are configured correctly.
  - Tools: OWASP Zed Attack Proxy; Burp Suite; OWASP Top 10; CWE 436; CWE

Input Validation Testing

- **WSTG-INPV-01** Testing for Reflected Cross Site Scripting
  - Identify variables that are reflected in responses. Assess the input they accept and the encoding that gets applied on return (if any).
  - Tools: PCS; Hackvertor; Burp Proxy; OWASP Top 10; CWE 79
- **WSTG-INPV-02** Testing for Stored Cross Site Scripting
  - Identify stored input that is reflected on the client side. Assess the input they accept and the encoding that gets applied on return (if any).
  - Tools: PCS; Hackvertor; Burp; XSS Proxy; Burp Proxy; OWASP Top 10; CWE 79
- **WSTG-INPV-03** Testing for HTTP Splitting/Smuggling
  - Craft custom HTTP requests to test the other methods to bypass URL authentication and authorization.
  - Tools: netcat; OWASP Top 10 - NA; CWE - NA
- **WSTG-INPV-04** Testing for HTTP Parameter Pollution
  - Identify the backend and the parsing method used. Assess injection points and/or bypassing input filters using HPP.
  - Tools: OWASP ZAP Passive/Active Rules; OWASP Top 10; CWE 235; CWE 120
- **WSTG-INPV-05** Testing for SQL Injection
  - Identify SQL injection points. Assess the severity of the injection and the level of access that can be achieved through it.
  - Tools: Fuzzdb - wordlists; Bernardo Damele A. G.; sqlmap; automatic SQL injection tool; Mohammin Ziyaifaar- MySqlsnif; MySql Injection takeover tool; OWASP Top 10; CWE 89
- **WSTG-INPV-06** Testing Blind SQL Injection
  - Testing for Oracle
  - Oracle - NGS Squirrel; Pangolin Lanmaster Multiple DBMS SQL Injection tool; Reversinglabs - sqlbftools; Bernardo Damele A. G.; sqlmap; automatic SQL injection tool; Mohammin Ziyaifaar- MySql Injection takeover tool
  - Testing for MySQL
  - Testing for SQL Server
  - Testing for PostgreSQL
  - Testing for MS Access
  - Testing for NoSQL Injection
  - Testing for ORM Injection
  - Testing for Client-side
  - Testing for LDAP Injection
  - Identify LDAP injection points. Assess the severity of the injection.
  - Tools: Softerra LDAP Browser; OWASP Top 10; CWE 90
- **WSTG-INPV-07** Testing for XML Injection
  - Identify XML injection points. Assess the types of exploits that can be attained and their severities.
  - Tools: XML Injection Fuzz Strings (from wfuzz tool); OWASP Top 10; CWE 91; CWE 611
- **WSTG-INPV-08** Testing for SSI Injection
  - Identify SSI Injection points. Assess the severity of the injection.
  - Tools: Web Proxy; Burp Suite; OWASP Top 10; CWE 97
- **WSTG-INPV-09** Testing for IMAP/SMTP Injection
  - Identify IMAP/SMTP injection points. Understand the data flow and deployment structure of the system. Assess the injection risks.
  - Tools: Burp Proxy; ZAP; OWASP Top 10; CWE - 147
- **WSTG-INPV-10** Testing for Code Injection
  - Identify injection points where you can inject code into the application.
  - Tools: Burp Proxy; ZAP; Lifu; Fuzzdb; Jsploit; LFI Suite; OWASP Top 10
  - Testing for File Inclusion
  - Testing for Command Injection
  - Identify and assess the command injection points.
  - Tools: OWASP WebGoat; Comix; OWASP Top 10; CWE 77; CWE 78; CWE 434; CWE 1236
- **WSTG-INPV-11** Testing for Buffer Overflow
  - Testing for Heap Overflow
  - Testing for Stack Overflow
  - Testing for Format String Injection
  - Assess whether injecting format string conversion specifiers into user controlled fields causes unexpected behavior from the application.
  - Tools: OWASP ZAP; OWASP Top 10; CWE 134; CWE
- **WSTG-INPV-12** Testing for Incubated Vulnerability
  - Identify injections that are stored and require a recall step to the stored injection.
  - Tools: XSS proxy; OWASP Zed Attack Proxy (ZAP); Burp Suite; Metasploit; OWASP Top 10; CWE 79
- **WSTG-INPV-13** Testing for Mass Assignment
  - Assess if the application is vulnerable to spliting, identifying what possible attacks are achievable.
  - Tools: Burp Proxy; ZAP; curl; value; OWASP Top 10; CWE 915; CWE 444
- **WSTG-INPV-14** Testing for HTTP Incoming Requests
  - Monitor all incoming and outgoing HTTP requests to the Web Server to inspect any suspicious requests.
  - Tools: Fiddler; TCPMon; Charles Web Debugging Proxy; Wireshark; PowerEdit Plugin; proxytrace; netsparker; Zaproxy; OWASP Top 10 - NA; CWE - NA
- **WSTG-INPV-15** Testing for Host Header Injection
  - Assess if the Host header is being parsed dynamically in the application. Bypass security controls that rely on the header.
  - Tools: Burp Proxy; OWASP Top 10; CWE 644; CWE 116
- **WSTG-INPV-16** Testing for Server-side Template Injection
  - Detect template injection vulnerability points. Identify the templating engine.
  - Tools: Tplmap; Backslash Powered Scanner Burp Suite extension; Template expression test strings/payloads list; OWASP Top 10; CWE 1336; CWE
- **WSTG-INPV-17** Testing for Server Side Request Forgery
  - Identify SSRF injection points. Test if the injection points are exploitable.
  - Tools: OWASP Top 10; CWE 918; CWE
- **WSTG-INPV-18** Testing for Mass Assignment
  - Identify requests that modify objects. Assess if it is possible to modify fields, never intended to be modified from outside
  - Tools: OWASP Top 10; CWE 915; CWE

