Author

CAST



**OWASP API 2019 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 -2019 Top 10 Vulnerabilities

2.2. OWASP - API1:2019 Broken Object Level Authorization

2.3. OWASP - API2:2019 Broken User Authentication

2.4. OWASP - API3:2019 Excessive Data Exposure

2.5. OWASP - API4:2019 Lack of Resources & Rate Limiting

2.6. OWASP - API5:2019 Broken Function Level Authorization

2.7. OWASP - API6:2019 Mass Assignment

2.8. OWASP - API7:2019 Security Misconfiguration

2.9. OWASP - API8:2019 Injection

2.10. OWASP - API9:2019 Improper Assets Management

2.11. OWASP - API10:2019 Insufficient Logging & Monitoring

3. Appendix

3.1. About CAST Software Intelligence

3.2. About CAST Security

3.3. Applicability of OWASP API 2019 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 the CAST Application Intelligence Platform (AIP) to automatically scan the implementation of these applications to review the architecture, design, and code against OWASP standards.

CAST AIP 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 2019 standard.

## OWASP API -2019 Top 10 Vulnerabilities

The [OWASP Top 10](https://owasp.org/www-project-api-security/)focuses on identifying the most serious web application security risks for a broad array of organizations.

List of OWASP API -2019 rules that had any findings in this application.

|  |  |  |  |
| --- | --- | --- | --- |
| OWASP-2019 | Total Vulnerabilities | Added Vulnerabilities | Removed Vulnerabilities |
| API8:2019 Injection | 0 | 0 | 0 |
| A… | 0 | 0 | 0 |

*Table 2: OWASP API 2019 Top 10 Rules*

## OWASP - API1:2019 Broken Object Level Authorization

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

List of API:2019 Broken Object 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 3: API1:2019 Broken Object Level Authorization rules*

## OWASP - API2:2019 Broken User 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 API2:2019 Broken User 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: API2:2019 Broken User Authentication rules*

## OWASP - API3:2019 Excessive Data Exposure

Looking forward to generic implementations, developers tend to expose all object properties without considering their individual sensitivity, relying on clients to perform the data filtering before displaying it to the user.

List of API3:2019 Excessive Data Exposure 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: API3:2019 Excessive Data Exposure rules*

## OWASP - API4:2019 Lack of Resources & Rate Limiting

Quite often, APIs do not impose any restrictions on the size or number of resources that can be requested by the client/user. Not only can this impact the API server performance, leading to Denial of Service (DoS), but also leaves the door open to authentication flaws such as brute force.

List of API4:2019 Lack of Resources & Rate Limiting 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: API4:2019 Lack of Resources & Rate Limiting rules*

## OWASP - API5:2019 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 gain access to other users’ resources and/or administrative functions.

List of API5:2019 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: API5:2019 Broken Function Level Authorization rules*

## OWASP - API6:2019 Mass Assignment

Binding client provided data (e.g., JSON) to data models, without proper properties filtering based on a whitelist, usually lead to Mass Assignment. Either guessing objects properties, exploring other API endpoints, reading the documentation, or providing additional object properties in request payloads, allows attackers to modify object properties they are not supposed to.

List of API6:2019 Mass Assignment 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: API6:2019 Mass Assignment rules*

## OWASP - API7:2019 Security Misconfiguration

Security misconfiguration is commonly a result of unsecure default configurations, incomplete or ad-hoc configurations, open cloud storage, misconfigured HTTP headers, unnecessary HTTP methods, permissive Cross-Origin resource sharing (CORS), and verbose error messages containing sensitive information.

List of API7:2019 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 9: API7:2019 Security Misconfiguration rules*

## OWASP - API8:2019 Injection

Injection flaws, such as SQL, NoSQL, Command Injection, etc., occur when untrusted data is sent to an interpreter as part of a command or query. The attacker’s malicious data can trick the interpreter into executing unintended commands or accessing data without proper authorization.

List of API8:2019 Injection 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: API8:2019 Injection rules*

## OWASP - API9:2019 Improper Assets Management

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

List of API9:2019 Improper Assets 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: API9:2019 Improper Assets Management rules*

## OWASP - API10:2019 Insufficient Logging & Monitoring

Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain persistence, pivot to more systems to tamper with, extract, or destroy data. Most breach studies demonstrate the time to detect a breach is over 200 days, typically detected by external parties rather than internal processes or monitoring.

List of API10:2019 Insufficient Logging & Monitoring 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: API10:2019 Insufficient Logging & Monitoring 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 2019 in CAST Solution

| Standards | Description | Applicability |
| --- | --- | --- |
| API1:2019 | Broken Object Level Authorization | N |
| API2:2019 | Broken User Authentication | N |
| … | … | … |