# COMPLEX APPLICATIONS SECURITY ASSESSMENT

For: Virtual Tech
By: Hacken
Date: October 9$^{th}$, 2023

# Table of Contents

HACKEN

This document contains confidential information about IT systems and the network infrastructure of the customer, as well as information about potential vulnerabilities and methods of their exploitation.
This confidential information is for internal use by the customer only and shall not be disclosed to third parties.

## Document

| Name: | COMPLEX SECURITY ASSESSMENT FOR VIRTUAL TECH |
|---|---|
| Type: | Detailed Penetration Test Report Remediation |
| Revision: | Version 3 |
| Date: | October 9th, 2023 |

## Contractor Contacts

| Role | Name | Email |
|---|---|---|
| Project Lead | Andrew Matiukhin | a.matiukhin@hacken.io |

# Introduction

We thank Virtual Tech for allowing us to conduct a Web Application Security Assessment. This document outlines our methodology, limitations, and results of the security assessment.

# Executive Summary

Hacken OÜ (Consultant) was contracted by Virtual Tech (Customer) to perform a third-party, independent security assessment of their web applications.

This report presents the findings of the security assessment of Web application & API security testing that was conducted between August 31$^{nd}$, 2023 – October 9$^{st}$, 2023.

The purpose of the engagement was to utilize active exploitation techniques to evaluate the security of the web application against best practices and to validate its security mechanisms.

Next vulnerabilities and mistakes were identified during the assessment:

|  | Web | Overall (after remediation check) | Unable to check |
|---|---|---|---|
| **Critical** | 0 | 0 | - |
| **High** | 2 | 0 | - |
| **Medium** | 1 | 0 | - |
| **Low** | 0 | 0 | - |
| **Informational** | 0 | 0 | - |

## Overall Security Benchmark (remediation)



Highly Insecure                                                      Highly Secure
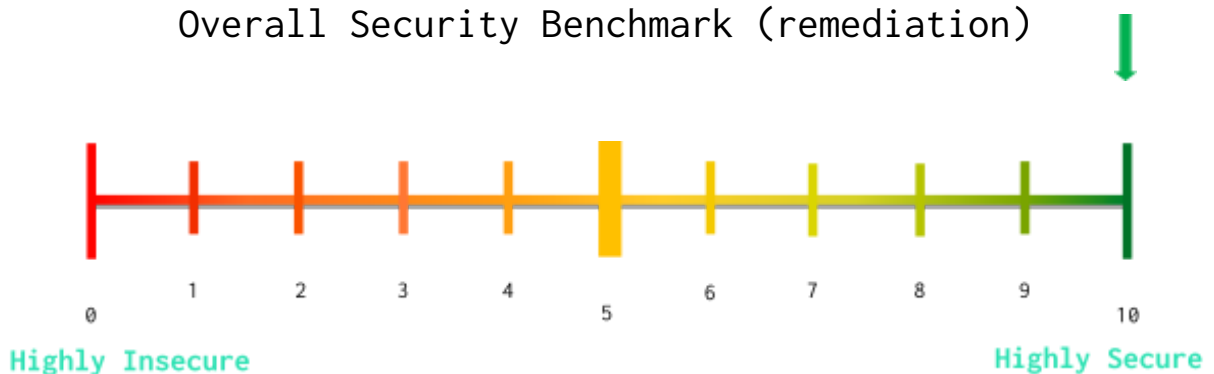
Based on our understanding of the environment, as well as the nature of the vulnerabilities discovered, their exploitability, and the potential impact we have

assessed the level of risk for your organization to be **Highly Secure**. No direct path of external attacker to full system compromise was discovered.

The overall rating of Customer Applications, after the security assessment by the Consultant's Security Team, stands out to be 10 out of 10. The security assessment was carried out following the in-house test cases, manual methods, exploitation, and automated tools.

