**XXX Company**

Report on the Provision of Penetration Test Service for a1 Web Application

April 30 2025

**Report on the Provision of Penetration Test Services for XXX's application**

Dear Sirs/ Madams,

Our assessment was performed from April 10 2025 to and follow-up verification was not completed at the time of report issuance. We are pleased to enclose the report of our findings and recommendations from our testing.

Copies of this report in its entirety may be made available to XXX’s system management vendors provided it is made clear to such recipients that we accept no responsibility to them in respect thereof. This report must not be made available or copied in whole or in part to any other person without our express written permission.

Our work was limited to the specific procedures and analysis described herein and was based only on the information made available up to the date of this Report. Accordingly, changes in circumstances after this date could affect the findings outlined in this Report and we reserve the right to amend findings, conclusions or recommendations, if necessary, based on factual information that comes to our attention after that date.

This report has been prepared solely for the internal use of XXX’s management. The report shall not be used for any other purpose other than that described above. The information contained in the report remains confidential to XXX. It shall not be distributed or quoted, in whole or in part, to any other party without prior written consent from us. We do not assume responsibility nor liability for any losses suffered by XXX, third party and/or any user as a result of the circulation, publication, reproduction or other use of the report contrary to the provisions of this section.

Given the inherent limitations in any system of control, projection of any evaluation of the controls to future periods is subject to the risk that the control procedures may become inadequate because of changes in systems, conditions or the degree of compliance with those procedures. Therefore, constant monitoring and proactive actions are needed to ensure that system controls that exist remain effective over time.

Yours sincerely.

TABLE OF CONTENTS

[1. Executive Summary](#_Toc4443661)

[Introduction](#_Toc4443662)

[Key Findings](#_Toc4443663)

[Summary of Findings](#_Toc4443664)

[2. Scope Of Work](#_Toc4443665)

[3. List Function/API Pentest ${SUMARY1}](#_Toc4443666)

[4. Sources of information](#_Toc4443667)

[5. Limitations and Restrictions](#_Toc4443668)

[Limitations of Controls](#_Toc4443669)

[Limitations of Penetration Tests](#_Toc44436610)

[Appendix A – Detailed Findings and Recommendations](#_Toc44436611)

[1. INSECURE DIRECT OBJECT REFERENCES (Manipulating other users' data)](#_Toc44436612)

[2. UNRESTRICTED FILE UPLOAD](#_Toc44436613)

[3. BUSINESS LOGIC DATA VALIDATIO](#_Toc44436614)

[4. CROSS SITE SCRIPTING (XSS)](#_Toc44436615)

[5. INSECURE DIRECT OBJECT REFERENCES (Viewing other users' data)](#_Toc44436616)

[6. SECURITY MISCONFIGURATION](#_Toc44436617)

[APPENDIX B – Determining the Severity of the Risk:](#_Toc44436618)

[1. CVSS v3.1 Equations Base Score](#_Toc44436619)

[2. Exploitability Metrics:](#_Toc44436620)

[3. Impact Metrics](#_Toc44436621)

[APPENDIX C – OUR APPROACH AND METHODOLOGY](#_Toc44436622)

[1. Our penetration testing methodology](#_Toc44436623)

[2. Secure code review methodology overview](#_Toc44436624)

[3. Penetration testing checklist](#_Toc44436625)

# Executive Summary

## Introduction

We, Information security management (hereinafter known as “ISM”), have completed the Penetration Test Services (“the Assessment”) for a1in accordance with our statement of work. Our assessment was performed in April 2025 and follow-up verification was not completed at the time of report issuance.

The objective of the Assessment was to identify technical security weaknesses on the applications and systems in the XXX’s networks. Practical recommendations were subsequently provided to assist XXX in addressing the identified gaps.

During the course of the work, we have discussed our findings and recommendations with the XXX IT team.

## Key Findings

From the assessment, we have identified a total of **2** Issues. Our findings are summarized below and the detailed findings can be found in **Appendix A**. We have discussed our findings with the relevant XXX staff and their comments are incorporated into this report.

