EVEREST PROJECT

Ohio Secretary of State

ES&S SYSTEM

MICROSOLVED, INC.

TECHNICAL MANAGER'S REPORT

CONFIDENTIAL[1]

[1] This report is released by Ohio Secretary of State Jennifer Brunner consistent with the Ohio Public Records Act, Ohio R.C. 149.43. The reader of this document is advised that any conduct intended to interfere with any election, including tampering with, defacing, impairing the use of , destroying, or otherwise changing a ballot, voting machine, marking device, or piece of tabulating equipment, is inconsistent with Ohio law and may result in a felony conviction under, among other sections, Ohio R.C. 3599.24 and 3599.27.

Table of Contents

Table of Contents

Overview

The Ohio Secretary of State (SoS) retained the services of MicroSolved, Inc. (MSI) as a part of the overall EVEREST project to examine the security of the electronic voting systems in use in Ohio. As a part of that study, the MSI team performed red team penetration tests against the ES&S voting system and attempted to identify attacks that could be exploited against the confidentiality, integrity and availability of the system and/or the overall elections processes. This report details the methodology, findings and results of the ES&S system testing.

This report is report number two in a series of three reports. This report is geared toward explaining the general processes undertaken to review the ES&S system, explaining the various phases of the work, identifying the overall issues found and attempting to provide root causes for the problems. The report also contains general suggestions for improvement and mitigation of the discovered issues and comparison of the system against a twelve step framework of best practices. An executive summary of the process and findings (report #1) and a specific catalog of technical findings (report #3) were delivered alongside this report to the SoS. Please see the appropriate report if you seek more general or more specific information.

The MSI team tested the ES&S systems without any access to the source code of the components. Attacks were performed by emulating both the common access of the voter at the precinct level and access that is available to various people who come into contact with the systems during their life-span - from deployment and implementation to the regular access members of the board of elections, etc.

The overall results of the testing showed serious vulnerabilities in the system and its components. These vulnerabilities demonstrate the capability for attackers to execute arbitrary code on many of the components given access to them. Further, specific scenarios were identified where attackers who successfully gained access to the system and exploited identified vulnerabilities could likely impact the results of elections. Generally speaking, the vulnerabilities identified in the study stem largely from the lack of adoption of industry standard best practices that have been developed for the IT industry over the last several years. Adoption of the best practices for IT systems, networking, information security and application development as suggested by NIST, the Center for Internet Security, OWASP, SANS and other working groups would eliminate a large amount of the risk associated with the findings contained in this report.

General Testing Information

The testing of the ES&S systems was conducted onsite at the facility provided by the SoS. Our testing process took place from November 5th, 2007 through November 16th, 2007. The MSI team was provided basic training on the systems from ES&S. This training was roughly equivalent to the training provided to poll workers on the general use of the system and the deployment in the polling place. MSI did not have access to the source code of the applications nor to any specific “insider information” other than data that was publicly available from the vendor and from the Internet. MSI was provided with access to the systems in an unrestricted manner for the purposes of testing. This access to the systems was used to identify the vulnerabilities of the system. Obviously, attackers would not be given such wide access to the systems in question, thus we take this into consideration when we discuss the identified issues. However, it should be noted that access could likely be obtained by determined and/or well-resourced attackers through a variety of means ranging from bribery and breaking-and-entering to social engineering and outright coercion. History has shown that determined attackers often find powerful ways to gain access to their targets.

ES&S System Information

The following components were tested as a part of this study:

DEVICE	MODEL OR VERSION NUMBER
Unity Election Management Software	3.0.1.1
Automark	87000 with CF memory card media and paper ballots
3 iVotronic DRE units	90998-BL, 91057-BL & 93038-BL including CF memory card media, serial printers and PEB units
Precinct Optical Scanner	Model 100 with PCMCIA memory card media, paper ballots and ballot box
Central Optical Scanner	Model 650 with zip disk media and paper ballots
Windows 2003 Small Business Server	Dell hardware - used for additional storage of elections data
Windows XP Professional Workstation	Dell hardware - used to manage the election, host of the Unity software

General System Operation

The ES&S system is a widely distributed system with groups of components located at each precinct (polling place) and another group of components located at the central Board of Elections (BOE). Communication between the decentralized components and the centralized components takes place in Ohio via the human movement of PEB and PCMCIA memory cards holding the election information and the individual voting machine recorded ballots. In Ohio, no network connection or modem use is permitted between the decentralized precincts and the centralized Boards of Election.