## Security Assessment Overview

### Scope

The following table provides a synopsis of target systems that were within the scope of this Security Assessment.

| # | Name | Type |
|---|------|------|
| 1 | WEB Applications | https://vrcscan.com/ |
| 2 | API:<br>API Documentation: | |
| 3 | Additional subdomains detected: | dxb.vrcscan.com |

Security Assessment start and end dates were coordinated by email according to the following table:

| Testing start date: | August 31th, 2023 |
|---|---|
| Testing end date: | September 18th, 2023 |
| Remediation test date | October 9st, 2023 |
| Reporting: | October 9th, 2023 |

### Team Composition

The project team consisted of 3 security experts with the following roles, certifications, and responsibilities:

| Role | Responsibility |
|------|----------------|
| Project Manager | Customer communication<br>Project delivery and quality control |
| Penetration Tester #1<br>(Lead Penetration tester, OSCP, Node.js, React, PHP, Websockets) | Project planning and executing<br>Penetration Testing<br>Identify security and business risks for the application |

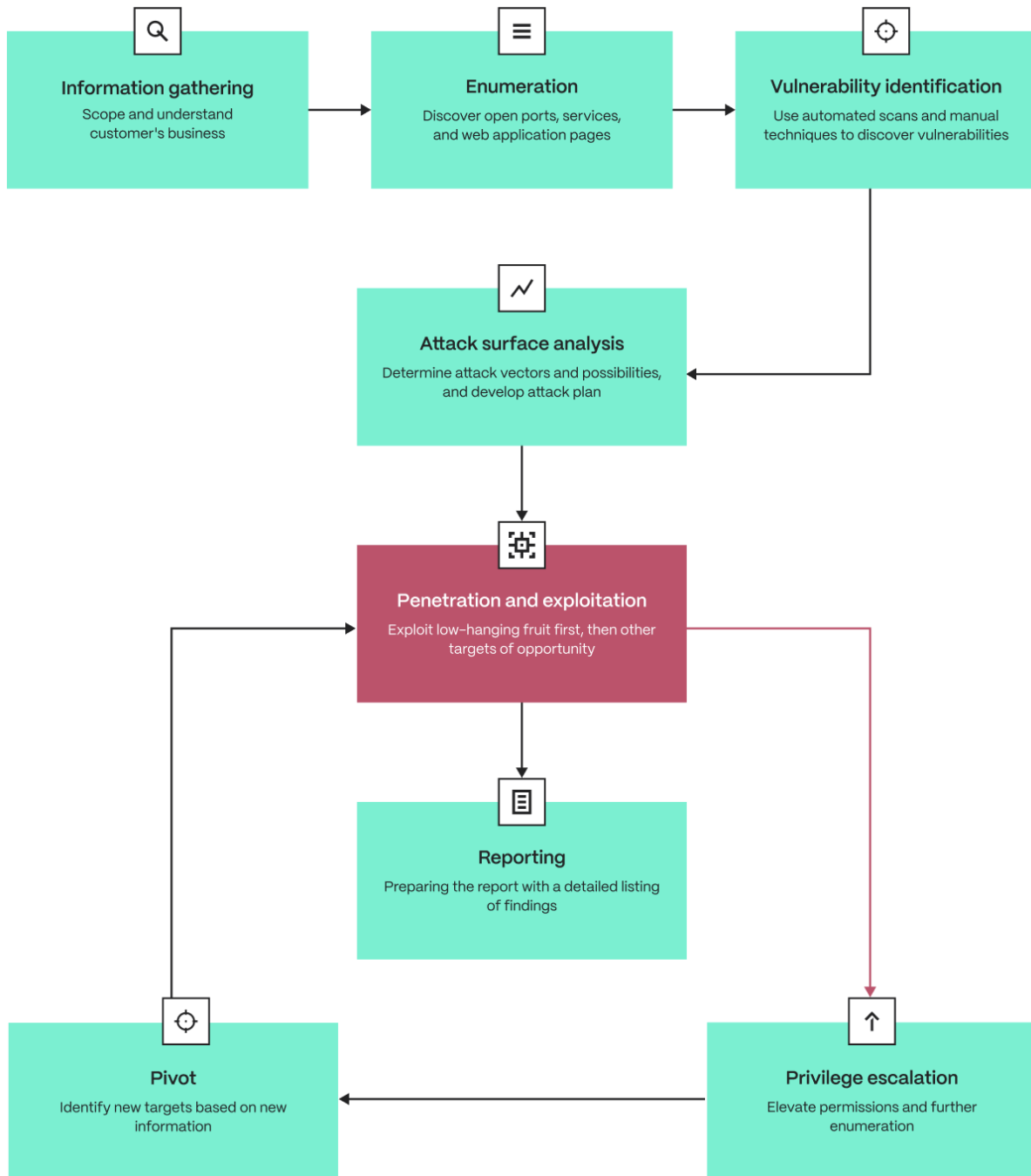| | Preparing artefacts and deliverables Results Presentation |
|---|---|
| Penetration Tester #2 (Penetration tester, Java, PHP, Node.js, Databases) | Penetration Testing Identify security and business risks for infrastructure |