## Summary of Findings

The total number of findings and their associated risk ratings are summarised below.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Area of Work** | **Risk Rating** | | | | | **Total** |
| Critical | High | Medium | Low | Recommend |
| **a1** Web Application Penetration Testing | **0** | **1** | **1** | **0** | **0** | **2** |
| **Total** | **0** | **1** | **1** | **0** | **0** | **2** |

The descriptions of the risk ratings are as follows:

|  |  |
| --- | --- |
| **Risk Rating** | **Description** |
|
| **Critical** | The item mentioned is a requirement to be followed as it has a significant impact to controls, finance or operations. We highly recommend that it be complied with and addressed immediately. |
| **High** | The item mentioned is a requirement to be followed as it has a significant impact to controls, finance or operations. We highly recommend that it be complied with and addressed immediately. |
| **Medium** | The item mentioned may be a requirement based on regulatory requirements and / or industry good practices, and has a moderate impact on controls, finance or operations. Compensating controls should be put in place if they are not implemented. |
| **Low** | The item mentioned has a low impact to controls, finance or operations if left uncorrected, but should be followed-up and evaluated as an opportunity for improvement. |
| **Recommend** | The item mentioned will ensure the application more safety and secure |

# Scope Of Work

|  |  |
| --- | --- |
| **Area of Work** | **Work Performed** |
|
| a1 Web Application Penetration Test | We have tested the application in UAT environment using both automated and manual approach. The scope of the assessment was limited to the following URLs:  a1.a.com |

# List Function/API Pentest

**Scope is not provided.**

# Sources of information

In carrying out this work, we relied on information and material provided to us by XXX. Our work was based on the examination of relevant materials made available to us. We also interviewed and discussed our work with relevant personnel including the business unit and application team to confirm our understanding of the various work areas.

# Limitations and Restrictions

## Limitations of Controls

Given the inherent limitations in any system of control, projection of any evaluation of the controls to future periods is subject to the risk that the control procedures may become inadequate because of changes in systems, conditions or the degree of compliance with those procedures. Therefore, constant monitoring is needed to ensure that system controls that exist remain effective over time.

## Limitations of Penetration Tests

Penetration test is conducted over a limited period and are performed on the system at a single point of time. As such, the scope was limited to current known vulnerabilities and current system configuration during the work period. The penetration tests and vulnerability assessments may not yield any vulnerability, which does not indicate that the system has no vulnerability exposure. New vulnerabilities may be discovered over time and therefore continuous and timely measures should be taken to address new vulnerabilities.

# Appendix A – Detailed Findings and Recommendations

|  |  |  |  |
| --- | --- | --- | --- |
| **Ref.** | **Issue Description** | **Risk Rating** | **Status** |
| Application Penetration Testing on a1 | | | |
| A1-01 | SQL Injection | **High** | Open |
| A1-02 | information disclosure | **Medium** | Open |

1. SQL Injection

|  |  |
| --- | --- |
| **Title: SQL Injection**  **Component: Web Application** | **Ref:**  **A1-01** |
| **Risk assessment score (CVSS v3.1)**  SCORE:  **7.3**  VECTOR: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:L | **Risk Rating:**  **High** |
| **Description**  sdsdss  **Likelihood:** **Recommend**  Attack Vector (AV): Network  Attack Complexity (AC): Low  Privileges Required (PR): None  User Interaction (UI): None  **Impact: Recommend**  – vuln.impact | |
| **Evidence**  hình ảnh lỗi | |
| **Affected URL**   |  |  |  |  | | --- | --- | --- | --- | | **No** | **Function** | **API** | **Params** | | 1 | dsds | /api/sds | d | | |
| **Recommendations** | |
| **References** | |
| **ISM Follow-up**  Status: Open | |