It should also be noted that the memory cards are not the legal and official ballot of record in Ohio. The paper tapes generated by each voting machine are, in fact, the ballot of record and are the legal representation of the ballots cast by the voters. This is especially important to remember as attacks against the electronic systems are discussed. Attacks that modify the electronic records but not the paper records, or disruption/destruction of the electronic records could likely be performed, but if auditing against the paper records showed inconsistencies or errors, or if the electronic records were unavailable, the election would be decided based upon the paper tape records of the machine.

Voters interact with the precinct voting systems and their information is returned to the Board of Elections to be processed, recorded and tallied to determine the election results. Each memory card is read into the central Unity server that performs the tally and results reporting. The Unity server can be thought of as the election system “brain”.

Methodology Overview

The methodology used for the study was MSI’s traditional application assessment process. It consists of the following phases: attack surface mapping, threat modeling, poor trust/cascading failure analysis, vulnerability assessment, penetration testing and reporting. As a convenience for comparing each of the three systems against one another, we also established a twelve step framework of industry standard best practices and assigned a pass/fail to each value. More information about this framework and process will be detailed in the specific section titled Baseline Comparison in this report. Each phase of the study is detailed in the sections below.

Attack Surface Mapping

The purpose of the attack surface mapping phase is to provide the team with a graphical representation of the areas of the holistic system that would be available for assault by an attacker. This process also presents a graphical format to the team for beginning to understand the relationship between the surfaces and is an excellent tool for helping the team identify bad assumptions on the part of the developers and possible areas where cascading failures of security mechanisms could carry through from component to component. The output of this phase of work is a set of graphical object maps that are intended for internal team use only.

The mapping of the ES&S system was performed with broad approaches, mapping the many areas where the system inputs or outputs data and interacts with other objects or components. The attack surface mapping revealed to the team the importance of these paper tape records and their proper handling. However, in Ohio, each county Board of Elections operates using their own policies and processes that are based upon the guidance from the SoS for handling the paper records and all other parts of the election. Throughout the testing, this circumstance would prove to be a seriously dangerous issue for the security of the elections data. Without a common, centrally managed, best practices compliant set of policies and processes it is difficult to ensure that elections data is handled with consistency and effective security across the 88 counties of Ohio. This problem is magnified by the fact that each Board of Election varies by size, capability, funding and staffing level. As such, the attack surface mapping phase helped the team identify that the security and management of the paper tape voting records is an area of the greatest importance, is a highly likely target for attackers and is likely to be an area where security controls will vary greatly in their adoption, effectiveness and use. Review of this attack surface is outside the scope of our assessment, but we highly recommend that other components of the EVEREST project explore this attack surface and identify any underlying security issues and possible mitigations.

The other issue identified in the attack surface mapping phase was that the need to protect the Unity server became apparent. Since the Unity server defines the election settings, is a key component for creating the electronic ballots and memory cards, acts as the centralized aggregator of results and performs the tally processes to determine the outcome of the election - it is THE key component to the ES&S system. Successful attacks against the integrity or availability of the Unity server could have serious consequences. The Boards of Election around Ohio take established precautions during the elections cycle to protect the Unity server, however, general questions and answers from other EVEREST project teams have indicated that protection of the Unity server may be less than satisfactory in some locations outside of the elections cycle. Again, analysis of this issue is outside of the scope of our assessment but has been turned over to other teams for exploration. Basically, the Unity server must be protected physically and from network intrusion during its entire life. Illicit access, at any point from deployment to destruction, could have serious impact on the integrity and availability of any elections performed using the system going forward. Each Board of Election should take high levels of caution to protect the Unity server at all times. Physical access must be restricted at all times using dual-person access controls to prevent anyone from being alone with the system, and it should be powered down with the hard disks relocated to a locked safe or physically secure location separate from the machine when not in use. Hopefully, other practices and processes will be identified by other EVEREST teams that will enhance the security of the ES&S system during its life cycle including before, during, after and between election cycles.

