Five Cedars

Alternate Format Ballot (AFB) Source Code Review Test Report for California

CDL-17018-CSTP-01

Prepared for:

Vendor Name	*Five Cedars*
Vendor System	*Alternate Format Ballot (AFB) v4.3*

Prepared by:

4720 Independence St.
Wheat Ridge, CO 80033
303-422-1566
www.SLICompliance.com

Accredited by the Election Assistance Commission (EAC) for Selected Voting System Test Methods or Services

Revision History

Date	Release	Author	Revision Summary
8.28.2017	1.0	*M. Santos*	Initial Release
8.29.2017	1.1	*M. Santos*	Updates for CASOS comments

Disclaimer

The information reported herein must not be used by the client to claim product certification, approval, or endorsement by NVLAP, NIST, or any agency of the Federal Government.

Trademarks

- SLI is a registered trademark of SLI Compliance.
- All products and company names are used for identification purposes only and may be trademarks of their respective owners.

TABLE OF CONTENTS

INTRODUCTION

This Test Report outlines the test approach SLI Compliance (SLI) followed when performing Software Testing on the ***Five Cedars Alternate Format Ballot (AFB)*** *voting system* against the California Voting System Standards (CVSS). The purpose of this document is to provide an understanding of the work SLI conducted.

The California Voting Systems Standards (CVSS) were written in such a way to be applicable to a wide variety of voting technology. Therefore, the relevant portions of the CVSS are reviewed as they relate to the Remote Accessible Vote By Mail (RAVBM) for purposes of this report. The use of "voting system" shall apply to the RAVBM system.

The AFB application is an HTML 5 SPA (Single Page Application), which means that once the initial server call for the application is processed the entire application runs in the current browser session.

REVIEW SPECIFICATIONS

The following are the specifications for source code testing conducted on the **Five Cedars AFB** remote accessible vote by mail system (RAVBMS).

Source Code Review

The testing of the **Five Cedars AFB** Remote Accessible Vote By Mail (RAVBM) includes proprietary source code. The **Five Cedars AFB** voting system code was tested to the applicable CVSS requirements, and any applicable industry standards, as detailed below.

SLI conducted a source code review of the source code for compliance to the CVSS.

The source code was reviewed for adherence to the applicable standards in sections 5 and 7 of the CVSS.

- The expected outcome was that no issue would be found.
- The actual outcome was a determination that were ten discrepancies written for the "Sufficient Header Comments" (CVSS 5.2.6.a-h) requirement found in the source code base reviewed, as a result, ten discrepancies were written against the code base.

The source code was reviewed for adherence to other applicable coding format conventions and standards including best practices for the coding language used.

- The expected outcome for this review was that no issue would be found.
- The actual outcome for this review was a determination that the source code was clean and met all CVSS and applicable standards requirements in this category.

Analysis of the program logic and branching structure

- The expected outcome was that no issue would be found.
- The actual outcome was a determination that the program logic and branching structure was reasonable and sufficient for the functionality implemented.

Evaluate whether the system is designed in a way that allows meaningful analysis, including:

  - Whether the architecture and code is amenable to an external review
  - Whether code analysis tools can be usefully applied
  - Whether the code complexity is at a level that obfuscates its logic

- The expected outcome was that no issue would be found.
- The actual outcome was a determination that the architecture and code is amenable to external review and that the code complexity does not obfuscate the logic. Code analysis tools could be applied to this code base, but it is of a small quantity that manual review was as useful, if not more so.

The AFB source code was searched for exposures to commonly exploited vulnerabilities including buffer overflows and SQL issues.

- The expected outcome for this review was that no exposures to commonly exploited vulnerabilities would be found in the AFB source code.
- The actual outcome for this review was a determination that no exposures to commonly exploited vulnerabilities were found in the AFB source code.

The AFB source code was evaluated for the use and correct implementation of cryptography and key management.

- The expected outcome for this review was that cryptography and key management would be found to be correctly implemented in the AFB source code, as per the CVSS.

- The actual outcome for this review was a determination that cryptography and key management is correctly implemented in the AFB source code.

The AFB source code was analyzed for its ability to appropriately accommodate error and exception handling.

- The expected outcome for this review was that no issues with error and exception handling would be found in the AFB source code.
- The actual outcome for this review was a determination that no error and exception handling issues were found in the AFB source code.

The AFB source code was evaluated in two areas for the likelihood of security failures being detected.

a. Evaluate whether audit mechanisms are reliable and tamper resistant.
   - The expected outcome for this review was that audit mechanisms in the AFB source code would be found to be reliable and tamper resistant.
   - The actual outcome for this review that no issues were found – audit mechanisms in the AFB source code were found to be reliable and tamper resistant.

b. Evaluate whether data that might be subject to tampering is properly validated and authenticated.
   - The expected outcome for this review was that any data in the AFB source code that might be subject to tampering would be properly validated and authenticated.
   - The actual outcome for this review was that no issues were found – any data in the AFB source code that might be subject to tampering is properly validated and authenticated.