1. information disclosure

|  |  |
| --- | --- |
| **Title: information disclosure**  **Component: Web Application** | **Ref:**  **A1-02** |
| **Risk assessment score (CVSS v3.1)**  SCORE:  **6.3**  VECTOR: CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:L/A:L | **Risk Rating:**  **Medium** |
| **Description**  Information disclosure, also known as information leakage, is when a website unintentionally reveals sensitive information to its users. Depending on the context, websites may leak all kinds of information to a potential attacker, including:     - Data about other users, such as usernames or financial information   - Sensitive commercial or business data   - Technical details about the website and its infrastructure  **Likelihood:** **Medium**  Attack Vector (AV): Network  Attack Complexity (AC): Low  Privileges Required (PR): Low  User Interaction (UI): None  **Impact: Medium**  Information disclosure vulnerabilities can have both a direct and indirect impact depending on the purpose of the website and, therefore, what information an attacker is able to obtain. In some cases, the act of disclosing sensitive information alone can have a high impact on the affected parties. For example, an online shop leaking its customers' credit card details is likely to have severe consequences.    On the other hand, leaking technical information, such as the directory structure or which third-party frameworks are being used, may have little to no direct impact. However, in the wrong hands, this could be the key information required to construct any number of other exploits. The severity in this case depends on what the attacker is able to do with this information. – vuln.impact | |
| **Evidence**        (Image 1.1 Mô tả hình ảnh)      (Image 1.2 evidence này) | |
| **Affected URL**  No affected URLs listed. | |
| **Recommendations**  Preventing information disclosure completely is tricky due to the huge variety of ways in which it can occur. However, there are some general best practices that you can follow to minimize the risk of these kinds of vulnerability creeping into your own websites.     - Make sure that everyone involved in producing the website is fully aware of what information is considered sensitive. Sometimes seemingly harmless information can be much more useful to an attacker than people realize. Highlighting these dangers can help make sure that sensitive information is handled more securely in general by your organization.   - Audit any code for potential information disclosure as part of your QA or build processes. It should be relatively easy to automate some of the associated tasks, such as stripping developer comments. | |
| **References**  https://portswigger.net/web-security/information-disclosure | |
| **ISM Follow-up**  Status: Open | |

# APPENDIX B – Determining the Severity of the Risk:

## 1. CVSS v3.1 Equations Base Score

The CVSS v3.1 equation Base Score is defined below.

|  |
| --- |
| The Base Score is a function of the Impact and Exploitability sub score equations. Where the Base score is defined as, If (Impact sub score = 0) 0 else,Scope Unchanged4 𝑅𝑜𝑢𝑛𝑑𝑢𝑝(𝑀𝑖𝑛𝑖𝑚𝑢𝑚[(𝐼𝑚𝑝𝑎𝑐𝑡 + 𝐸𝑥𝑝𝑙𝑜𝑖𝑡𝑎𝑏𝑖𝑙𝑖𝑡𝑦), 10]) Scope Changed 𝑅𝑜𝑢𝑛𝑑𝑢𝑝(𝑀𝑖𝑛𝑖𝑚𝑢𝑚[1.08 × (𝐼𝑚𝑝𝑎𝑐𝑡 + 𝐸𝑥𝑝𝑙𝑜𝑖𝑡𝑎𝑏𝑖𝑙𝑖𝑡𝑦), 10]) and the Impact sub score (ISC) is defined as, Scope Unchanged 6.42 × 𝐼𝑆𝐶Base Scope Changed 7.52 × [𝐼𝑆𝐶𝐵𝑎𝑠𝑒 - 0.029] - 3.25 × [𝐼𝑆𝐶𝐵𝑎𝑠𝑒 - 0.02]15 Where, 𝐼𝑆𝐶𝐵𝑎𝑠𝑒 = 1 - [(1 - 𝐼𝑚𝑝𝑎𝑐𝑡𝐶𝑜𝑛𝑓) × (1 - 𝐼𝑚𝑝𝑎𝑐𝑡𝐼𝑛𝑡𝑒𝑔) × (1 - 𝐼𝑚𝑝𝑎𝑐𝑡𝐴𝑣𝑎𝑖𝑙)] And the Exploitability sub score is, 8.22 × 𝐴𝑡𝑡𝑎𝑐𝑘𝑉𝑒𝑐𝑡𝑜𝑟 × 𝐴𝑡𝑡𝑎𝑐𝑘𝐶𝑜𝑚𝑝𝑙𝑒𝑥𝑖𝑡𝑦 × 𝑃𝑟𝑖𝑣𝑖𝑙𝑒𝑔𝑒𝑅𝑒𝑞𝑢𝑖𝑟𝑒𝑑 × 𝑈𝑠𝑒𝑟𝐼𝑛𝑡𝑒𝑟𝑎𝑐𝑡𝑖𝑜𝑛 |