Threat Modeling

The second phase of the study was to perform modeling of the potential threats against the ES&S system. The SoS specifically requested that our assessment be based on the following attacker goals:

- Confidentiality - the attacker would like to breach the veil of ballot secrecy and identify how specific voters cast their ballot

- Integrity - the attacker would like to perform actions that impact the ability of the system to accurately reflect the will of the voters, the attacker would like to influence or modify the outcome of the election

- Availability - the attacker would like to perform actions that impact the capability for an election to be held or for the outcome to be determined in a timely fashion

- General Chaos - the attacker would like to introduce enough issues into the elections process that the general public would fail to have confidence in the Boards of Election, the Secretary of State and/or the election itself

If ANY of these capabilities are reached by the attacker, then they have successfully compromised the election or elections process. At the minimum, they would impact local races and political processes. At the maximum, they could impact the results of a national election or do severe damage to the state's reputation or public faith in the State of Ohio.

Our threat models were established using four broad ranges of threat agents or attackers. These include:

Note: Attackers may begin at one level of the threat agent model and move higher on the scale during the process of the attack. Threat agents should be classified as their highest achievement of capability.

THREAT AGENT	DETAILS
Casual External Attackers	These attackers are interested in exploration of the voting system and/or possibly performing attacks against the elections process. This group of attackers lacks any access to the systems beyond the normal interactions presented to the voting public. They do not have sufficient skills, motivation, resources or capabilities to gain access to non-public components of the system or system functions. An example of this threat agent might be an individual hacker attempting to breach the security of the elections process for personal gain or understanding. Generally, this group of attackers is unlikely to impact the elections process in any meaningful way given the extremely distributed nature of the system.

THREAT AGENT	DETAILS
Focused and/or Resourced External Attackers	These attackers are interested in performing attacks against the elections processes using larger amounts of skills, resources and capabilities. However, to fit this category, they must be unable to gain access to any components or system functions beyond those presented to the voting public. An example of this threat agent might be a group of attackers with a specific agenda who are attempting to attack the system on a wide scale. This group of threat agents has higher capabilities and may be able to inject enough issues into the elections processes to achieve the General Chaos attack goal. They are, however, unlikely to achieve any of the other goals defined in this study.
Casual Internal Attackers	These attackers have obtained the ability to access the system or components beyond those surfaces normally exposed to the general voting public. They may have gained access to core system components, software functions or other protected resources. This group of attackers holds moderate skill and no true agenda to cause harm. An example of this threat agent might be a poll worker or employee of the Board of Elections who is interested in exploring the system or components. Another example might be a hacker who uses social engineering to gain access to the system or components for the purposes of exploration, personal gain or understanding. This group of threat agents have a higher capability to achieve attacker goals. Even without a harmful agenda, they present a risk to the system based upon mistakes, inadvertent or dangerous disclosures and exposure of the system to potential threats from malware and other attack vectors. They are likely to be capable of meaningful attacks against the elections process.

THREAT AGENT	DETAILS
Focused and/or Resourced Internal Attackers	These attackers are the highest threat to the system. They have achieved access to non-public system functions or components and have great capability and desire to perform malicious activity to achieve the attacker goals. These attackers are likely highly skilled, highly resourceful and capable of creating a myriad of scenarios for gaining access to the system. An example of this threat agent might be the agents of a foreign nation state or other well-resourced organization with specific political intent. They may use bribery, coercion or social engineering to gain access to the non-public functions of the system. They are likely capable of subtle attacks that can be leveraged to achieve the attacker goals, even on a wide scale. Attackers in this threat agent group are highly likely to achieve the attacker goals with meaningful impact on the elections processes. In many cases, given specific scenarios, detection and response to these attacks may be difficult. Again, these attackers form the most significant risk to the system.

The team also utilized the STRIDE method for performing threat modeling against each of the attack surfaces. Those surfaces found to be open to exploitation (exposure nodes) were evaluated for specific forms of testing. The STRIDE method evaluates each attack surface of the system for the following types of threats:

- Spoofing
- Tampering of inputs
- Repudiation attacks
- Information leakage or disclosure
- Denial of service attacks
- Escalation of privileges

The outcome of this analysis generated our test cases for the vulnerability assessment phase of the engagement.

Poor Trusts/Cascading Failures Analysis

In this phase of the process the team begins to examine the surface maps for areas where compromise could be spread from one component to the other or be leveraged for access from external-facing components or functions to the core of the system. In this case, the team reviewed research conducted by other testing teams and reviewed the relationships of the surface maps generated in phase one. Any identified issues are added to the test cases and help the team to focus on important exposure nodes during the vulnerability assessment phase.