# Methodology

Our methodology for Security Assessment is based on our own experience, best practices in the area of information security, international methodologies, and guides such as PTES and OWASP.

Security Assessment has been conducted following workflow:

- Pre-engagement Interactions
- Gray box security assessment
    - Intelligence gathering activities against a target
    - Service detection and identification
    - Vulnerabilities detection, verification, and analysis
    - Business logic flows
    - The exploitation of vulnerabilities
    - Lateral movement and privilege escalation
- Mapping application code against industry best practices OWASP ASVS
- Preparing the final report with a detailed listing of findings, along with the related risks and recommendations.

The diagram below illustrates the standard security assessment methodology followed by Hacken Team. A cyclical approach to security assessment is leveraged so new information is incorporated into subsequent on the environment.

**Information gathering**
Scope and understand customer's business

**Enumeration**
Discover open ports, services, and web application pages

**Vulnerability identification**
Use automated scans and manual techniques to discover vulnerabilities

**Attack surface analysis**
Determine attack vectors and possibilities, and develop attack plan

**Penetration and exploitation**
Exploit low-hanging fruit first, then other targets of opportunity

**Reporting**
Preparing the report with a detailed listing of findings

**Pivot**
Identify new targets based on new information

**Privilege escalation**
Elevate permissions and further enumeration

# Objectives

Web application security assessment was conducted in a "gray box" mode (with an approved account) and had the following objectives:
- Identify technical and functional vulnerabilities
- Estimate their severity level (ease of use, impact on information systems)

- Modeling the "most likely" attack vectors against the Customer's Information System
- Proof of concept and exploitation of vulnerabilities
- Draw up a prioritized list of recommendations to address identified weaknesses

## Limitations and Assumptions

This project is limited by the scope of this document
During this project, the Consultant will follow the following limitations:

- The operational impact to the networks will be maintained to the minimum and coordinated with the client
- No denial of service attacks will be used
- No active backdoor or Trojans will be installed
- No client data will be copied, modified, or destroyed

The following security tests shall be considered Out of Scope for this assessment:

- Internal networks assessment
- Denial of Service testing
- Physical Social Engineering testing

## Disclaimer

This security assessment was conducted for the Customer prod environment and is valid on the date of the report submission hereto. The description of findings, recommendations, and risks was valid on the date of submission of the report hereto. Any projection to the future of the report's information is subject to risk due to changes in the Infrastructure architecture, and it may no longer reflect its logic and controls.

