WSTG Notes

Table of Contents

[Recon: 8](#_Toc166311553)

[· Info 8](#_Toc166311554)

[· Inspect 9](#_Toc166311555)

[· View code source (Check links and urls) 9](#_Toc166311556)

[· Check functions 9](#_Toc166311557)

[· Decode VIEWSTAT by burn extension or other tools and check three methods to find vulnerability if app prog is ASP.net 9](#_Toc166311558)

[· Use browser extensions for detect vul such as hackbar,wapp 9](#_Toc166311559)

[· Secure Headers 9](#_Toc166311560)

[· Cookies 9](#_Toc166311561)

[· Find waf vendor 9](#_Toc166311562)

[Information Gathering 10](#_Toc166311563)

[**Conduct Search Engine Discovery Reconnaissance for Information Leakage** 10](#_Toc166311564)

[Tools: 10](#_Toc166311565)

[10](#_Toc166311566)

[**Fingerprint Web Server** 10](#_Toc166311567)

[Tools: 10](#_Toc166311568)

[10](#_Toc166311569)

[**Review Webserver Metafiles for Information Leakage** 10](#_Toc166311570)

[Tools: 10](#_Toc166311571)

[11](#_Toc166311572)

[**Enumerate Applications on Webserver** 11](#_Toc166311573)

[Tools: 11](#_Toc166311574)

[11](#_Toc166311575)

[**Review Web Page Content for Information Leakage** 11](#_Toc166311576)

[Tools: 11](#_Toc166311577)

[· Burpsuite 11](#_Toc166311578)

[· Gobuster 11](#_Toc166311579)

[· Dirbuster 11](#_Toc166311580)

[· Secure 11](#_Toc166311581)

[11](#_Toc166311582)

[**Identify Application Entry Points** 11](#_Toc166311583)

[Tools: 11](#_Toc166311584)

[· Burp 11](#_Toc166311585)

[· Zed attack proxy 11](#_Toc166311586)

[· SQLmap 11](#_Toc166311587)

[· Waf 11](#_Toc166311588)

[· Nikto 11](#_Toc166311589)

[· Nessus 12](#_Toc166311590)

[· Openvas 12](#_Toc166311591)

[· Attack surface detector (ASD) 12](#_Toc166311592)

[12](#_Toc166311593)

[**Map Execution Paths Through Application** 12](#_Toc166311594)

[Tools: 12](#_Toc166311595)

[12](#_Toc166311596)

[**Fingerprint Web Application Framework** 12](#_Toc166311597)

[Tools: 12](#_Toc166311598)

[· Burp 12](#_Toc166311599)

[· Zed 12](#_Toc166311600)

[· Nikto 12](#_Toc166311601)

[· Nmap 12](#_Toc166311602)

[· Nessus 12](#_Toc166311603)

[· Wireshark 12](#_Toc166311604)

[· Metasploit 12](#_Toc166311605)

[· Cain & abel 12](#_Toc166311606)

[· Sqlmap 12](#_Toc166311607)

[· Nexpose 12](#_Toc166311608)

[· Openvas 12](#_Toc166311609)

[· Nessus 12](#_Toc166311610)

[· Acunetix 12](#_Toc166311611)

[· W3af 13](#_Toc166311612)

[· Skipfish 13](#_Toc166311613)

[· BeEF 13](#_Toc166311614)

[· Owasp zap 13](#_Toc166311615)

[· Sql ninja 13](#_Toc166311616)

[· DSSS 13](#_Toc166311617)

[· Whatweb 13](#_Toc166311618)

[· BlindElephant 13](#_Toc166311619)

[· Wappalyzer 13](#_Toc166311620)

[· seuriBlE 13](#_Toc166311621)

[· PTE 13](#_Toc166311622)

[13](#_Toc166311623)

[**Fingerprint Web Application (Merged into WSTG-INFO-08)** 13](#_Toc166311624)