Given the various types of media in use in the ES&S system, the team identified that failures to secure the integrity of any of those media components could cascade into security issues for the Unity server. If the PEB device, CF or PCMCIA memory cards or zip disks could be altered to deliver malware or illicit data to the Unity server, then the integrity of the Unity server could be affected. This situation is made all the more risky by the lack of proper anti-virus and adequate security controls on the Unity server itself. Attackers who gain illicit access to one or more of the memory media, or who can introduce Trojan memory media into the elections cycle could pose a grave danger to the elections processes.

Additionally, given the high amounts of human access to the system components provided to insiders, the team identified that best practice-based security policies and processes were a critical component as well. Human failures, dishonesty, incompetence or malicious behaviors from poll workers, members of the Boards of Elections or other key people could greatly influence the achievement of attacker goals. In our experience, and after discussions with the EVEREST project team, we assert that proper policies and processes are critical components of information security initiatives and a requirement for compliance with best practices. Again, given that this finding is outside of the scope of our assessment, we urge the SoS, Boards of Election and other key elements of the elections process to expend resources to study, compile, approve and implement a series of best practice-focused security policies and processes across all counties. If needed, the Boards of Election, should create an advisory council or steering committee of various membership with a defined charter of creating these policies and processes, working with the SoS to audit their adoption and implementation and to periodically update them as threats, controls and technology continue to evolve.

Vulnerability Assessment

Now that the attack surfaces of the components had been identified and analyzed, the vulnerability assessment phase was undertaken. In this phase we performed systematic testing of the surfaces to identify the presence of any known or unknown vulnerabilities.

It should be noted that the vulnerability assessment phase emulated the various groups of threat agents and performed testing as appropriate for each group. That is to say that components and functions were tested repeatedly with various levels of access and capability.

Generally, our vulnerability assessment covered the following attack vectors:

- Physical access
    - The team tested the components for vulnerabilities through physical access. The team probed the lock mechanisms, the accessible ports of the devices and any of the input/output subsystems that were available on the components. They also disassembled some of the components in search of ways to exploit the system.
        - The system performed poorly in these tests. Methods of interacting with the DRE devices without the PEB units were identified, physical locks were easily circumvented, operating system access was gained on the Automark component and various mechanisms for interfering with the availability of the components were found.
- Network and communication access

  - The team tested the components that perform networking and communications for vulnerabilities. The team used network scanners, serial port probes, sniffing tools and exploit code to probe for exposed vulnerabilities in the communications processes of the system.
    - The system performed at a medium level in these tests. Lack of a firewall and a plethora of available network services on the Windows 2003 Server exposed the system to various levels of risk. Additionally, while attempts to tamper with serial data were unsuccessful against the components and their data flows were encrypted, a high risk vulnerability was identified that could allow attackers to alter or destroy the paper tape records on the DRE units.
- File system access
  - The team tested the components for vulnerabilities in the processing of elections data or in the way that the underlying operating system or applications interact with the file system. The team used a technique called "fuzzing" to mutate the files used in the input/output processes of the system. Fuzzing essentially tests the system by creating files with contents that are known to likely cause problems in applications and with random data of various types including strings, integers and binary data.
    - The system performed poorly in these tests. Several components were found to be vulnerable to input manipulation attacks that could introduce arbitrary code to the system. These vulnerabilities are typically leveraged by attackers to inject malware or to take control of the components themselves. The m650 Central Scanner also showed unusual behavior during these tests and counting mechanisms were successfully tampered to alter elections data in erratic ways and with obvious, but unpredictable results.

Penetration Testing

In the penetration phase, our team explored the damage of exploiting the vulnerabilities identified in the previous phase. We attempted to gain access to the components and influence the underlying performance of the components and applications. We also leveraged the security weaknesses to cascade the failures and create verified paths to the system core.

At the physical layer, the attacks against the system were extremely successful. The team found that, as expected, physical access to many of the components could be leveraged to cause availability issues, attack the integrity of the elections data and process and introduce chaos in the elections process. While physical access to the precinct equipment led to control over only one device, physical access to the centralized Board of Elections components could be used to completely compromise the election.