## Definitions & Abbreviations

The severity level (criticality level) of each vulnerability is determined based on the exploitation difficulty and the access level to an enterprise or end-user information system an attacker can gain in case of successful exploitation. The lower the exploitation difficulty level and the higher the access level which an attacker can get, the higher the vulnerability severity level will be. The matrix below illustrates the general methodology followed for identifying the severity level of each finding:
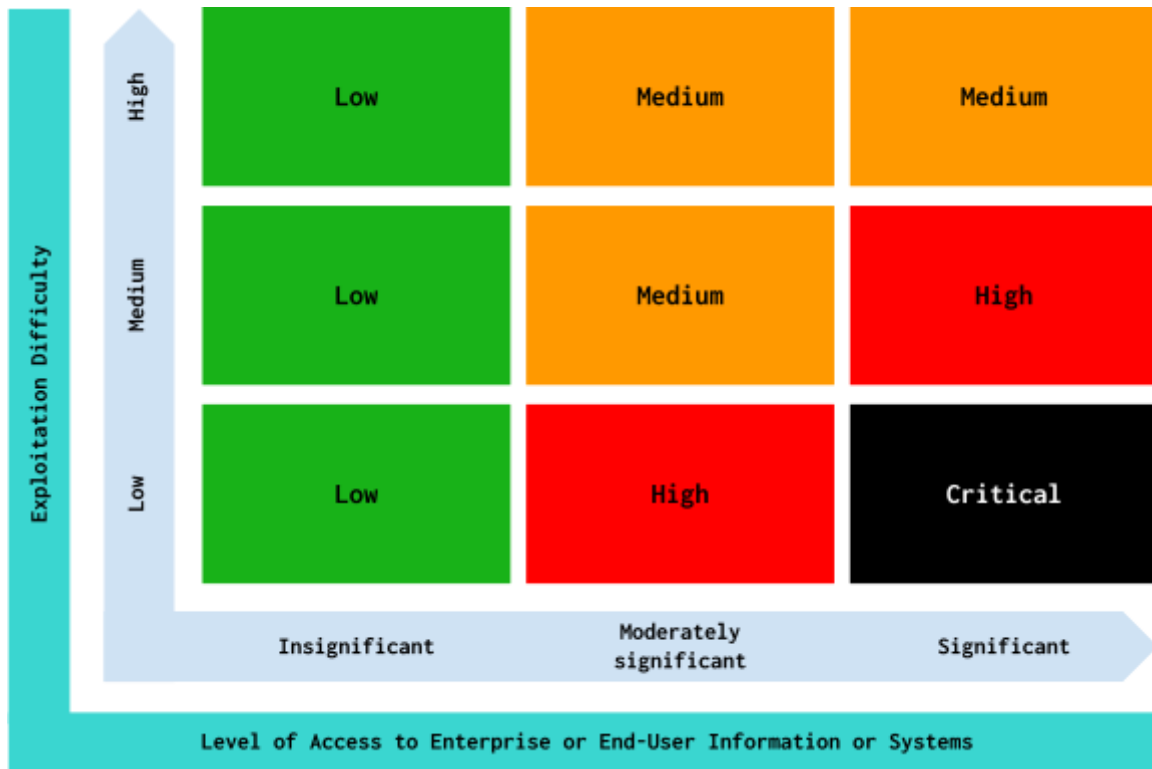
To be the CVSS Compliant Hacken Security Assessment Team utilizes the Common Vulnerability Scoring System (CVSS) values for each level of vulnerability, such a solution will help prioritize the fixing vulnerabilities approach and make the results of the Security Assessment more objective.