## 2. Exploitability Metrics:

This metric reflects the context by which vulnerability exploitation is possible. This metric value (and consequently the Base score) will be larger the more remote (logically, and physically) an attacker can be in order to exploit the vulnerable component.

* **Attack Vector (AV):** Network (AV: N), Adjacent network (AV: A), Local (AV: L), Physical (AV: P)
* **Attack Complexity (AC):** Low (AC: L), High (AC: H)
* **Privileges Required (PR):** None (PR: N), Low (PR: L), High (PR: H)
* **User Interaction (UI):** None (UI: N), Required (UI: R)
* **Scope (S):** Unchanged (S: U), Changed (S: C)

## 3. Impact Metrics

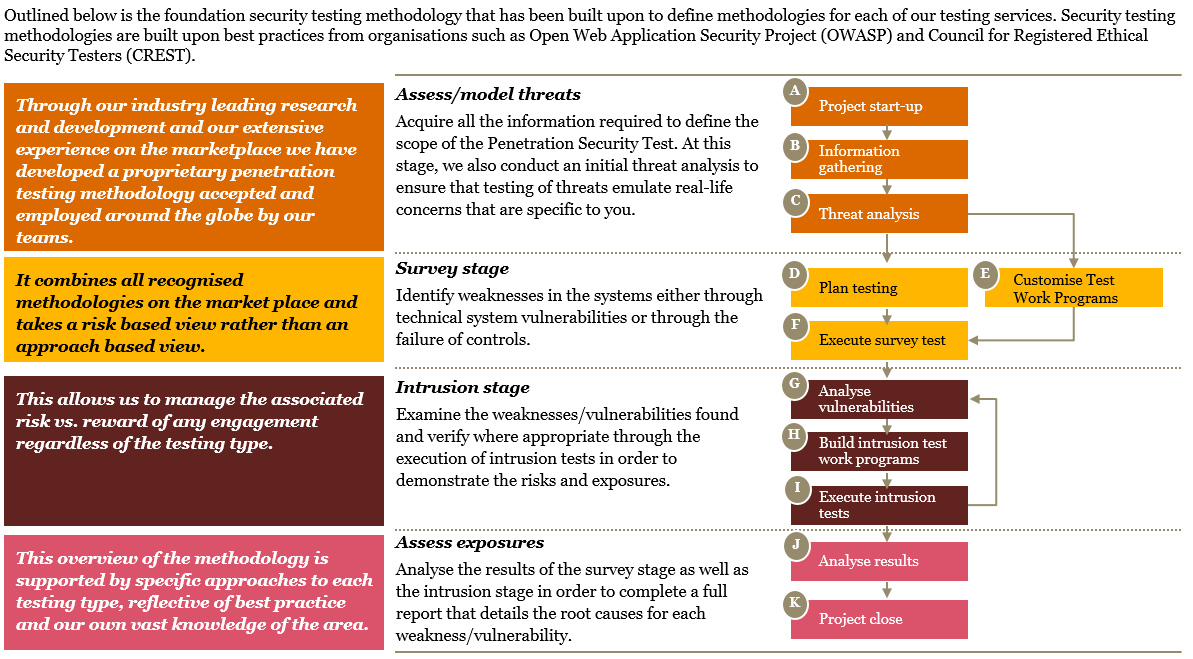
* Confidentiality Impact (C): None (C: N), Low (C: L), High (C: H)
* Integrity Impact (I): None (I: N), Low (I: L), High (I: H)
* Availability Impact (A): None (A: N), Low (A: L), High (A: H)

