



New York State Technology Enterprise Corporation

Report to New York State
Board of Elections

for

NYSTEC Independent Review of CIBER
Functional Security Test Plans, by Machine

Submitted to:

New York State Board of Elections
40 Steuben Place, Albany NY 12207

December 18, 2006

Final

Introduction:

NYSTEC has been engaged by the SBOE to facilitate an independent review of security test plans being developed by CIBER for Electronic Voting Machine security certification. The scope of this document includes the individual functional security test plans for the six voting systems that are in scope.

NYSTEC received copies of these Functional Security Test Plans, by system, from CIBER on 12/8/06.

The following are NYSTEC's comments based on the outcome of its review of these plans.

Executive Summary:

NYSTEC's overall finding is that the contents of the provided plans will not facilitate an effective and timely security review of the in-scope voting systems. The test plans are incomplete and inconsistent across systems in many areas, and they did not incorporate previous NYSTEC suggestions. The format of the plans is basically a listing of all the requirements found in the various federal and state regulations and guidelines — with extremely limited individual test cases cited to address each regulation. While this approach may provide testing for each requirement, the existence and effectiveness of security controls cannot be determined with the level of confidence necessary for electronic voting systems.

A more effective approach would be to develop test cases that package the tests by use case group — such as Auditing, Logging, Physical Security, Logical Security, etc. — for the major categories and functions of the voting systems, and to then map the requirements back to the test cases until all requirements are addressed. This, by the way, is how the master security test plan defines the approach; and the categories of test groups within that master plan would be a good place to start. Security tests on voting systems must be repeatable, i.e., they must produce matching results regardless of who executes them. The test plans as written do not achieve this goal.

NYSTEC Review Process:

As can be seen in the attached spreadsheet, NYSTEC's initial approach was to take the plans as delivered and review the test cases one by one to evaluate the effectiveness of each test. In order to complete this in a timely fashion, NYSTEC assigned four of the plans to four different team members. After reviewing a number of the cases and discussing the preliminary findings, NYSTEC found that this would not, in the end, provide valuable input to CIBER on how the plans should be packaged. For the purpose of this review, NYSTEC abandoned that approach and decided that the best approach would be to define a better way of packaging the tests.

Detailed Findings and Recommendations:

In order to assist CIBER in developing effective test cases, NYSTEC thinks it is worthwhile to provide the common findings from its review team. Those comments (numbered), followed by a recommendation (bulleted), are as follows:

1. Test plan editors may not have referenced the testing guidelines in VVSG Vol 2 — specifically sections 2.6 and 2.7, which describe the vendor requirements to provide testing specifications and the areas in which to focus testing efforts.

NYSTEC believes that CIBER must utilize the VVSG-required, vendor-provided testing procedures and information from these sections and use them as a starting point to develop test cases.

NYSTEC understands that it is impractical to test every condition; however, additional tests for effectiveness of controls are required here, and the VVSG does list the areas that are more important.

CIBER needs to utilize the TDP reviews and to incorporate all vendor-supplied testing specifications required in VVSG Vol 2 2.6 and 2.7.

2. Tests for each of the components (DRE, EMS, scanner, AutoMark) of the voting system were often missing. For example, a physical security test may be there for the ES&S OpScan, while there is no corresponding test for the AutoMark.

Quality assurance must be completed for each test plan to ensure that all components are tested.

3. Some test cases are unclear as to whether the steps being presented are for the System (OpScan or DRE), the Election Management Software (EMS), and/or the AutoMark. It is unclear to NYSTEC whether these test cases are incomplete or whether CIBER felt it was unnecessary to test certain components.

All tests and corresponding test procedures should clearly indicate which system the tests are to be performed on.

All tests should also have a separate result by component tested.

4. Many of the test steps state that the tester should go through the system documentation to find out if the system does something or has specific functionality.

This work should be done by the person creating the test plan, not by the tester. CIBER has completed this task as part of the TDP review; the resulting mappings should be present in each test case.

5. Many test cases indicate reviewing documentation for completeness. This review should have been done as part of the TDP review step in the plan.

References in the test plan should only be to documentation if the documentation is instructional to accomplish the test and to prove or disprove the existence or effectiveness of a particular control.

6. NYSTEC sees little value in testing the OS environments for EMS software because NYS counties will not be purchasing these systems from vendors — they, in fact, will be loading EMS software on their own systems.

NYSTEC recommends that all EMS-related security testing focus on the EMS software itself and on testing for any configuration settings or required installation steps on the EMS platform.

All EMS testing must be preceded by an EMS system setup during which all vendor installation tasks have been completed. The EMS platform configuration settings per vendor instructions will be tested, but there will be no overall EMS platform vulnerability testing.

7. Test cases are not in logical order. In other words, one test case could end up breaking something in the system that would need to be rebuilt for the next test case. Along these same lines, many procedures are repeated across requirements.

Where feasible, combine requirements within test cases, when those requirements require similar steps or cover the same components. Prepare test cases in a sequence that takes advantage of the system setup to reduce the steps required to rebuild or to start over.

Clearly indicate the mode (Test, Poll, etc.) in which the system should be.

8. Different approaches are presented for performing a given test. For example, Test Case 15 in the Diebold AccuVote OS lists other requirements that cover the requirement. However, the Sequoia Test provides actual steps on how to perform the test and makes no reference to other requirements that are related.

While the detailed steps may be different for each machine, the approach should be the same.

9. Many tests refer to “reviewing the logs,” but makes no reference to how or from where to retrieve the logs.

Provide log locations and instructions on how to retrieve the logs. This will save the tester valuable time in performing the test.

10. Many of the test plans lack any element of code review. If a code review is specified in a test plan for a particular machine, one would logically expect a code review to occur for the same requirement on the other test machines.

Provide consistent use of test procedure where feasible.

Incorporate code reviews into all relevant requirements.

11. The requirement rationale for each requirement is frequently off base — not reflecting the intention of the requirement.

Ensure that the rationale truly reflects the theme of the requirement. The rationale is beneficial, since it can help determine if the test case developer understands the requirement.

12. None of the test cases indicate if the requirement was also addressed either in the functional test or during the source code review.

Incorporate these references.

13. Most procedures are vague — they depend too much on the tester’s decisions, judgment, ability to do research, time available to spend, etc. This creates a test plan that would enable two different testers easily coming up with different outcomes.

Each test case requires detailed setup and process steps to guarantee consistency from tester to tester.

Recommendation:

Complying with Detailed Finding 1 should provide significant guidance in formatting the test cases and solve many of the other findings.

NYSTEC recommends that all of the findings and recommendations in this document be utilized to develop and finalize a comprehensive and effective Functional Security review process for each in-scope voting system.