Dominion Democracy Suite 5.19
Voting System Software Test Report
for
California Secretary of State

DOM-23001-CSTR-01

Prepared for:

Vendor Name	*Dominion Voting Systems*
Vendor System	*Democracy Suite 5.19*

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
10/30/2024	1.0	*B. Roberson*	Initial Release

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 report outlines the test approach SLI Compliance (SLI) followed when performing Software Testing on the **Dominion Democracy Suite 5.19 (Dominion DS 5.19)** Voting System against the California Voting System Standards (CVSS). The purpose of this document is to provide an understanding of the work SLI conducted.

REVIEW SPECIFICATIONS

The following are the specifications for source code testing conducted on the **Dominion DS 5.19** voting system.

Source Code Review

The **Dominion DS 5.19** voting system includes proprietary software and firmware. The voting system code base was tested to the applicable CVSS requirements.

Review of the code included:

- Adherence to the applicable standards in sections 5 and 7 of the CVSS.
- Adherence to other applicable coding format conventions and standards including best practices for the coding language used.
- Analysis of the program logic and branching structure.
- Evaluate whether the system is designed in a way that allows meaningful analysis, including:
  - Whether the architecture and code are 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.

Security considerations reviewed against the code base included:

- Search for exposures to commonly exploited vulnerabilities.
- Evaluation of the use and correct implementation of cryptography and key management.
- Analysis of error and exception handling.
- Evaluation of the likelihood of security failures being detected.
  - Evaluation of whether audit mechanisms are reliable and tamper resistant.
  - Evaluation of whether data that might be subject to tampering is properly validated and authenticated.
- Evaluation of the risk that a user can escalate his or her capabilities beyond those authorized.

- Evaluation of the 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.
  - Other adverse conditions.
- Evaluation for embedded, exploitable code (such as “Easter eggs”) that can be triggered to affect the system.
- Evaluation of the 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.
- Evaluate the code 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.

Components and coding languages involved in the voting system applications are shown in Table 1.

Table 1 – **Dominion DS 5.19** Components

Component	Language/s	Lines of Code	Standard
ImageCast Central (ICC)	C/C++	475,433	CPlusPlus_CodingStandard-5.19-CA.pdf
ImageCast X (ICX)	Java	235,764	dvs_JavaCodingStandards.pdf
ImageCast Evolution (ICE)	C/C++	789584	CPlusPlus_CodingStandard-5.19-CA.pdf
Democracy Suite Election Management System (EMS)	C#	1,923,782	Csharp_AutomatedCodeReview-5.19-CA.pdf
ImageCast Adjudication (ADJ)	C#	77,307	Csharp_AutomatedCodeReview-5.19-CA.pdf
ImageCast Precinct 2 (ICP2)	C/C++	448,605	CPlusPlus_CodingStandard-5.19-CA.pdf

Source Code Review Tools utilized by SLI included:

- Module Finder: an SLI proprietary application used to parse module names from C/C++, Java and VB code and populate the identified module names into the review documents.
- CheckMarx: a commercial application used to review code to stated requirements.

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.

EMS source code review

No source code requirements were found to be an issue within the EMS source code base reviewed; as a result, no discrepancies were written against the code base.

ICP2 source code review

No source code requirements were found to be an issue within the ICP2 source code base reviewed; as a result, no discrepancies were written against the code base.

ICX source code review

No source code requirements were found to be an issue within the ICX source code base reviewed; as a result, no discrepancies were written against the code base.

ICE source code review

No source code requirements were found to be at issue within the ICE source code base reviewed; as a result, no discrepancies were written against the code base.

ICC source code review

No source code requirements were found to be at issue within the ICC source code base reviewed; as a result, no discrepancies were written against the code base.

ADJ source code review

No source code requirements were found to be at issue within the ADJ source code base reviewed; as a result, no discrepancies were written against the code base.

Vulnerabilities

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

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

- Voter: Usually has low knowledge of the voting machine design and configuration. Some may have more advanced knowledge. May carry out attacks designed by others. They have access to the machine(s) for less than an hour.
- Poll worker: Usually has low knowledge of the voting machine design and configuration. Some may have more advanced knowledge. May carry out attacks designed by others. They have access to the machine(s) for up to one week, but all physical security has been put into place before the machines are received.
- Elections official insider: Wide range of knowledge of the voting machine design and configuration. May have unrestricted access to the machine 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.
- Vendor insider: Great knowledge of the voting machine design and configuration. They have unlimited access to the machine 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.

SLI will not verify or demonstrate exploitability of the vulnerability, but the report of the vulnerability will identify factors involved in the exploitation.

Any vulnerability theories developed by the source code review team members shall, to the extent possible, be referred to the Secretary of State staff.

EMS source code vulnerability review

No source code vulnerabilities were found to be an issue within the EMS source code base reviewed; as a result, no discrepancies were written against the code base.

ICP2 source code vulnerability review

No source code vulnerabilities were found to be an issue within the ICP2 source code base reviewed; as a result, no discrepancies were written against the code base.

ICX source code vulnerability review

No source code vulnerabilities were found to be an issue within the ICX source code base reviewed; as a result, no discrepancies were written against the code base.

ICE source code vulnerability review

No source code vulnerabilities were found to be an issue within the ICC source code base reviewed; as a result, no discrepancies were written against the code base.

ADJ source code vulnerability review

One improper error handling source code vulnerability was found to be an issue within the ADJ source code base reviewed. The issue was determined to be of low severity and have minimal impact on the operation of the system. It should be noted manual verification found this to only be exploitable by someone with extensive knowledge of the system such as a vendor insider and that the issue can be prevented by ensuring proper use procedures are implemented and followed.

FINAL REPORT

No discrepancy findings were located within the **Dominion DS 5.19** code base.

One vulnerability was identified within the **Dominion DS 5.19** code base. The vulnerability was examined and categorized as low risk.

As directed 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