Author

CAST

******

**OWASP API 2023 TOP 10**

**Summary Report**

Application Name –

Version –

CAST AIP -

|  |
| --- |
|  |
|  |

Monday, xx July 2012

My Application Name

Version Number

My CAST Version

# Table of Content

Table of Content

1. Introduction

1.1. Application Characteristics

2. Security Violation Overview

2.1. OWASP API -2023 Top 10 Vulnerabilities

2.2. OWASP - API1:2023 Broken Object Level Authorization

2.3. OWASP - API2:2023 Broken User Authentication

2.4. OWASP - API3:2023 Excessive Data Exposure

2.5. OWASP - API4:2023 Lack of Resources & Rate Limiting

2.6. OWASP - API5:2023 Broken Function Level Authorization

2.7. OWASP - API6:2023 Mass Assignment

2.8. OWASP - API7:2023 Security Misconfiguration

2.9. OWASP - API8:2023 Injection

2.10. OWASP - API9:2023 Improper Assets Management

2.11. OWASP - API10:2023 Insufficient Logging & Monitoring

3. Appendix

3.1. About CAST Software Intelligence

3.2. About CAST Security

3.3. Applicability of OWASP API 2023 in CAST Solution

# Introduction

This assessment is an effort to determine the security health of the application and identify some of the root causes of current Security concerns, as well as any risks of future degradation. This assessment uses CAST GateKeeper to automatically scan the implementation of these applications to review the architecture, design, and code against OWASP standards.

CAST GateKeeper adapts the quality rules from best-in-class industry standards (OWASP, CWE, CISQ). With its unique ability to perform dataflow and system-level analysis (From Presentation layer to Database layer), CAST provides the most accurate security findings, reducing a lot of false positives.

## Application Characteristics

This assessment is focused solely on the technical implementation of the said application (user interface to database), with no investigation of the functionality.

|  |  |
| --- | --- |
| Name | Value |
| kLoC | 504 |
| Files | 6,586 |
| Classes | 593 |
| SQL Art. | 0 |
| Tables | 119 |

*Fig 1: Application Technology characteristics Table 1: Application characteristics*

# Security Violation Overview

This section provides a summary of the most severe security vulnerability identified in the structural quality analysis and measurement by CAST AIP against the OWASP API Security 2023 standard.

## OWASP API -2023 Top 10 Vulnerabilities

The [OWASP API Security Risk Top 10](https://owasp.org/www-project-api-security/)focuses on strategies and solutions to understand and mitigate the unique vulnerabilities and security risks of Application Programming Interfaces (APIs).

List of OWASP API -2023 rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| OWASP-API-2023 | Total | Added | Removed |
| API1:2023 xxx | 0 | 0 | 0 |
| A… | 0 | 0 | 0 |

*Table 2: OWASP API 2023 Top 10 Rules*

## OWASP - API1:2023 Broken Object Level Authorization

APIs tend to expose endpoints that handle object identifiers, creating a wide attack surface of Object Level Access Control issues. Object level authorization checks should be considered in every function that accesses a data source using an ID from the user.

List of API:2023 Broken Object Level Authorization rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total | Added | Removed |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 3: API1:2023 Broken Object Level Authorization rules*

## OWASP - API2:2023 Broken Authentication

Authentication mechanisms are often implemented incorrectly, allowing attackers to compromise authentication tokens or to exploit implementation flaws to assume other user’s identities temporarily or permanently. Compromising system’s ability to identify the client/user, compromises API security overall.

List of API2:2023 Broken Authentication rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Vulnerabilities | Added Vulnerabilities | Removed Vulnerabilities |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 4: API2:2023 Broken Authentication rules*

## OWASP - API3:2023 Broken Object Property Level Authorization

This category combines former [API3:2019 Excessive Data Exposure](https://owasp.org/API-Security/editions/2019/en/0xa3-excessive-data-exposure/) and [API6:2019 - Mass Assignment](https://owasp.org/API-Security/editions/2019/en/0xa6-mass-assignment/), focusing on the root cause: the lack of or improper authorization validation at the object property level. This leads to information exposure or manipulation by unauthorized parties.

List of API3:2023 Broken Object Property Level Authorization rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Vulnerabilities | Added Vulnerabilities | Removed Vulnerabilities |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 5: API3:2023 Broken Object Property Level Authorization rules*

## OWASP - API4:2023 Unrestricted Resource Consumption

Satisfying API requests requires resources such as network bandwidth, CPU, memory, and storage. Other resources such as emails/SMS/phone calls or biometrics validation are made available by service providers via API integrations, and paid for per request. Successful attacks can lead to Denial of Service or an increase of operational costs.