Physical compromise of the Automark unit was achieved by simply disassembling the cover of the device and attaching a USB keyboard to the available USB port inside the machine. Once connected, standard Windows CE commands could be used to interact with the system, capture the ES&S software and introduce malware to the system and media. The effects of this attack on the Automark are minimal however, since the component simply marks paper ballots using the touch screen mechanism for guidance and performs no tally or counting functions relevant to the election. Attackers who leveraged this access could introduce malware to change the marked ballots, but since the user visually inspects the ballot before it is scanned by an optical scanner, detection of such issues is extremely likely. Greater risk is that the CF media could be altered with malware destined for the Unity server. The CF memory cards from the Automark system hold only ballot definition data, and are not immediately returned to the Unity server for processing, but are, however, recycled from election to election and reloaded by the Unity server for each election. Thus, this could be leveraged as a mechanism to introduce malware to the system, albeit without impact on the current election cycle.

Physical attacks were also identified against the m100 precinct optical scanner. Two high risk vulnerabilities were found. The first is a simple physical manipulation of the m100 when it is in voting mode. If the attacker simultaneously depresses [redacted] [specified] buttons on the system, the m100 will automatically, and without authentication perform the poll closing function. Note that this occurs, without the administrative key when the key is set to "Vote" mode. Closing the polls in this manner produces the normal reports, but puts the m100 into a mode where no further voting is possible. To compound the problem, if the attacker knows the password to reopen the polls, (which defaults to [redacted] [defaulted password] and is a simple three digit number), then they can re-open the polls on the device and zero the totals of the ballots received so far, effectively nullifying any electronic records of the ballots inside the ballot box. This attack could be used to delete records of some votes, but is likely to be easily discovered if used on a large scale.

The second physical attack against the m100 is more difficult to detect. A vulnerability was discovered in the component that allows attackers with physical access to the PCMCIA memory card to set its write protect mode to on. This is done (and could be done accidently) by simply moving a small switch to the right on the tail of the card. If this switch is activated after the polls are opened and reset before the polls are closed, none of the ballots scanned while the write protection mechanism was engaged will be recorded to the memory card. The internal counts of the m100, and the paper tape reports will be correct and the system will function normally, but the counts of the votes scanned will not be added to the electronic media delivered to the central Board of Elections. This essentially means that an attacker could enable and disable the write protection mechanism of the memory card in surgical ways to influence the counts from the precinct scanner. To add to the level of difficulty in detection of the exploit, while the physical ballots are in the ballot box in the correct number and the paper tape shows the correct number, the memory card is delivered to the central Board of Elections where it is read and processed. The current processes in use in most polling places are a simple review of the paper tapes, which would be correct. As such, it is likely that unless close scrutiny or recounts of the precinct were performed that surgical use of this vulnerability would go undetected.

The DRE systems used by the voters at the precinct also revealed physical security issues. The penetration team determined that interaction with and rebooting of the DRE components was possible using only a simple magnet to trigger the power switch. While no control was possible from this mechanism, and there was no exposure of menus or other underlying mechanisms, such access could be used to probe the system for further vulnerabilities in the future or to cause delays and confusion in the voting process by rebooting the components during the voting process.

On the Board of Elections premises, the physical testing revealed critical weaknesses in the security configurations of the computers running the Unity software components. The computers hosting the software failed to be secured from physical attack in even basic ways. Controls deployed on the system are simply inadequate to protect the system from complete compromise should an attacker gain access to the system at any time. Attackers could leverage these weaknesses to introduce malware or directly compromise the elections data.

Both the server and the workstation lacked proper password policies, anti-virus software and basic mechanisms for managing the integrity and security of the system. In other systems, the Ohio SoS has deployed Digital Guardian to help protect the elections data and software components. These same controls should be applied to all PC-based components of all voting systems used in Ohio and appropriate white-list controls should be implemented to prevent arbitrary execution of applications, attack tools and exploits. Anti-malware software should be implemented and all

PC-based components should be hardened against attack in accordance with industry-standard best practices. By applying these basic measures that are common in the IT industry, a majority of the physical risk to these systems could be reduced to more manageable levels. As indicated earlier, physical security policies and processes should also be applied to these components to ensure that they are properly protected against tampering during use, storage and throughout their entire life cycle.

When the MSI team moved into pen-testing the network and communications mechanisms, the components performed slightly better than in the physical testing. However, problems remained in both the precinct and Board of Elections deployed equipment.