[Tools: 13](#_Toc166311625)

[Whatweb 13](#_Toc166311626)

[Wappalyzer 13](#_Toc166311627)

[Blindelephant 13](#_Toc166311628)

[Nikto 13](#_Toc166311629)

[W3af 13](#_Toc166311630)

[DSS 13](#_Toc166311631)

[DIRB 13](#_Toc166311632)

[Burp 13](#_Toc166311633)

[Zed 13](#_Toc166311634)

[SQLmap 13](#_Toc166311635)

[Nmap 13](#_Toc166311636)

[Nessus 13](#_Toc166311637)

[Wireshark 13](#_Toc166311638)

[metaspl 13](#_Toc166311639)

[13](#_Toc166311640)

[**Map Application Architecture** 14](#_Toc166311641)

[Configuration and Deployment Management Testing 14](#_Toc166311642)

[· Test Network Infrastructure Configuration 14](#_Toc166311643)

[· Test Application Platform Configuration 14](#_Toc166311644)

[· Test File Extensions Handling for Sensitive Information 14](#_Toc166311645)

[· Review Old Backup and Unreferenced Files for Sensitive Information 14](#_Toc166311646)

[· Enumerate Infrastructure and Application Admin Interfaces 14](#_Toc166311647)

[· Test HTTP Methods 14](#_Toc166311648)

[· Test HTTP Strict Transport Security 14](#_Toc166311649)

[· Test RIA Cross Domain Policy (Removed) 14](#_Toc166311650)

[· Test File Permission 14](#_Toc166311651)

[· Test for Subdomain Takeover 14](#_Toc166311652)

[· Test Cloud Storage 14](#_Toc166311653)

[· Test for Content Security Policy 14](#_Toc166311654)

[· Test for Path Confusion 14](#_Toc166311655)

[Identity Management Testing 14](#_Toc166311656)

[· Test Role Definitions 14](#_Toc166311657)

[· Test User Registration Process 14](#_Toc166311658)

[· Test Account Provisioning Process 14](#_Toc166311659)

[· Testing for Account Enumeration and Guessable User Account 14](#_Toc166311660)

[· Testing for Weak or Unenforced Username Policy 14](#_Toc166311661)

[Authentication Testing 15](#_Toc166311662)

[· Testing for Credentials Transported over an Encrypted Channel (Merged into WSTG-CRYP-03) 15](#_Toc166311663)

[· Testing for Default Credentials 15](#_Toc166311664)

[· Testing for Weak Lock Out Mechanism 15](#_Toc166311665)

[· Testing for Bypassing Authentication Schema 15](#_Toc166311666)

[· Testing for Vulnerable Remember Password 15](#_Toc166311667)

[· Testing for Browser Cache Weaknesses 15](#_Toc166311668)

[· Testing for Weak Password Policy 15](#_Toc166311669)

[· Testing for Weak Security Question Answer 15](#_Toc166311670)

[· Testing for Weak Password Change or Reset Functionalities 15](#_Toc166311671)

[· Testing for Weaker Authentication in Alternative Channel 15](#_Toc166311672)

[· Testing Multi-Factor Authentication 15](#_Toc166311673)

[Authorization Testing 15](#_Toc166311674)

[· Testing Directory Traversal File Include 15](#_Toc166311675)

[· Testing for Bypassing Authorization Schema 15](#_Toc166311676)

[· Testing for Privilege Escalation 15](#_Toc166311677)

[· Testing for Insecure Direct Object References 15](#_Toc166311678)

[· Testing for OAuth Weaknesses 15](#_Toc166311679)

[Session Management Testing 15](#_Toc166311680)

[· Testing for Session Management Schema 15](#_Toc166311681)

[· Testing for Cookies Attributes 15](#_Toc166311682)

[· Testing for Session Fixation 15](#_Toc166311683)

[· Testing for Exposed Session Variables 15](#_Toc166311684)