List of API4:2023 Unrestricted Resource Consumption rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Vulnerabilities | Added Vulnerabilities | Removed Vulnerabilities |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 6: API4:2023 Unrestricted Resource Consumption rules*

## OWASP - API5:2023 Broken Function Level Authorization

Complex access control policies with different hierarchies, groups, and roles, and an unclear separation between administrative and regular functions, tend to lead to authorization flaws. By exploiting these issues, attackers can gain access to other users’ resources and/or administrative functions.

List of API5:2023 Broken Function Level Authorization rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Vulnerabilities | Added Vulnerabilities | Removed Vulnerabilities |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 7: API5:2023 Broken Function Level Authorization rules*

## OWASP - API6:2023 Unrestricted Access to Sensitive Business Flows

APIs vulnerable to this risk expose a business flow - such as buying a ticket, or posting a comment - without compensating for how the functionality could harm the business if used excessively in an automated manner. This doesn't necessarily come from implementation bugs.

List of API6:2023 Unrestricted Access to Sensitive Business Flows rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Vulnerabilities | Added Vulnerabilities | Removed Vulnerabilities |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 8: API6:2023 Unrestricted Access to Sensitive Business Flows rules*

## OWASP - API7:2023 Server Side Request Forgery

Server-Side Request Forgery (SSRF) flaws can occur when an API is fetching a remote resource without validating the user-supplied URI. This enables an attacker to coerce the application to send a crafted request to an unexpected destination, even when protected by a firewall or a VPN.

List of API7:2023 Server Side Request Forgery rules that had any findings in this application

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Vulnerabilities | Added Vulnerabilities | Removed Vulnerabilities |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 9: API7:2023 Server Side Request Forgery rules*

## OWASP - API8:2023 Security Misconfiguration

APIs and the systems supporting them typically contain complex configurations, meant to make the APIs more customizable. Software and DevOps engineers can miss these configurations, or don't follow security best practices when it comes to configuration, opening the door for different types of attacks.

List of API8:2023 Security Misconfiguration rules that had any findings in this application

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Vulnerabilities | Added Vulnerabilities | Removed Vulnerabilities |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 10: API8:2023 Security Misconfiguration rules*

## OWASP - API9:2023 Improper Inventory Management

APIs tend to expose more endpoints than traditional web applications, making proper and updated documentation highly important. A proper inventory of hosts and deployed API versions also are important to mitigate issues such as deprecated API versions and exposed debug endpoints.

List of API9:2023 Improper Inventory Management rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Vulnerabilities | Added Vulnerabilities | Removed Vulnerabilities |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 10: API9:2023 Improper Inventory Management rules*

## OWASP - API10:2023 Unsafe Consumption of APIs

Developers tend to trust data received from third-party APIs more than user input, and so tend to adopt weaker security standards. In order to compromise APIs, attackers go after integrated third-party services instead of trying to compromise the target API directly.

List of API10:2023 Unsafe Consumption of APIs rules that had any findings in this application -

|  |  |  |  |
| --- | --- | --- | --- |
| CAST Rules | Total Vulnerabilities | Added Vulnerabilities | Removed Vulnerabilities |
| Rule 1 | 0 | 0 | 0 |
| Rule 2 | 0 | 0 | 0 |
| Rule 3 | 0 | 0 | 0 |
| Rule 4 | 0 | 0 | 0 |
| Rule 5 | 0 | 0 | 0 |

*Table 12: API10:2023 Unsafe Consumption of APIs rules*

# Appendix

## About CAST Software Intelligence

Software Intelligence creates understanding into software architecture, end to end transaction flows, data access patterns and more, helping teams work confidently and faster. Hundreds of companies rely on CAST Software Intelligence to improve end-user satisfaction and time-to-market, prevent business disruption and reduce cost, enabling them to move past today’s obstacles and to tackle the next wave of innovation.

[Click here](https://www.castsoftware.com/software-intelligence) for more information about CAST Software Intelligence.

## About CAST Security

Cyber risk and application security require a proactive and intelligence-driven approach. CAST Software Intelligence shifts insight into security strategy blind spots before development starts. With its unique ability to do dataflow and system-level analysis, CAST provides the most accurate security findings, reducing a lot of false positives. CAST Security rules are adapted from best-in-class industry standards – CISQ, CWE, and OWASP.

To find out more about CAST Security, [click here](https://www.castsoftware.com/use-cases/application-security).

## Applicability of OWASP API 2023 in CAST Solution

| Standards | Description | Applicability |
| --- | --- | --- |
| API1:2023 | Broken Object Level Authorization | N |
| API2:2023 | Broken Authentication | N |
| … | … | … |