The DRE units deployed at the precinct showed a vulnerability in the printer connection and capability. The printer used by the DRE is a simple serial printer, connected to the DRE outside of the case and in full view and access to the public. This printer connection is not secured to the case by screws, cable locks or tamper tape. The testing team identified an attack in which the attacker can simply unplug the printer cable and connect it to another device such as a laptop, PDA or the like and then proceed to print their own results to the paper tape - including their own illicit voting records. In addition, the printer in use on the DRE units tested in our study have the capability to rewind the paper tape and print over the existing output. Leveraging this capability, attackers could design a program that rewinds the tape to the beginning of the spool and then overprints the existing ballot of record - thereby destroying the legal voter verified ballot data. While reprinting of this data would be possible from the memory card, the election would lose the ability to detect tampering and the public assertion that the voter had themselves validated their own data. Such exploitation could inject chaos into the elections process and achieve an attacker goal of causing reputational damage to Ohio.

Once again, at the Board of Elections deployment, the PC components hosting the Unity software applications proved to be critically vulnerable. Network attacks against both the Windows 2003 storage server and the Windows XP workstation proved to be possible. While the elections process uses a closed network, an attacker gaining access to that network in any way is likely to be able to compromise the elections process and data. Lack of firewalls on the PC devices, poor password and configuration policies and the availability of unneeded services all contribute to the risk. Given that these computers are not updated with operating system and application patches on a regular basis, public exploits, malware and known attacks grow ever more likely to be possible against them. It would be an easy task for an attacker who gains network access to compromise one or both of the computers and introduce malware to the system to alter voting data over time or outright destroy the software. Modern attack tools like rootkits and other custom forms of malware are likely to go undetected given the lack of security controls deployed.

To mitigate much of this risk, the SoS should ensure that all PC-based components are properly hardened against attack. This should include configuring the components in accordance with best practices, deployment of proper firewalls, anti-virus and other software controls. The installation and proper configuration of Digital Guardian on the component could provide additional security and integrity assurance if configured to enforce a white list of applications for execution and other rules the tool is capable of monitoring. Specific suggestions for these changes are contained in the Technical Details report provided to the SoS.

Finishing up the penetration testing phase, our team attacked the file system interaction of the components. Several vulnerabilities came to light in this testing at both the precinct and Board of Elections deployments. Many of these vulnerabilities could be used to introduce malware to the components or to cause availability issues.

On the precinct equipment, the interaction of the DRE units with their memory card proved to be extremely vulnerable. Removal of the card at specific times, or tampering with the contents of the files on the memory cards often caused unhandled exceptions in the application. These exceptions would cause the component to crash, ceasing operation until a hard reset was performed. In some cases, our team believes that attackers with deep knowledge of the DRE components could leverage these problems to introduce malware into the DRE component or its memory card and possibly succeed in getting the illicit code returned to the Unity server as discussed in the Cascading Failure analysis section of this report. While the access to the memory card is protected with tamper seals, they are easily circumvented. The software in use on the DRE system simply should be written in such a manner as to properly handle exceptions and recover in a more secure and graceful manner.

In the Board of Elections components, more critical vulnerabilities in the components and applications were identified at the file manipulation layer of the testing process. Problems with input validation and general logical protections of the voting data were identified.

For example, fuzzing of the Model 650 optical scanner's ".pr" files caused errors in the device's tabulation mechanism and firmware. Proper bounds checking on the data that the scanner receives from the file is not performed, which can be leveraged to cause buffer overflows in the firmware of the scanner or to manipulate the vote counts in the tabulation process. Attackers with access to the zip disk file system could tamper with these files and exploit the vulnerability. It should be noted that while these vulnerabilities exist, exploitation of them does not appear to be predictable. The tabulation process responds in a myriad of ways which appeared to be easily detectable and unlikely to be leveraged with accuracy that could impact the integrity of the elections process without detection.

The Unity software itself also showed several areas of exposure to file fuzzing and input formatting style attacks. Several of the applications responded poorly to illicitly tampered inputs and files. Crashes, unhandled exceptions and other improper behavior were exhibited by the software. By leveraging these vulnerabilities through either direct access or through malware, an attacker is likely to be able to damage the software or influence its proper operation and handling of vote data. The software should be updated to include proper bounds checking and other input protections.

