Author

CAST



**OWASP Mobile 2016**

**Compliance 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. OWASP Mobile 2016 Summary

2.1. OWASP Mobile 2016 Top 10 vulnerabilities Summary

2.2. OWASP -2016 Mobile M1 - Improper Platform Usage

2.3. OWASP -2016 Mobile M2 – Insecure Data Storage

2.4. OWASP -2016 Mobile M3 – Insecure Communication

2.5. OWASP -2016 Mobile M4 – Insecure Authentication

2.6. OWASP -2016 Mobile M5 – Insufficient Cryptography

2.7. OWASP -2016 Mobile M6 – Insecure Authorization

2.8. OWASP -2016 Mobile M7 – Client Code Quality

3. Security Violation Details

3.1. OWASP -2016 Mobile M1 – Improper Platform Usage

3.2. OWASP -2016 Mobile M2 – Insecure Data Storage

3.3. OWASP -2016 Mobile M3 – Insecure Communication

3.4. OWASP -2016 Mobile M4 – Insecure Authentication

3.5. OWASP -2016 Mobile M5 – Insufficient Cryptography

3.6. OWASP -2016 Mobile M6 – Insecure Authorization

3.7. OWASP -2016 Mobile M7 – Client Code Quality

4. Appendix

4.1. About CAST Software Intelligence

4.2. About CAST Security

# 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, PCI, NIST). 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*

# OWASP Mobile 2016 Summary

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 Mobile 2016 standard. Details about OWASP Security Standard can be found [here](https://www.owasp.org/index.php/OWASP_Mobile_Security_Project#tab=Top_10_Mobile_Risks).

## OWASP Mobile 2016 Top 10 vulnerabilities Summary

The [OWASP Top 10](https://www.owasp.org/index.php/OWASP_Mobile_Top_10#tab=Top_10_Mobile_Risks) focuses on identifying the most serious web application security risks for a broad array of organizations.

Findings summary for CAST under OWASP Mobile 2016 Standard.

|  |  |  |  |
| --- | --- | --- | --- |
| OWASP-Mobile 2016 | Total | Added | Removed |
| M1 | 0 | 0 | 0 |
| M2 | 0 | 0 | 0 |
| M3 | 0 | 0 | 0 |
| M4 | 0 | 0 | 0 |
| … | 0 | 0 | 0 |

*Table 2: OWASP Mobile 2016 Top 10 Rules*

## OWASP -2016 Mobile M1 - Improper Platform Usage

This category covers misuse of a platform feature or failure to use platform security controls. It might include Android intents, platform permissions, misuse of TouchID, the Keychain, or some other security control that is part of the mobile operating system. There are several ways that mobile apps can experience this risk.

List of M1- Improper Platform Usage vulnerabilities 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: M1- Improper Platform Usage Vulnerabilities*

## OWASP -2016 Mobile M2 – Insecure Data Storage

This covers insecure data storage and unintended data leakage.

List of M2-Insecure Data Storage vulnerabilities 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 4: M2- Insecure Data Storage Vulnerabilities*

## OWASP -2016 Mobile M3 – Insecure Communication

This covers poor handshaking, incorrect SSL versions, weak negotiation, cleartext communication of sensitive assets, etc.

List of M3-Insecure Communication vulnerabilities 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 5: M3-Insecure Communication vulnerabilities*

## OWASP -2016 Mobile M4 – Insecure Authentication

This category captures notions of authenticating the end user or bad session management. This can include: failing to identify the user at all when that should be required, failure to maintain the user's identity when it is required, weaknesses in session management

List of M4 – Insecure Authentication vulnerabilities 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 6: M4 – Insecure Authentication vulnerabilities*

## OWASP -2016 Mobile M5 – Insufficient Cryptography

The code applies cryptography to a sensitive information asset. However, the cryptography is insufficient in some way. Note that anything and everything related to TLS or SSL goes in M3. Also, if the app fails to use cryptography at all when it should, that probably belongs in M2. This category is for issues where cryptography was attempted, but it wasn't done correctly

List of M5 – Insufficient Cryptography vulnerabilities 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 7: M5 – Insufficient Cryptography vulnerabilities*

## OWASP -2016 Mobile M6 – Insecure Authorization

This is a category to capture any failures in authorization (e.g., authorization decisions in the client side, forced browsing, etc.). It is distinct from authentication issues (e.g., device enrolment, user identification, etc.). If the app does not authenticate users at all in a situation where it should (e.g., granting anonymous access to some resource or service when authenticated and authorized access is required), then that is an authentication failure not an authorization failure.

List of M6-Insecure Authorization vulnerabilities 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 8: M6- Insecure Authorization vulnerabilities*

## OWASP -2016 Mobile M7 – Client Code Quality

This was the "Security Decisions Via Untrusted Inputs", one of our lesser-used categories. This would be the catch-all for code-level implementation problems in the mobile client. That's distinct from server-side coding mistakes. This would capture things like buffer overflows, format string vulnerabilities, and various other code-level mistakes where the solution is to rewrite some code that's running on the mobile device.

List of M7 – Client Code Quality vulnerabilities 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 9: M7- Client Code Quality vulnerabilities*

# Security Violation Details

## OWASP -2016 Mobile M1 – Improper Platform Usage

|  |
| --- |
| Object name |
| Violation #1 |
| …. |
|  |

## OWASP -2016 Mobile M2 – Insecure Data Storage

|  |
| --- |
| Object name |
| Violation #1 |
| …. |

## OWASP -2016 Mobile M3 – Insecure Communication

|  |
| --- |
| Object name |
| Violation #1 |
| …. |
|  |
|  |

## OWASP -2016 Mobile M4 – Insecure Authentication

|  |
| --- |
| Object name |
| Violation #1 |
| …. |
|  |

## OWASP -2016 Mobile M5 – Insufficient Cryptography

|  |
| --- |
| Object name |
| Violation #1 |
| …. |
|  |

## OWASP -2016 Mobile M6 – Insecure Authorization

|  |
| --- |
| Object name |
| Violation #1 |
| …. |

## OWASP -2016 Mobile M7 – Client Code Quality

|  |
| --- |
| Object name |
| Violation #1 |
| …. |

# 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, PCI, NIST and OWASP.

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