[· Testing for Cross Site Request Forgery 15](#_Toc166311685)

[· Testing for Logout Functionality 15](#_Toc166311686)

[· Testing Session Timeout 15](#_Toc166311687)

[· Testing for Session Puzzling 15](#_Toc166311688)

[· Testing for Session Hijacking 15](#_Toc166311689)

[· Testing JSON Web Tokens 16](#_Toc166311690)

[Input Validation Testing 16](#_Toc166311691)

[· Testing for Reflected Cross Site Scripting 16](#_Toc166311692)

[· Testing for Stored Cross Site Scripting 16](#_Toc166311693)

[· Testing for HTTP Verb Tampering (Merged into WSTG-CONF-06) 16](#_Toc166311694)

[· Testing for HTTP Parameter Pollution 16](#_Toc166311695)

[· Testing for SQL Injection 16](#_Toc166311696)

[· Testing for LDAP Injection 16](#_Toc166311697)

[· Testing for XML Injection 16](#_Toc166311698)

[· Testing for SSI Injection 16](#_Toc166311699)

[· Testing for XPath Injection 16](#_Toc166311700)

[· Testing for IMAP SMTP Injection 16](#_Toc166311701)

[· Testing for Code Injection 16](#_Toc166311702)

[· Testing for Command Injection 16](#_Toc166311703)

[· Testing for Format String Injection 16](#_Toc166311704)

[· Testing for Incubated Vulnerability 16](#_Toc166311705)

[· Testing for HTTP Splitting Smuggling 16](#_Toc166311706)

[· Testing for HTTP Incoming Requests 16](#_Toc166311707)

[· Testing for Host Header Injection 16](#_Toc166311708)

[· Testing for Server-side Template Injection 16](#_Toc166311709)

[· Testing for Server-Side Request Forgery 16](#_Toc166311710)

[· Testing for Mass Assignment 16](#_Toc166311711)

[Testing for Error Handling 16](#_Toc166311712)

[· Testing for Improper Error Handling 16](#_Toc166311713)

[· Testing for Stack Traces (Merged into WSTG-ERRH-01) 16](#_Toc166311714)

[Testing for Weak Cryptography 16](#_Toc166311715)

[· Testing for Weak Transport Layer Security 16](#_Toc166311716)

[· Testing for Padding Oracle 16](#_Toc166311717)

[· Testing for Sensitive Information Sent via Unencrypted Channels 16](#_Toc166311718)

[· Testing for Weak Encryption 17](#_Toc166311719)

[Business Logic Testing 17](#_Toc166311720)

[· Introduction to Business Logic 17](#_Toc166311721)

[· Test Business Logic Data Validation 17](#_Toc166311722)

[· Test Ability to Forge Requests 17](#_Toc166311723)

[· Test Integrity Checks 17](#_Toc166311724)

[· Test for Process Timing 17](#_Toc166311725)

[· Test Number of Times a Function Can Be Used Limits 17](#_Toc166311726)

[· Testing for the Circumvention of Work Flows 17](#_Toc166311727)

[· Test Defenses Against Application Misuse 17](#_Toc166311728)

[· Test Upload of Unexpected File Types 17](#_Toc166311729)

[· Test Upload of Malicious Files 17](#_Toc166311730)

[· Test Payment Functionality 17](#_Toc166311731)

[Client Side Testing 17](#_Toc166311732)

[· Testing for DOM-Based Cross Site Scripting 17](#_Toc166311733)

[Common source 17](#_Toc166311734)

[o Document.URL 17](#_Toc166311735)

[o Document.documentURI 17](#_Toc166311736)

[o Document.URLUnencoded 17](#_Toc166311737)

[o Document.baseURI 17](#_Toc166311738)

[o Location 17](#_Toc166311739)

[o Document.cookie 17](#_Toc166311740)

[o Document.rererrer 17](#_Toc166311741)