The table below fully describes each level of vulnerabilities and ties to CVSS:

| Severity | Color Map | Description |
|---|---|---|
| Informational | ▪ | This level refers to vulnerabilities that do not pose an immediate security risk or require exploitation. Instead, they provide valuable information or insights about the system's configuration, weaknesses, or potential areas for improvement. While they may not directly lead to a security breach, addressing these informational vulnerabilities can |

| Severity | Color Map | Description |
|---|---|---|
|  |  | contribute to overall security enhancements and proactive risk mitigation efforts. |
| Low | CVSS (0.1 - 3.9) | This level encompasses vulnerabilities with a low exploitation difficulty and low level of access. These vulnerabilities pose a relatively lower risk to the system's security as they are easier to exploit and grant minimal access privileges to potential attackers. While they still require attention and remediation, their impact is limited due to the restricted level of access gained. |
| Medium | CVSS (4.0 - 6.9) | Vulnerabilities falling under this level have a moderate exploitation difficulty but the access level which can be gained by the attacker is greater compared to low-level vulnerabilities. They represent a medium level of risk to the system's security. Although they may be more challenging to exploit compared to low-level vulnerabilities, they still do not pose an immediate and severe threat. Appropriate measures should be taken to address these vulnerabilities promptly to prevent potential exploitation. |

| Severity | Color Map | Description |
|---|---|---|
| High | ■■■■<br>CVSS (7.0 - 8.9) | Vulnerabilities categorized as high-level vulnerabilities have a low exploitation difficulty and grant a higher access level for the attacker in case of successful exploitation. These vulnerabilities pose a significant risk to the system's security and require immediate attention. While they may be more challenging to exploit, their potential impact is substantial. Timely remediation and mitigation measures should be implemented to address these vulnerabilities effectively. |
| Critical | ■■■■■<br>CVSS (9.0 - 10.0) | This level encompasses vulnerabilities with a low exploitation difficulty but the highest access level granted to the attacker in case of successful exploitation. These vulnerabilities are considered critical and pose the most severe threat to the system's security. Immediate action should be taken to remediate and address these vulnerabilities to prevent potential unauthorized access and significant security breaches. |

# Summary of Findings

| Value | Number of issues (after remediation check) |
|---|---|
| Informational | 0 |
| Low | 0 |
| Medium | 0 |
| High | 0 |
| Critical | 0 |
| Unable to check | 0 |

# Appendix A. OWASP Testing Checklist

| Category | Test Name | Result | Details |
|---|---|---|---|
| Information Gathering | | | |
| OTG-INFO-001 | Conduct Search Engine Discovery and Reconnaissance for Information Leakage | Done | No vulnerability detected |
| OTG-INFO-002 | Fingerprint Web Server | Done | Done with whatweb and nmap |
| OTG-INFO-003 | Review Webserver Metafiles for Information Leakage | Done | Manual testing |
| OTG-INFO-004 | Enumerate Applications on Webserver | Done | Done with whatweb and nmap |
| OTG-INFO-005 | Review Webpage Comments and Metadata for Information Leakage | Done | Done with dirbuster |
| OTG-INFO-006 | Identify application entry points | Done | No vulnerability detected |
| OTG-INFO-007 | Map execution paths through application | Done | Done with Burp Suite |
| OTG-INFO-008 | Fingerprint Web Application Framework | Done | Done with whatweb and nmap |
| OTG-INFO-009 | Fingerprint Web Application | Done | Done with whatweb and nmap |
| OTG-INFO-010 | Map Application Architecture (WAF, Application server, identify application architecture) | Tested | No vulnerability detected |
| Configuration and Deploy Management Testing | | | |
| OTG-CONFIG-001 | Test Network/Infrastructure Configuration | Tested | No vulnerability detected |
| OTG-CONFIG-002 | Test Application Platform Configuration | Tested | No vulnerability detected |
| OTG-CONFIG-003 | Test File Extensions Handling for Sensitive Information | Tested | No vulnerability detected |
| OTG-CONFIG-004 | Backup and Unreferenced Files for Sensitive Information | Tested | No vulnerability detected |
| OTG-CONFIG-005 | Enumerate Infrastructure and Application Admin Interfaces | Tested | No vulnerability detected |
| OTG-CONFIG-006 | Test HTTP Methods | Tested | No vulnerability detected |
| OTG-CONFIG-007 | Test HTTP Strict Transport Security | Tested | No vulnerability detected |
| OTG-CONFIG-008 | Test RIA cross domain policy | Tested | No vulnerability detected |
| Identity Management Testing | | | |