**4. Qualitative Severity Rating**

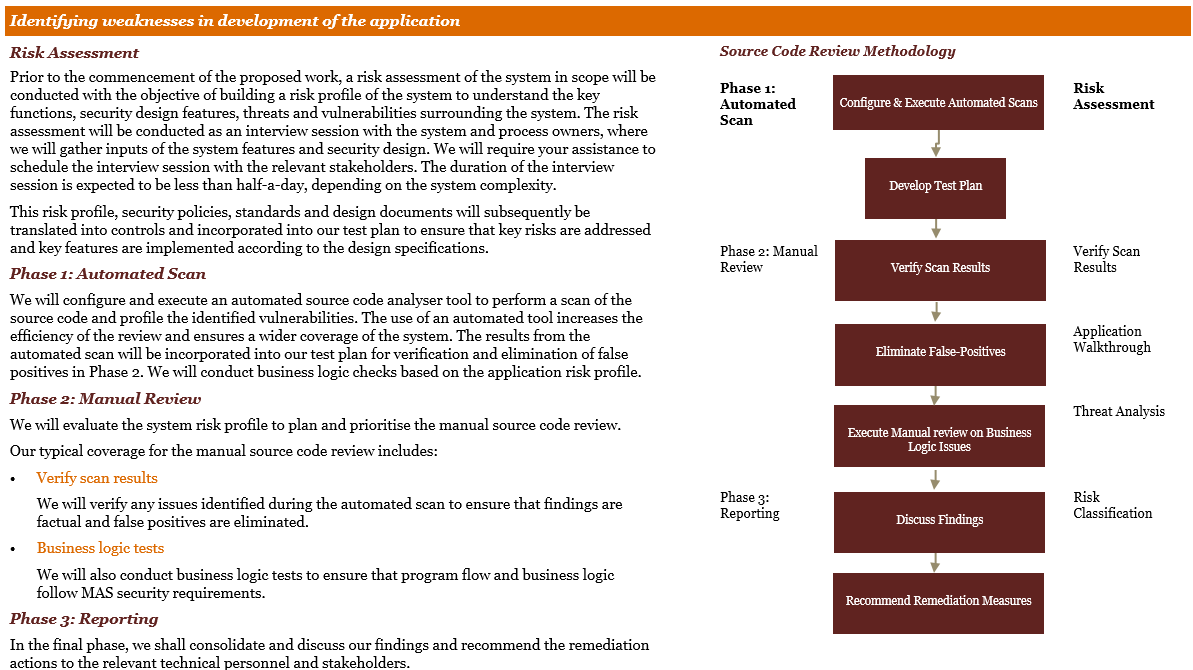
|  |  |
| --- | --- |
| **Rating** | **CVSS Score** |
| Low | 0.1 – 3.9 |
| Medium | 4.0 – 6.9 |
| High | 7.0 – 8.9 |
| Critical | 9.0 – 10 |