[o Windows.name 17](#_Toc166311742)

[o History.pushstate 17](#_Toc166311743)

[o History.replacestate 17](#_Toc166311744)

[o Localstorage 17](#_Toc166311745)

[o Sessionstorage 17](#_Toc166311746)

[o IndexedDB 17](#_Toc166311747)

[**Sink to lead DOM-XSS:** 17](#_Toc166311748)

[DOM XSS 18](#_Toc166311749)

[document.write() 18](#_Toc166311750)

[Open redirection 18](#_Toc166311751)

[windows.location 18](#_Toc166311752)

[Cookie manipulation 18](#_Toc166311753)

[document.cookie 18](#_Toc166311754)

[Javascript injection 18](#_Toc166311755)

[eval() 18](#_Toc166311756)

[Document-domain manipulation 18](#_Toc166311757)

[document.domain 18](#_Toc166311758)

[Websocket-URL poisoning 18](#_Toc166311759)

[websocket() 18](#_Toc166311760)

[Link manipulation 18](#_Toc166311761)

[element.src 18](#_Toc166311762)

[Web message manipulation 18](#_Toc166311763)

[postmessage() 18](#_Toc166311764)

[Ajax request-header manipulation 18](#_Toc166311765)

[setResuqestHeader() 18](#_Toc166311766)

[Local file-path manipulation 18](#_Toc166311767)

[FileReader.readAsText() 18](#_Toc166311768)

[Client-side SQl injection 18](#_Toc166311769)

[ExecuteSl() 18](#_Toc166311770)

[HTML5-storage manipulation 18](#_Toc166311771)

[sessionStorage.setItem() 18](#_Toc166311772)

[Client-side XPath injection 18](#_Toc166311773)

[document.evaluate() 18](#_Toc166311774)

[Client-side json injection 18](#_Toc166311775)

[JSON.parse() 18](#_Toc166311776)

[DOM-datamanipulation 18](#_Toc166311777)

[element.setAttribute() 18](#_Toc166311778)

[· Testing for JavaScript Execution 18](#_Toc166311779)

[18](#_Toc166311780)

[· Testing for HTML Injection 18](#_Toc166311781)

[· Testing for Client-side URL Redirect 18](#_Toc166311782)

[· Testing for CSS Injection 18](#_Toc166311783)

[· Testing for Client-side Resource Manipulation 18](#_Toc166311784)

[· Testing Cross Origin Resource Sharing 18](#_Toc166311785)

[· Testing for Cross Site Flashing 18](#_Toc166311786)

[· Testing for Clickjacking 18](#_Toc166311787)

[· Testing WebSockets 18](#_Toc166311788)

[· Testing Web Messaging 18](#_Toc166311789)

[· Testing Browser Storage 18](#_Toc166311790)

[· Testing for Cross Site Script Inclusion 18](#_Toc166311791)

[· Testing for Reverse Tabnabbing 18](#_Toc166311792)

[API Testing 18](#_Toc166311793)

[· Testing GraphQL 18](#_Toc166311794)

# Recon:

* Info

|  |  |  |
| --- | --- | --- |
| url | internal | External |
|  |  |  |

* Inspect
* View code source (Check links and urls)
* Check functions
* Decode VIEWSTAT by burn extension or other tools and check three methods to find vulnerability if app prog is ASP.net
* Use browser extensions for detect vul such as hackbar,wapp

|  |  |  |
| --- | --- | --- |
| technologies | type | info |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

* Secure Headers
* Cookies
* Find waf vendor
  + Shodan & cencys
  + Search SPF records and TXT records
  + Explore your target site in the browser
  + Analyze the HTTP headers or cookies (e.g. Cloudflare set cf\_clearance cookie)
  + If none of the step above helped you figure out the WAF, try to perform an automated request to your target page. Use an HTTP client and make a GET request. The response produced by the server may provide useful data

|  |  |  |
| --- | --- | --- |
| WAF | cookies | Info |
| cloudflare | cf\_clearance |  |
| Akamai | AkamaiHttpServer |  |
| Incapsula | Incap\_ses\_${SiteID}\_${SessionID} |  |
| Imperva | IN-FLX |  |
| F5 | BIGipServer |  |
| AWS WAF | X-Amzen-Trace-Id |  |