| OTG-SESS-002 | Testing for Cookies attributes | Tested | No vulnerability detected |
|---|---|---|---|
| OTG-SESS-003 | Testing for Session Fixation | Tested | No vulnerability detected |
| OTG-SESS-004 | Testing for Exposed Session Variables | Tested | No vulnerability detected |
| OTG-SESS-005 | Testing for Cross Site Request Forgery | Tested | No vulnerability detected |
| OTG-SESS-006 | Testing for logout functionality | Tested | No vulnerability detected |
| OTG-SESS-007 | Test Session Timeout | Tested | No vulnerability detected |
| OTG-SESS-008 | Testing for Session puzzling | Tested | No vulnerability detected |
| Data Validation Testing | | | |
| OTG-INPVAL-001 | Testing for Reflected Cross Site Scripting | Tested | No vulnerability detected |
| OTG-INPVAL-002 | Testing for Stored Cross Site Scripting | Tested | No vulnerability detected |
| OTG-INPVAL-003 | Testing for HTTP Verb Tampering | Tested | No vulnerability detected |
| OTG-INPVAL-004 | Testing for HTTP Parameter pollution | Tested | No vulnerability detected |
| OTG-INPVAL-005 | Testing for SQL Injection | Tested | No vulnerability detected |
| OTG-INPVAL-006 | Testing for LDAP Injection | Tested | No vulnerability detected |
| OTG-INPVAL-007 | Testing for ORM Injection | Tested | No vulnerability detected |
| OTG-INPVAL-008 | Testing for XML Injection | Tested | No vulnerability detected |
| OTG-INPVAL-009 | Testing for SSI Injection | Tested | No vulnerability detected |
| OTG-INPVAL-010 | Testing for XPath Injection | Tested | No vulnerability detected |
| OTG-INPVAL-011 | IMAP/SMTP Injection | Tested | No vulnerability detected |
| OTG-INPVAL-012 | Testing for Code Injection | Tested | No vulnerability detected |
| OTG-INPVAL-013 | Testing for Command Injection | Tested | No vulnerability detected |
| OTG-INPVAL-014 | Testing for Buffer overflow | Tested | No vulnerability detected |
| OTG-INPVAL-015 | Testing for incubated vulnerabilities | Tested | No vulnerability detected |
| OTG-INPVAL-016 | Testing for HTTP Splitting/Smuggling | Tested | No vulnerability detected |
| Error Handling | | | |
| OTG-ERR-001 | Analysis of Error Codes | Tested | No vulnerability detected |