# APPENDIX C – OUR APPROACH AND METHODOLOGY

## 1. Our penetration testing methodology



## 2. Secure code review methodology overview



## 3. Penetration testing checklist

|  |  |
| --- | --- |
| **ID** | **Test Name** |
| **Information Gathering** | |
| WSTG-INFO-001 | Conduct Search Engine Discovery and Reconnaissance for Information Leakage |
| WSTG-INFO-002 | Fingerprint Web Server |
| WSTG-INFO-003 | Review Webserver Metafiles for Information Leakage |
| WSTG-INFO-004 | Enumerate Applications on Webserver |
| WSTG-INFO-005 | Review Webpage Comments and Metadata for Information Leakage |
| WSTG-INFO-006 | Identify application entry points |
| WSTG-INFO-007 | Map execution paths through application |
| WSTG-INFO-008 | Fingerprint Web Application Framework |
| WSTG-INFO-009 | Fingerprint Web Application |
| WSTG-INFO-010 | Map Application Architecture |
| **Configuration and Deploy Management Testing** | |
| WSTG-CONFIG-001 | Test Network/Infrastructure Configuration |
| WSTG-CONFIG-002 | Test Application Platform Configuration |
| WSTG-CONFIG-003 | Test File Extensions Handling for Sensitive Information |
| WSTG-CONFIG-004 | Backup and Unreferenced Files for Sensitive Information |
| WSTG-CONFIG-005 | Enumerate Infrastructure and Application Admin Interfaces |
| WSTG-CONFIG-006 | Test HTTP Methods |
| WSTG-CONFIG-007 | Test HTTP Strict Transport Security |
| WSTG-CONFIG-008 | Test RIA cross domain policy |
| WSTG-CONFIG-009 | Test file permission |
| WSTG-CONFIG-010 | Testing for subdomain Takeover |
| **Identity Management Testing** | |
| WSTG-IDENT-001 | Test Role Definitions |
| WSTG-IDENT-002 | Test User Registration Process |
| WSTG-IDENT-003 | Test Account Provisioning Process |
| WSTG-IDENT-004 | Testing for Account Enumeration and Guessable User Account |
| WSTG-IDENT-005 | Testing for Weak or unenforced username policy |
| WSTG-IDENT-006 | Test Permissions of Guest/Training Accounts |
| WSTG-IDENT-007 | Test Account Suspension/Resumption Process |
| **Identity Management Testing** | |
| WSTG-IDENT-001 | Test Role Definitions |
| WSTG-IDENT-002 | Test User Registration Process |
| WSTG-IDENT-003 | Test Account Provisioning Process |
| WSTG-IDENT-004 | Testing for Account Enumeration and Guessable User Account |
| WSTG-IDENT-005 | Testing for Weak or unenforced username policy |
| WSTG-IDENT-006 | Test Permissions of Guest/Training Accounts |
| WSTG-IDENT-007 | Test Account Suspension/Resumption Process |
| **Authentication Testing** | |
| WSTG-AUTHN-001 | Testing for Credentials Transported over an Encrypted Channel |
| WSTG-AUTHN-002 | Testing for default credentials |
| WSTG-AUTHN-003 | Testing for Weak lock out mechanism |
| WSTG-AUTHN-004 | Testing for bypassing authentication schema |
| WSTG-AUTHN-005 | Test remember password functionality |
| WSTG-AUTHN-006 | Testing for Browser cache weakness |
| WSTG-AUTHN-007 | Testing for Weak password policy |
| WSTG-AUTHN-008 | Testing for Weak security question/answer |
| WSTG-AUTHN-009 | Testing for weak password change or reset functionalities |
| WSTG-AUTHN-010 | Testing for Weaker authentication in alternative channel |
| **Authorization Testing** | |
| WSTG-AUTHZ-001 | Testing Directory traversal/file include |
| WSTG-AUTHZ-002 | Testing for bypassing authorization schema |
| WSTG-AUTHZ-003 | Testing for Privilege Escalation |
| WSTG-AUTHZ-004 | Testing for Insecure Direct Object References |
| **Session Management Testing** | |
| WSTG-SESS-001 | Testing for Bypassing Session Management Schema |