# Information Gathering

**Conduct Search Engine Discovery Reconnaissance for Information Leakage**

Tools:

* Google Dorks
* Bing Search
* Yandex search
* DuckDuckGo search

**Fingerprint Web Server**

Tools:

* Nmap
* Whatweb
* Nikto
* Wappalyzer
* Nessus

**Review Webserver Metafiles for Information Leakage**

Tools:

* Dirbuster
* Burp suite
* Gobuster
* Secure
* Fimap

**Enumerate Applications on Webserver**

Tools:

* Nmap
* Nikto
* Wappalyzer
* Dirbuster
* Whatweb

**Review Web Page Content for Information Leakage**

Tools:

* Burpsuite
* Gobuster
* Dirbuster
* Secure
* Fimap

**Identify Application Entry Points**

Tools:

* Burp
* Zed attack proxy
* SQLmap
* Waf
* Nikto
* Nessus
* Openvas
* Attack surface detector (ASD)

**Map Execution Paths Through Application**

Tools:

* DAST tools such as burp,zap,nikto
* IAST tools such as contrast security and hdiv
* Sca tools such as black duck by synipsys and whitesource
* SAST tools such as SonaQube and Fortify
* Fuzz testing tools such as fuzzer
* Binary analysis tools such as Binary ninja and radare 2
* Debugging tools as GBD and LLDB
* Network protocol analyzers such as wireshark and OpenVAS
* Peneteration testing framework such as Metasploit and kali

**Fingerprint Web Application Framework**

Tools:

* Burp
* Zed
* Nikto
* Nmap
* Nessus
* Wireshark
* Metasploit
* Cain & abel
* Sqlmap
* Nexpose
* Openvas
* Nessus
* Acunetix
* W3af
* Skipfish
* BeEF
* Owasp zap
* Sql ninja
* DSSS
* Whatweb
* BlindElephant
* Wappalyzer
* seuriBlE
* PTE

**Fingerprint Web Application (Merged into WSTG-INFO-08)**

Tools:

Whatweb

Wappalyzer

Blindelephant

Nikto

W3af

DSS

DIRB

Burp

Zed

SQLmap

Nmap

Nessus

Wireshark

metaspl

**Map Application Architecture**

# Configuration and Deployment Management Testing

* Test Network Infrastructure Configuration
* Test Application Platform Configuration
* Test File Extensions Handling for Sensitive Information
* Review Old Backup and Unreferenced Files for Sensitive Information
* Enumerate Infrastructure and Application Admin Interfaces
* Test HTTP Methods
* Test HTTP Strict Transport Security
* Test RIA Cross Domain Policy (Removed)
* Test File Permission
* Test for Subdomain Takeover
* Test Cloud Storage
* Test for Content Security Policy
* Test for Path Confusion

# Identity Management Testing

* Test Role Definitions
* Test User Registration Process
* Test Account Provisioning Process
* Testing for Account Enumeration and Guessable User Account
* Testing for Weak or Unenforced Username Policy

# Authentication Testing

* Testing for Credentials Transported over an Encrypted Channel (Merged into WSTG-CRYP-03)
* Testing for Default Credentials
* Testing for Weak Lock Out Mechanism
* Testing for Bypassing Authentication Schema
* Testing for Vulnerable Remember Password
* Testing for Browser Cache Weaknesses
* Testing for Weak Password Policy
* Testing for Weak Security Question Answer
* Testing for Weak Password Change or Reset Functionalities
* Testing for Weaker Authentication in Alternative Channel
* Testing Multi-Factor Authentication