The AFB source code was evaluated for the risk that a user can escalate his or her capabilities beyond those authorized.

- The expected outcome for this review was that in the AFB source code, a user cannot escalate his or her capabilities beyond those authorized.
- The actual outcome for this review was a determination that in the AFB source code, a user cannot escalate his or her capabilities beyond those authorized.

The AFB source code was evaluated for embedded, exploitable code (such as “Easter eggs”) that can be triggered to affect the system.

- The expected outcome for this review was that no embedded, exploitable code (such as “Easter eggs”) that can be triggered to affect the system would be found to be resident in the AFB source code.

- The actual outcome for this review was a determination that no embedded, exploitable code (such as “Easter eggs”) that can be triggered to affect the system was found to be resident in the AFB source code.

The AFB source code was evaluated to determine that no code for dynamic memory access features which would permit the replacement of certificated executable code or control data or insertion of exploitable code or data would be found.

- The expected outcome for this review was that code for dynamic memory access features which would permit the replacement of certificated executable code or control data or insertion of exploitable code or data would not be found in the AFB source code.
- The actual outcome for this review was a determination that no code for dynamic memory access features which would permit the replacement of certificated executable code or control data or insertion of exploitable code or data was found in the AFB source code.

The AFB source code was evaluated for use of runtime scripts, instructions, or other control data that can affect the operation of security relevant functions or the integrity of the data.

- The expected outcome for this review was that no use of runtime scripts, instructions, or other control data would be found in the AFB source code.
- The actual outcome for this review was a determination that no use of runtime scripts, instructions, or other control data that can affect the operation of security relevant functions or the integrity of the data was found in the AFB source code.

The AFB source code was evaluated to determine that no code for dynamic memory access features which would permit the replacement of certificated executable code or control data or insertion of exploitable code or data would be found.

- The expected outcome for this review was that code for dynamic memory access features which would permit the replacement of certificated executable code or control data or insertion of exploitable code or data would not be found in the AFB source code.
- The actual outcome for this review was a determination that no code for dynamic memory access features which would permit the replacement of certificated executable code or control data or insertion of exploitable code or data was found in the AFB source code.

The AFB source code was evaluated for design and implementation to ensure that sound, generally accepted engineering practices are followed, checking to verify that code is defensively written against bad data, errors in other modules, changes in environment, user errors, and other adverse conditions

- The expected outcome for this review was that generally accepted engineering practices are followed and the code is defensively written in the AFB source code.
- The expected outcome for this review was a determination that in the AFB source code, generally accepted engineering practices are followed and the code is defensively written against bad data, errors in other modules, changes in environment, user errors, and any other potential adverse conditions.

REVIEW RESULTS

Discrepancies

Discrepancies are reported such that the California Secretary of State is provided with a basis for evaluating the extent to which the source code meets applicable standards.

Ten discrepancies for the “Sufficient Header comments” requirement were found in the AFB source code base reviewed, as a result, ten discrepancies were written against the code base.

The discrepancies reported do not impact the functionality nor does it affect the operation of the voting system. Therefore, SLI does not recommend that fixing these discrepancies be required.

Vulnerabilities

For any vulnerabilities discovered, SLI was tasked with identifying the particular standards applicable to each vulnerability.

To the extent possible, reported vulnerabilities included an indication of whether the exploitation of the vulnerability would require access by:

- A Voter. Voters usually have low knowledge of the Remote Accessible Vote by Mail Machine System (RAVBMS) design and configuration. Some may have more advanced knowledge. A voter may carry out attacks designed by others.
- An Elections official insider. Elections official have a wide range of knowledge of the RAVBMS design and configuration. An official may have

unrestricted access to the RAVBMS for long periods of time. Their designated activities include:

- Set up and pre-election procedures;
- Election operation;
- Post-election processing of results; and
- Archiving and storage operations.

- A Vendor insider: A vendor insider has great knowledge of the RAVBMS design and configuration. They have unlimited access to the RAVBMS before it is delivered to the purchaser and, thereafter, may have unrestricted access when performing warranty and maintenance service and when providing election administration services.

No vulnerabilities were found within the source code reviewed, as a result, no findings were written against the code base.

Summary

Ten discrepancy findings were located within the AFB RAVBMS system.

No potential vulnerabilities were identified within the AFB code base.

Within the AFB code base, all findings were low risk vulnerabilities that would require an in-depth knowledge of the code base and how it operates to be able to successfully subvert the system. To exploit them successfully, it would require modifying the code.

As per the direction given by the California Secretary of State, this software testing report does not include any recommendation as to whether or not the system should be approved.

End of Software Test Report