Of the highest concern to the testing team, access at the file system level revealed two critical issues. First, using simple editor applications such as Notepad on the applications themselves from the Unity package revealed several hard coded passwords, SQL statements and other sensitive data that are hard coded in the software. These secrets appear in the binary distributions as simple strings, without encryption. Attackers who gain access to copies of these binaries are likely to be able to learn a great deal of sensitive information that could be used to design malware for specific purposes or to allow the attacker to easily compromise the software itself. Care should be taken in future versions of the software to properly encrypt sensitive data that is stored in the application binary.

Secondly, the team identified that a mechanism exists in the Unity software for the user to arbitrarily edit the voting totals. While this feature may be required for recounts or the like in some cases, such powerful capabilities to directly impact the outcome of an election should be properly authenticated and protected. In the current version, any user of the Unity software can directly edit the data using this mechanism. Future versions of the application should require multiple authentications, including multiple users and/or an authentication token or the like. While the system logs the use of this capability, attackers may be able to delete those entries or they could go unnoticed. In either case, the feature could be used to directly impact the outcome of elections and requires additional higher levels of authentication and control than are present in the application today.

Overall, the ES&S system fails to ensure the integrity and availability of the elections data. Various issues surround the components, but much of the risk seems to stem from the lack of adoption and implementation of common IT industry standard best practices. Implementation of these standards would go a long way toward increasing the overall security of the ES&S system.

Baseline Comparison

In order to provide an easy means of understanding the security posture of the voting system in use in Ohio, the MSI team created a simple framework for the baselining of each system against industry standard best practices. The framework created was adapted from the PCI standards, of which our team has deep knowledge, and we felt gave an easily grasped way to concisely aggregate the various standards and guidelines being reviewed by the EVEREST project. We feel that this framework incorporates all of the existing standards associated with both general information security and specifically with the security of electronic voting systems.

To ensure ease of communications and to create a level playing field for all the systems to be compared against, we chose to implement a system of pass/fail grading for each of the twelve requirements of the framework. Passing a category means that the system meets the best practices requirements for that area, while failing indicates that the system does not meet industry standard best practices in the mind of our team.

Below are the specific twelve areas of the framework and the score assigned to the system for each one, along with our reasoning for the score:

BEST PRACTICE	PASS/FAIL	COMMENTS
Are firewall technologies and configurations adequate to protect systems and data?	Fail	Firewalls are not deployed on the Windows 2003 Server component, PC components are generally not well configured to resist compromise
Are password implementations sufficient to provide basic security?	Fail	Passwords across the components are poorly implemented and configurations are not sufficient to enforce complex password use
Is the core data protected during storage?	Fail	The core elections data is subject to manipulation by attackers gaining access to the Unity software in any manner
Is the core data encrypted during transit?	Pass	The contents of the memory cards and other media are properly encrypted during transit, Interaction with PEB device in any meaningful way was not achieved

BEST PRACTICE	PASS/FAIL	COMMENTS
Are anti-virus applications used and up to date?	Fail	Anti-virus software is not deployed
Are the components of the system securely developed, configured and up to date?	Fail	Common programming flaws exist on the system, Sensitive information is present in the binaries
Are access controls deployed to enforce “need to know” and/or “need to access” boundaries?	Fail	Access controls are not enforced on the system, Shared accounts are in use, Powerful capabilities to alter core elections data is available to all users
Are user authentication mechanisms unique enough to provide non-repudiation?	Fail	Operators of the components use common accounts and complexity controls are not appropriately configured
Is access to the system logged, monitored and audited?	Fail	Logging is not configured in accordance with best practices on the PC components
Are the systems routinely audited and tested for new vulnerabilities?	Fail	Critical patches are missing from the Windows XP workstation hosting the Unity application
Are security policies and processes in place to adequately protect the system, its components and the core data?	Fail	Boards of Election have not created best-practice based consistent policies and processes to protect the core data

Framework Comparison Summary:

Score (Pass/Fail): 1/12

Root Cause Determination

Review of the various vulnerabilities in the system identifies a couple of specific root causes. First and most importantly, the vulnerabilities demonstrate a lack of adoption of industry standard best practices with regards to general

IT functions, networking, system and information security and secure application development. The ES&S system fails to meet most of the twelve basic best practices requirements. If ES&S would simply adopt a common set of best practices for system development, implementation and deployment, many of the underlying issues could be mitigated. If ES&S would take the best practice steps of hardening the systems in accordance with Center for Internet Security, NIST, SANS, OWASP and/or other frameworks of best practices, they could greatly enhance the security posture of the system as a whole.

The SoS implementation of Digital Guardian may also be able to assist in the efforts to better secure the system. If the Digital Guardian tool were properly configured and implemented to enforce best practices, it would greatly enhance the security of the Unity server. However, without a configuration to protect itself and the Unity server/application from common attacks, the tool does little to enhance the security of the overall system.

Lastly, a key root cause for much of the risk to the system is the lack of consistent, best practices-based security policies and processes surrounding the system. Given the roles of the SoS and the county Boards of Election, inconsistent management, implementation and handling are key reasons for concern. If the counties identified best practices with regards to the system and implemented them consistently across the state, security improvements are likely to be gained. Further, a consistent set of policies and processes would simplify the oversight of elections security and provide the public with a verifiable set of auditable requirements that are likely to increase public trust in the elections process.

Suggestions for Improvement

The first and primary step in improving the security of the ES&S system is for all parties involved to embrace industry standard best practices and enforce them through technology, policy and process and education throughout the entire system. If all of the major stake holders, from the vendor to the SoS and from the Boards of Election to the poll workers had a consistent and usable set of rules to enforce, the overall security of the system would be enhanced.

Secondly, immediate concern and mitigation of the malware risks are required. Additional controls need to be applied to all of the components to prevent the introduction of malware. Given the time required to update the code base, test and upgrade all deployed systems, additional policy and process controls over the deployment, storage, physical security and media management must be identified and implemented if proper levels of security for the voting data are to be obtained.

Specific technical resolutions to each of the identified vulnerabilities is contained in the technical details report delivered to the SoS. This report should be consulted for specific tactical enhancements and techniques to mitigate the security issues. Such mitigations should be performed as soon as possible.

Lastly, ES&S must undertake a systematic approach to mitigating the identified vulnerabilities in the system. This includes repair of the software, hardware configurations, basic deployment images, default passwords and general security posture of the system. Each issue mitigated by the vendor greatly reduces the amount of risk management that must be transferred to the counties by policy and process controls. Given the lack of resources many of the counties face, this is likely to have significant impact on the entire elections process.

Summary

The Ohio Secretary of State (SoS) retained the services of MicroSolved, Inc. (MSI) as a part of the overall EVEREST project to examine the security of the electronic voting systems in use in Ohio. As a part of that study, the MSI team performed red team penetration tests against the ES&S voting system and attempted to identify attacks that could be exploited against the confidentiality, integrity and availability of the system and/or the overall elections processes. This report details the methodology, findings and results of the ES&S system testing.

The MSI team identified several key threats to the security of the system. These threats range from common attacks such as input manipulation and malware to the specific issues in how components of the system handle error conditions. Many of these issues stem from a lack of adoption of industry standard best practices across the spectrum of the elections system, from technical implementations to policies and processes in use at the county level. Adoption of best practices and implementation of additional controls to create a defense-in-depth security posture would enhance the security of the ES&S system.

Definitions/Reference Section

Terms and Definitions:

Buffer Overflow - Writing outside the bounds of a block of allocated memory can corrupt data, crash the program, [redacted]r cause the execution of malicious code. For more information, please see: http://www.owasp.org/index.php/Buffer_Overflow

Fuzzing - Fuzz testing or Fuzzing is a Black Box software testing technique, which basically consists in finding implementation bugs using malformed/semi-malformed data injection in an automated fashion. For more information, please see: http://www.owasp.org/index.php/Fuzzing

Rootkit - A rootkit is a particulary hard to detect type of malware. Rootkits allow attackers to achieve absolute control over computer systems and their applications. For more information, please see: http://en.wikipedia.org/wiki/Rootkit

Sites for Best Practices and Frameworks:

The Center for Internet Security - http://www.cisecurity.com/

NIST (National Institute of Standards and Technology) - http://www.nist.gov/

SANS (SANS Institute) - http://www.sans.org

OWASP (The Open Web Application Application Security Project) - http://www.owasp.org

PCI DSS (Payment Card Industry Data Security Standard) - http://www.pcisecuritystandards.org

EVEREST Project Information:

Ohio Secretary of State EVEREST Project - http://www.sos.state.oh.us/sos/info/everest.aspx