| WSTG-SESS-002 | Testing for Cookies attributes |
| WSTG-SESS-003 | Testing for Session Fixation |
| WSTG-SESS-004 | Testing for Exposed Session Variables |
| WSTG-SESS-005 | Testing for Cross Site Request Forgery |
| WSTG-SESS-006 | Testing for logout functionality |
| WSTG-SESS-007 | Test Session Timeout |
| WSTG-SESS-008 | Testing for Session puzzling |
| WSTG-SESS-009 | Testing for session Hijacking |
| **Data Validation Testing** | |
| WSTG-INPVAL-001 | Testing for Reflected Cross Site Scripting |
| WSTG-INPVAL-002 | Testing for Stored Cross Site Scripting |
| WSTG-INPVAL-003 | Testing for HTTP Verb Tampering |
| WSTG-INPVAL-004 | Testing for HTTP Parameter pollution |
| WSTG-INPVAL-005 | Testing for SQL Injection |
|  | Oracle Testing |
|  | MySQL Testing |
|  | SQL Server Testing |
|  | Testing PostgreSQL |
|  | MS Access Testing |
|  | Testing for NoSQL injection |
|  | Testing for ORM Injection |
|  | Testing for Client-side SQL injection |
| WSTG-INPVAL-006 | Testing for LDAP Injection |
| WSTG-INPVAL-007 | Testing for XML Injection |
| WSTG-INPVAL-008 | Testing for SSI Injection |
| WSTG-INPVAL-009 | Testing for XPath Injection |
| WSTG-INPVAL-010 | IMAP/SMTP Injection |
| WSTG-INPVAL-011 | Testing for Code Injection |
|  | Testing for Local File Inclusion |
|  | Testing for Remote File Inclusion |
| WSTG-INPVAL-012 | Testing for Command Injection |
| WSTG-INPVAL-013 | Testing for Format string Injection |
| WSTG-INPVAL-014 | Testing for incubated vulnerabilities |
| WSTG-INPVAL-015 | Testing for HTTP Splitting/Smuggling |
| WSTG-INPVAL-016 | Testing for HTTP incoming Requests |
| WSTG-INPVAL-017 | Testing for Host Header Injection |
| WSTG-INPVAL-018 | Testing for server-side template injection |
| WSTG-INPVAL-019 | Testing for server-side request forgery |
| **Error Handling** | |
| WSTG-ERR-001 | Testing for Improper Error Handling |
| WSTG-ERR-002 | Analysis of Stack Traces |
| **Cryptography** | |
| WSTG-CRYPST-001 | Testing for Weak SSL/TSL Ciphers, Insufficient Transport Layer Protection |
| WSTG-CRYPST-002 | Testing for Padding Oracle |
| WSTG-CRYPST-003 | Testing for Sensitive information sent via unencrypted channels |
| **Business logic Testing** | |
| WSTG-BUSLOGIC-001 | Test Business Logic Data Validation |
| WSTG-BUSLOGIC-002 | Test Ability to Forge Requests |
| WSTG-BUSLOGIC-003 | Test Integrity Checks |
| WSTG-BUSLOGIC-004 | Test for Process Timing |
| WSTG-BUSLOGIC-005 | Test Number of Times a Function Can be Used Limits |
| WSTG-BUSLOGIC-006 | Testing for the Circumvention of Work Flows |
| WSTG-BUSLOGIC-007 | Test Defenses Against Application Misuse |
| WSTG-BUSLOGIC-008 | Test Upload of Unexpected File Types |
| WSTG-BUSLOGIC-009 | Test Upload of Malicious Files |
| **Client Side Testing** | |
| WSTG-CLIENT-001 | Testing for DOM based Cross Site Scripting |
| WSTG-CLIENT-002 | Testing for JavaScript Execution |
| WSTG-CLIENT-003 | Testing for HTML Injection |
| WSTG-CLIENT-004 | Testing for Client Side URL Redirect |
| WSTG-CLIENT-005 | Testing for CSS Injection |
| WSTG-CLIENT-006 | Testing for Client Side Resource Manipulation |
| WSTG-CLIENT-007 | Test Cross Origin Resource Sharing |
| WSTG-CLIENT-008 | Testing for Cross Site Flashing |
| WSTG-CLIENT-009 | Testing for Clickjacking |
| WSTG-CLIENT-010 | Testing WebSockets |
| WSTG-CLIENT-011 | Test Web Messaging |
| WSTG-CLIENT-012 | Test Local Storage |
| WSTG-CLIENT-013 | Testing for cross Site script inclusion |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |