Tester: {{Author\_1}}, OSCP, CISSP

{{Customer\_Name}} - Web Application Penetration Test

**Table of Contents**

[Introduction 2](#_Toc187143972)

[Executive Summary 2](#_Toc187143973)

[Scoping 3](#_Toc187143974)

[Service Description 3](#_Toc187143975)

[Goals and Objectives 5](#_Toc187143976)

[Process and Methodology 5](#_Toc187143977)

[Security Findings Overview 7](#_Toc187143978)

[Vulnerability Details Table 8](#_Toc187143979)

[Appendix: Tools Used 11](#_Toc187143980)

[Appendix: Vulnerability Exploit Details 12](#_Toc187143981)

## Introduction

The {{Vendor\_Name}} Penetration Testing team specializes in penetration testing techniques that help identify areas for improvement across in-scope applications. The Penetration Testing team focuses on manual assessments that go beyond basic automated tests to identify real attack vectors that can be used against your application(s).

## Executive Summary

This report presents the results of the “Black Box” penetration testing for the {{Customer\_Name}} web application. The recommendations provided in this report are structured to provide details of key vulnerabilities and facilitate remediation of the identified security risks. This document serves as a formal attestation for the web application penetration testing engagement.

Evaluation ratings compare information gathered during the engagement to “best in class” criteria for security standards. The statements made in this document provide an accurate assessment of the {{Customer\_Name}}’s current security as it relates to the {{Customer\_Name}}’s in-scope application(s).

Overall Security Posture Rating:

After conducting comprehensive testing, {{Vendor\_Name}} determined that the overall security posture of the evaluated sites and applications is robust. The findings indicate a well-secured environment with strong defenses in place to mitigate potential threats. Lower risk findings that include web server configuration weaknesses, TLS weaknesses, information disclosure, and outdated software are some of the most common issues exposed publicly. These types of findings are common for similar types of organizations.

However, a small number of web host configuration weaknesses were identified during the assessment. While these issues do not significantly undermine the overall security, addressing them would further enhance the resilience of the tested systems against potential attacks.

This is an area where more details can be provided to the customer about enterprise-wide foundational issues that may assist in preventing the types of vulnerabilities that were discovered. Take your time to outline all issues tactically but support the findings with what they may mean strategically across all enterprise applications and assets.

The overall Security Posture grade for the assessed environment:

|  |  |  |
| --- | --- | --- |
| **Scope** | **Grade** | **Security Level** |
| {{Application\_Name}} Web Application | {{Security\_Score}} | Good |

Security Posture Framework:

A structured framework is used for evaluating the overall security posture of the tested environment, categorizing it into four grades: **A (Excellent)**, **B (Good)**, **C (Fair)**, and **D (Poor)**.

Each grade reflects how well the organization aligns with "Industry Best Practice" security standards, based on findings from each security assessment.

|  |  |  |
| --- | --- | --- |
| **Grade** | **Security Level** | **Description** |
| **A** | Excellent | The security exceeds “Industry Best Practice” standards. The overall posture was found to be excellent with only a few low-risk findings identified. |
| **B** | Good | The security meets accepted standards for “Industry Best Practice.” The overall posture was found to be strong with only a handful of medium- and low-risk shortcomings identified. |
| **C** | Fair | Current solutions protect some areas of the enterprise from security issues. Moderate changes are required to elevate the discussed areas to “Industry Best Practice” standards |
| **D** | Poor | Significant security deficiencies exist. Immediate attention should be given to the discussed issues to address exposures identified. Major changes are required to elevate to “Industry Best Practice” standards. |

## Scoping

{{Vendor\_Name}} was engaged to perform {{service\_scope}} penetration test against the in-scope application(s). The purpose of the engagement was to identify and prioritize any potential areas of security vulnerability across the following in-scope assets(s):

**Hostnames and/or IP Addresses**

{{Hostname}}

{{IP\_Address}}

**User Accounts**

{{UserAcct1}}

{{UserAcct2}}

The engagement was performed during {{Start\_Date}} and {{End\_Date}} and included TOTAL\_DAYS days of testing/analysis and 2 days of documentation. This was a time-boxed, point-in-time assessment.

## Service Description

Web Application Penetration Testing involves simulating real-world attacks using the same techniques employed by malicious actors. Unlike basic vulnerability scans, this in-depth assessment requires skilled professionals and sufficient time to thoroughly evaluate all exposed functionality and content.

Our approach combines automated tools and manual techniques to ensure comprehensive coverage. While initial vulnerability scans were conducted using commercial web application scanners, the bulk of the testing was performed through manual analysis and request manipulation, utilizing a variety of commercial and open-source tools.

The Web Application Testing service plays a vital role in safeguarding customer application services against a wide range of security threats.

**Addressing the Complexity of Application Security**

Application security vulnerabilities are among the most prevalent and increasingly complex challenges facing organizations today. While we use the OWASP Top Ten vulnerabilities as a baseline for identifying security flaws, our testing extends beyond these categories. Our experts identify both well-documented and novel vulnerabilities, offering insights into how unique application configurations can be exploited. Along with our findings, we provide actionable recommendations for remediation to strengthen your application's security posture.

Vulnerability Definitions and Ratings

The vulnerabilities detailed in this report are assessed using industry-standard methodologies and {{Vendor\_Name}} ratings. These ratings are determined through a comprehensive evaluation of security factors, including:

* Ease of Exploit (Sophisticated, Moderate, Trivial): This metric assesses how easily an attacker could leverage a vulnerability to compromise the system. It considers factors such as the level of access required (e.g., local, remote), the technical complexity of the exploit, and the availability of relevant information, whether through public sources or insider knowledge.
* Vulnerability Impact (Critical, High, Medium, Low): This rating evaluates the potential consequences of a successful exploit. It is informed by the nature of the vulnerability, the type of exploit observed, and the potential for cascading effects on other critical systems or assets within the network.
* Remediation Effort (High, Medium, Low): This metric measures the resources required to address the identified vulnerability effectively. It considers the time, effort, technical expertise, and testing needed to implement a robust and reliable resolution.

These ratings provide a structured and actionable framework for prioritizing vulnerabilities and addressing them with appropriate measures. An attempt is made to determine the *overall risk* of a vulnerability using the following Severity Scoring table:

|  |  |  |  |
| --- | --- | --- | --- |
| Rating | Description | Likelihood | Impact |
| Critical | Exploitation of the vulnerability would result in a complete compromise of the system, network, or application. Immediate remediation is required. | Exploitation is almost certain due to ease of exploitation, publicly available tools, or automated attack methods. | Severe impact: Full data loss, system control, regulatory penalties, or significant business disruption. |
| High | Exploitation would lead to major exposure or compromise, significantly affecting operations, confidentiality, or integrity. Prompt remediation is necessary | Exploitation is likely because of feasible attack methods, a known exploit, or widespread vulnerability awareness. | Significant impact: Potential data breaches, unauthorized access, and disruption of key services. |
| Medium | Exploitation poses a moderate risk to systems, with limited or localized impact on operations, confidentiality, or integrity. Remediation should be prioritized accordingly. | Exploitation is possible but requires specific conditions, user interaction, or uncommon attack vectors. | Moderate impact: Unauthorized access to non-critical data, localized service interruptions, or partial exposure of information. |
| Low | Exploitation is unlikely to have a substantial impact on operations or confidentiality, representing only minor weaknesses. Address during regular maintenance cycles. | Exploitation is unlikely due to limited attack feasibility or mitigations already in place. | Minimal impact: Negligible effect on business operations or confidentiality. |
| Informational | The issue does not represent a security vulnerability but provides details that could enhance understanding of the system or assist in future assessments. | Exploitation is not applicable, as this is purely informational and poses no direct risk. | No impact: The finding is purely informational and does not affect security, operations, or data integrity. |

## Goals and Objectives

The objective of this assessment was to evaluate the security posture of the in-scope devices, servers, and applications through a structured and comprehensive penetration testing process. This approach included phases such as information gathering, fingerprinting, foot printing, threat modeling, service enumeration, and analysis. Automated vulnerability scanning tools were used to identify potential weaknesses, supplemented by a detailed manual analysis performed by experienced penetration testers to uncover additional risks and validate the findings within the in-scope applications.

## Process and Methodology

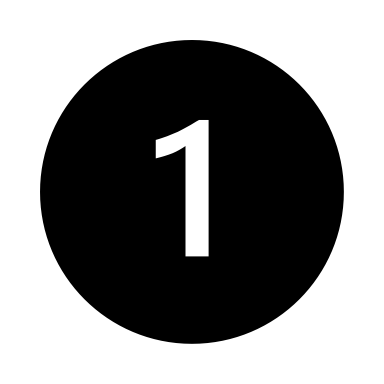
This test was designed to provide insight into methods of attack against specific infrastructure components, products, or suite of resident applications and present a reasonable example of what an attacker might accomplish. This assessment was not intended to provide a comprehensive security evaluation of the architecture, organization, or department; rather, it concentrated on modeling specific attack scenarios, identifying vulnerabilities, and validating exploitation possibilities.

Our application penetration testing methodology is based on the following industry standard:

* [**Open Web Application Security Project (OWASP) Testing Guide**](https://owasp.org/www-project-web-security-testing-guide/stable/)
* [**Penetration Testing Execution Standard (PTES)**](http://www.pentest-standard.org/index.php/Main_Page)

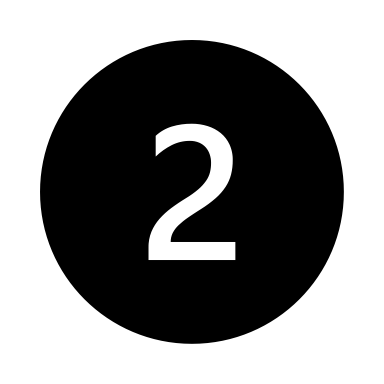
The **Open Web Application Security Project (OWASP) Testing Guide** is a comprehensive framework for assessing the security of web applications. It serves as a roadmap for testers, developers, and security practitioners, offering practical methods to identify and mitigate security vulnerabilities in web applications.

The **Penetration Testing Execution Standard (PTES)** is a comprehensive framework that defines best practices and a structured approach for conducting penetration tests. It covers every phase of a testing engagement, including pre-engagement interactions, intelligence gathering, threat modeling, vulnerability analysis, exploitation, post-exploitation, and reporting. PTES emphasizes clear communication with stakeholders, understanding the organization's business context, and tailoring the assessment to align with its unique risks and objectives. It incorporates technical and procedural guidance to ensure thoroughness, repeatability, and actionable results, helping organizations identify and mitigate vulnerabilities while improving their overall security posture.

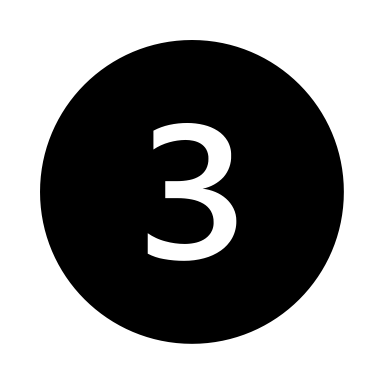
****Our testing methodology can be summarized in the following key phases:

**Reconnaissance**

The primary goal in this step of the process is to discover crucial deployment details, authentication details, authorization details, data and functionality available through the tested application(s), all of which provide the foundation for a tailored penetration test. This helps us develop a picture of the host, application, and related services. Reconnaissance is carried out via network exposed information, automated scans or scripts, application fingerprinting and discovery, and stepping through the application while recording the application map.

**Automated Testing**

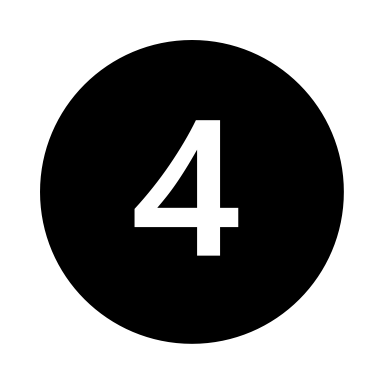
Vulnerability scanners and scripts provide a foundation for the full manual assessment, and each finding is manually verified to ensure accuracy and remove false positives.

**Exploration, Evaluation and Verification**

Testers use the results of automated scans paired with their knowledge and experience, to conduct a manual security analysis of the in-scope applications.

The process of application assessments is different because it challenges the penetration tester to assess the application and its functionality in unintended ways. The exploration techniques used while testing web application security include the following:

* Information Gathering
* Configuration and Deployment Management Testing
* Identity Management Testing
* Authentication Testing
* Authorization Testing
* Session Management Testing
* Input Validation Testing
* Testing for Error Handling
* Testing for Weak Cryptography
* Business Logic Testing
* Client-side Testing

**Assessment Reporting**

Once the engagement is complete, testers deliver a detailed analysis and threat report, including remediation steps. Our analysts deliver industry standard, clear and concise reports, prioritizing the highest risk vulnerabilities first.

## Security Findings Overview

Testing revealed that the security posture of the tested sites and applications is generally strong, with only a limited number of web host configuration weaknesses identified.

These findings included:

* Insert bullet list of selected vulnerabilities

While the immediate risk of exploitation is mitigated by the application’s small user base, this risk will increase as the number of users and roles grows. Exploitation of these vulnerabilities requires authenticated access, but once such access is achieved, an attacker could:

* Insert bullet list of each impact from each selected vulnerability

Addressing these vulnerabilities proactively will help ensure the application remains secure as its user base expands.

Penetration testers performed manual security testing according to the OWASP Web Application Testing Methodology to verify and validate each of the following security issues:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Severity** | **Critical** | **High** | **Medium** | **Low** | **Informational** |
|  |  |  |  |  |  |
| Number of Issues Identified | 10 | 7 | 5 | 3 | 1 |

**Vulnerability Security Impact Matrix and Heatmap**

The vulnerability heatmap included in this report provides a visual representation of identified vulnerabilities, mapped based on their likelihood of exploit, ease of exploit, impact severity, and the amount of time and resources required for resolution. By plotting these factors, the heatmap allows stakeholders to quickly assess and prioritize vulnerabilities that pose the most significant risk to the organization. This method ensures that attention is directed to vulnerabilities that are not only easily exploitable but also have a high potential impact, enabling efficient allocation of remediation efforts. Additionally, this approach supports strategic planning by highlighting vulnerabilities that require substantial resources to address, helping teams to balance risk mitigation with operational constraints. Through this visual tool, decision-makers gain a comprehensive perspective, facilitating informed choices and fostering a proactive security posture.

{{heatmap}}

## Vulnerability Details Table

The *Vulnerability Details* table describes potential vulnerabilities, the likelihood or difficulty of exploit, the overall business impact, and the {{Vendor\_Name}}’s recommendations.

Vulnerabilities are arranged in order of impact, with the highest impact issues appearing first. These vulnerabilities were identified and verified during this Web Application Penetration Test.

| Vulnerability Description | Ease of Exploit | Impact | Recommendation |
| --- | --- | --- | --- |
| **{{Vulnerability\_name}}**  {{Vulnerability\_Description}} | **{{Ease\_of\_exploit}}.** {{Ease\_of\_exploit\_description}} | **{{Impact}}.** {{Impact\_Description}} | {{Vulnerability\_solution}} |

## Appendix: Tools Used

Below is a list of the tools used by testers to help identify and validate findings during the testing phases of this engagement:

* Manual testing: Burp Suite Pro (Commercial Edition), Postman, Curl
* Vulnerability scan: Nessus, Nikto, Burp
* Network scan: Nmap, Masscan, Whatweb, Dig
* Directory enumeration: gobuster, dirb
* Injection testing tools: XSSHunter, SQLmap, Burp
* Encryption: TestSSL, SSLscan
* Custom Scripting: Bash, Python

## Appendix: Vulnerability Exploit Details

Below are further details to clarify how the identified issues were identified and may assist when retesting once fixes are put in place.

1. **Unrestricted File Upload**: Users can upload arbitrary files to the web server. It appears that any type of file is accepted by the application. Note that a .php file was successfully uploaded, however the only option was to download the file. So, executing malicious files on the server was not achievable.