Authentication Testing

- **WSTG-ATHN-01** Testing for Credentials Transported over an Encrypted Channel
  - Check weather weather its HTTP or HTTPS. Sending data through HTTP and HTTPS.
  - Tools: Burp Proxy; ZAP; Loxo; Curl; OWASP Top 10 - NA; CWE - NA
- **WSTG-ATHN-02** Testing for Default Credentials
  - Determine whether the application has any user accounts with default passwords.
  - Tools: Burp Intruder; THC Hydra; Nikto 2; OWASP Top 10; CWE 1392; CWE
- **WSTG-ATHN-03** Testing for Weak Lock Out Mechanism
  - Evaluate the account lockout mechanism's ability to mitigate brute force password guessing.
  - Tools: Browser; OWASP Top 10; CWE 307
- **WSTG-ATHN-04** Testing for Bypassing Authentication Schema
  - Ensure that authentication applied across all services that require it.
  - Tools: Browser; WebGoat; OWASP Top 10; CWE 287; CWE 288; CWE 289; CWE 290; CWE 291; CWE 292; CWE 293; CWE 294; CWE 295; CWE 296; CWE 297; CWE 298; CWE 299; CWE 300; CWE 301; CWE 302; CWE 303; CWE 304; CWE 305; CWE 306; CWE 307; CWE 308; CWE 309; CWE 645; CWE 1390; CWE 1391; CWE 592; CWE 287
- **WSTG-ATHN-05** Testing for Vulnerable Remember Password
  - Validate that the remember session is managed securely and do not put the users credentials in danger.
  - Tools: OWASP ZAP; Burp Suite; OWASP Top 10 - NA; CWE 308; CWE 309; CWE 640; CWE 565; CWE 384; CWE 613; CWE 614; CWE 539; CWE 646; CWE 603; CWE 784; CWE 565; CWE 315; CWE 539; CWE 565; CWE 784; CWE 315; CWE 539; CWE 565; CWE 784; CWE 315; CWE 539; CWE 565; CWE 784