# Authorization Testing

* Testing Directory Traversal File Include
* Testing for Bypassing Authorization Schema
* Testing for Privilege Escalation
* Testing for Insecure Direct Object References
* Testing for OAuth Weaknesses

# Session Management Testing

* Testing for Session Management Schema
* Testing for Cookies Attributes
* Testing for Session Fixation
* Testing for Exposed Session Variables
* Testing for Cross Site Request Forgery
* Testing for Logout Functionality
* Testing Session Timeout
* Testing for Session Puzzling
* Testing for Session Hijacking
* Testing JSON Web Tokens

# Input Validation Testing

* Testing for Reflected Cross Site Scripting
* Testing for Stored Cross Site Scripting
* Testing for HTTP Verb Tampering (Merged into WSTG-CONF-06)
* Testing for HTTP Parameter Pollution
* Testing for SQL Injection
* Testing for LDAP Injection
* Testing for XML Injection
* Testing for SSI Injection
* Testing for XPath Injection
* Testing for IMAP SMTP Injection
* Testing for Code Injection
* Testing for Command Injection
* Testing for Format String Injection
* Testing for Incubated Vulnerability
* Testing for HTTP Splitting Smuggling
* Testing for HTTP Incoming Requests
* Testing for Host Header Injection
* Testing for Server-side Template Injection
* Testing for Server-Side Request Forgery
* Testing for Mass Assignment

# Testing for Error Handling

* Testing for Improper Error Handling
* Testing for Stack Traces (Merged into WSTG-ERRH-01)

# Testing for Weak Cryptography

* Testing for Weak Transport Layer Security
* Testing for Padding Oracle
* Testing for Sensitive Information Sent via Unencrypted Channels
* Testing for Weak Encryption

# Business Logic Testing

* Introduction to Business Logic
* Test Business Logic Data Validation
* Test Ability to Forge Requests
* Test Integrity Checks
* Test for Process Timing
* Test Number of Times a Function Can Be Used Limits
* Testing for the Circumvention of Work Flows
* Test Defenses Against Application Misuse
* Test Upload of Unexpected File Types
* Test Upload of Malicious Files
* Test Payment Functionality

# Client Side Testing

* Testing for DOM-Based Cross Site Scripting

Common source

* + Document.URL
  + Document.documentURI
  + Document.URLUnencoded
  + Document.baseURI
  + Location
  + Document.cookie
  + Document.rererrer
  + Windows.name
  + History.pushstate
  + History.replacestate
  + Localstorage
  + Sessionstorage
  + IndexedDB

**Sink to lead DOM-XSS:**

|  |  |
| --- | --- |
| DOM XSS | document.write() |
| Open redirection | windows.location |
| Cookie manipulation | document.cookie |
| Javascript injection | eval() |
| Document-domain manipulation | document.domain |
| Websocket-URL poisoning | websocket() |
| Link manipulation | element.src |
| Web message manipulation | postmessage() |
| Ajax request-header manipulation | setResuqestHeader() |
| Local file-path manipulation | FileReader.readAsText() |
| Client-side SQl injection | ExecuteSl() |
| HTML5-storage manipulation | sessionStorage.setItem() |
| Client-side XPath injection | document.evaluate() |
| Client-side json injection | JSON.parse() |
| DOM-datamanipulation | element.setAttribute() |

* Testing for JavaScript Execution

* Testing for HTML Injection
* Testing for Client-side URL Redirect
* Testing for CSS Injection
* Testing for Client-side Resource Manipulation
* Testing Cross Origin Resource Sharing
* Testing for Cross Site Flashing
* Testing for Clickjacking
* Testing WebSockets
* Testing Web Messaging
* Testing Browser Storage
* Testing for Cross Site Script Inclusion
* Testing for Reverse Tabnabbing

# API Testing

* Testing GraphQL