| OTG-IDENT-001 | Test Role Definitions | Tested | No vulnerability detected |
|---|---|---|---|
| OTG-IDENT-002 | Test User Registration Process | Tested | No vulnerability detected |
| OTG-IDENT-003 | Test Account Provisioning Process | Tested | No vulnerability detected |
| OTG-IDENT-004 | Testing for Account Enumeration and Guessable User Account | Tested | No vulnerability detected |
| OTG-IDENT-005 | Testing for Weak or unenforced username policy | Tested | No vulnerability detected |
| OTG-IDENT-006 | Test Permissions of Guest/Training Accounts | Tested | No vulnerability detected |
| OTG-IDENT-007 | Test Account Suspension/Resumption Process | Tested | No vulnerability detected |
| Authentication Testing | | | |
| OTG-AUTHN-001 | Testing for Credentials Transported over an Encrypted Channel | Tested | No vulnerability detected |
| OTG-AUTHN-002 | Testing for default credentials | Tested | No vulnerability detected |
| OTG-AUTHN-003 | Testing for Weak lock out mechanism | Tested | No vulnerability detected |
| OTG-AUTHN-004 | Testing for bypassing authentication schema | Tested | No vulnerability detected |
| OTG-AUTHN-005 | Test remember password functionality | Tested | No vulnerability detected |
| OTG-AUTHN-006 | Testing for Browser cache weakness | Tested | No vulnerability detected |
| OTG-AUTHN-007 | Testing for Weak password policy | Tested | No vulnerability detected |
| OTG-AUTHN-008 | Testing for Weak security question/answer | Tested | No vulnerability detected |
| OTG-AUTHN-009 | Testing for weak password change or reset functionalities | Tested | No vulnerability detected |
| OTG-AUTHN-010 | Testing for Weaker authentication in alternative channel | Tested | No vulnerability detected |
| Authorization Testing | | | |
| OTG-AUTHZ-001 | Testing Directory traversal/file include | Tested | No vulnerability detected |
| OTG-AUTHZ-002 | Testing for bypassing authorization schema | Tested | No vulnerability detected |
| OTG-AUTHZ-003 | Testing for Privilege Escalation | Tested | No vulnerability detected |
| OTG-AUTHZ-004 | Testing for Insecure Direct Object References | Tested | No vulnerability detected |
| Session Management Testing | | | |
| OTG-SESS-001 | Testing for Bypassing Session Management Schema | Tested | No vulnerability detected |

| OTG-ERR-002 | Analysis of Stack Traces | Tested | No vulnerability detected |
| --- | --- | --- | --- |
| Cryptography | | | |
| OTG-CRYPST-001 | Testing for Weak SSL/TSL Ciphers, Insufficient Transport Layer Protection | Tested | No vulnerability detected |
| OTG-CRYPST-002 | Testing for Padding Oracle | Tested | No vulnerability detected |
| OTG-CRYPST-003 | Testing for Sensitive information sent via unencrypted channels | Tested | No vulnerability detected |
| Client Side Testing | | | |
| OTG-CLIENT-001 | Testing for DOM based Cross Site Scripting | Tested | No vulnerability detected |
| OTG-CLIENT-002 | Testing for JavaScript Execution | Tested | No vulnerability detected |
| OTG-CLIENT-003 | Testing for HTML Injection | Tested | No vulnerability detected |
| OTG-CLIENT-004 | Testing for Client Side URL Redirect | Tested | No vulnerability detected |
| OTG-CLIENT-005 | Testing for CSS Injection | Tested | No vulnerability detected |
| OTG-CLIENT-006 | Testing for Client Side Resource Manipulation | Tested | No vulnerability detected |
| OTG-CLIENT-007 | Test Cross Origin Resource Sharing | Tested | No vulnerability detected |
| OTG-CLIENT-008 | Testing for Cross Site Flashing | Tested | No vulnerability detected |
| OTG-CLIENT-011 | Test Web Messaging | Tested | No vulnerability detected |
| OTG-CLIENT-012 | Test Local Storage | Tested | No sensitive data stored in Local or Session storage detected |
| Business Logic Testing | | | |
| OTG-BUSLOGIC-001 | Test Business Logic Data Validation | Tested | No vulnerability detected |
| OTG-BUSLOGIC-002 | Test Ability to Forge Request | Tested | No vulnerability detected |
| OTG-BUSLOGIC-003 | Test Integrity Checks | Tested | No vulnerability detected |
| OTG-BUSLOGIC-004 | Test for Process Timing | Tested | No vulnerability detected |
| OTG-BUSLOGIC-005 | Test Numbers of Times a Function Can be Used Limits | Tested | No vulnerability detected |
| OTG-BUSLOGIC-006 | Test for the Circumvention of Work Flows | Tested | No vulnerability detected |
| OTG-BUSLOGIC-007 | Test Upload of Unexpected File Types | Tested | No vulnerability detected |
| OTG-BUSLOGIC-008 | Test Upload of Malicious Files | Tested | No vulnerability detected |