System Integrity Management Requirements Gap Analysis

The System Installation requirements generally mapped well to the principles and guidelines. We did not find many gaps in our principles and guidelines, although small changes in the requirements were identified.

Other areas for future changes and development to the VVSG system integrity management requirements include:

- Two requirements (5.5.1-A, 5.5.1-B) discuss using a tamper-resistant hardware module to perform validation and integrity checking. One requirement refers to boot validation and the other checks the integrity of binaries. The NIST Security Team believes both of these requirements are necessary, but suggest the removal of ***tamper-resistant hardware module*** from the requirements. The goal is to provide a set of technology neutral requirements.

- For section 5.5.3, the NIST team is considering removing these requirements and would like feedback from the Working Group. We believe the other Principles and Guidelines, sufficiently protect against this threat.

- Requirement 5.5.3-C mentions the use of digital signatures, message authentication codes, or hashes. This NIST team would like to know the Working Group's thoughts on which cryptographic primitive is most appropriate.

- 5.5.4-D, E suggest a malware scan to be performed every 24 hours. The NIST team would like to engage the Working Group to discuss the entire concept of malware scanning, including type of systems being scanned and the method of doing so.

5.5.1-A Protecting the integrity of the boot process

Requirement:	Before boot up or initialization, electronic devices SHALL verify the integrity of the components used to boot up or initialize the electronic device using a tamper-resistant hardware module.
Applies to:	Electronic device
Test Reference:	Part 3:4.5 "Source Code Review", 5.2 "Functional Testing"
Discussion:	A tamper-resistant hardware module, such as a trusted platform module (TPM), can be used to store the cryptographic software reference information of the components that are required to boot the electronic device. The specific types of components required for booting vary by device type, but examples of these components are boot loader files and kernel modules on a PC. The device will not boot if the files have been modified or the boot storage has been removed from the voting system. This requirement augments [VVSG2005] I.7.4.6 by explicitly requiring integrity checking of components used to boot up or initialize an electronic device.
Source:	[VVSG2005] I.7.4.6-a, I.7.4.6-b, I.7.4.6-e
Principle(s)/ Guideline(s):	*Security: Software Integrity* *Voting systems prevent the unauthorized installation or modification of firmware, software, and critical configuration files.*

	Justification: This requirement mapped best to the Software Integrity principle itself. We may need a guideline to map better.

5.5.1-B Integrity verification of binaries before execution or memory load

Requirement:	Electronic devices SHALL verify the integrity of binaries (e.g., device drivers, library files, applications, and utilities) using a tamper-resistant hardware module and confirm that the binaries have been specified by the manufacturer as being required for the current voting system state before they are executed or loaded into memory.
Applies to:	Electronic device
Test Reference:	Part 3:4.5 "Source Code Review", 5.2 "Functional Testing"
Discussion:	Verifying the integrity of binaries prevents modified binaries, such as those infected with malware or inadvertently corrupted by a software or hardware failure, from being executed or loaded. A tamper-resistant hardware module, such as a trusted platform module (TPM), can be used to store the cryptographic software reference information to be used to verify integrity and voting system state specifications. Binaries that are not required for a particular state should not be executed while a device is in that state. The potential impact of permitting the binaries' execution varies depending on the state and the nature of the binaries – examples include altering or disrupting the functionality of the system. This requirement augments [VVSG2005] 7.4.6-b by mandating cryptographic software reference information as a mechanism for verifying the integrity of binaries, by specifying that binary integrity checking must be performed before binaries are executed or loaded into memory, and by requiring that only binaries specified as required for a particular voting system mode may be executed or loaded into memory during that mode.
Source:	[VVSG2005] I.7.4.6-b
Principle(s)/ Guideline(s):	*Security: Software Integrity* *Voting systems prevent the unauthorized installation or modification of firmware, software, and critical configuration files.* **Justification:** This requirement mapped best to the Software Integrity principle itself. We may need a guideline to map better.

5.5.3 Backup and recovery

Backup and recovery requirements describe minimum authorization, auditing, and protective measures, without regard to specific media.

5.5.3-A Restricting backup and restore capabilities

Requirement:	Electronic devices other than EMSs SHALL NOT provide backup or restore capabilities.
Applies to:	Electronic device
Test Reference:	Part 3:5.2 "Functional Testing"
Discussion:	Backup and restore capabilities introduce security holes into systems because backup operations could disrupt system functionality (e.g., locking files that the

	system needs to access) or give an attacker access to the system's data, and restore operations could alter system functionality or data (e.g., replacing existing files with previous versions). Therefore, use of backup and restore capabilities should be minimized. EMSs are permitted, but are not required, to have backup and restore capabilities because of the types of information they store.
Source:	[NIST05] Security Control SC-2
Principle(s)/ Guideline(s):	*Security: Data Protection* *The voting system protects sensitive data from unauthorized access, modification, or deletion.* **Justification:** By disallowing electronic devices (other than EMSs) from providing backup/restore, this prevents unauthorized access as mentioned in this guideline. *Security: System Integrity* *The voting system limits its attack surface by reducing unnecessary code, data paths, physical ports and by using other technical controls.* **Justification:** Limiting capabilities within the EMS can limit the attack surface.

5.5.3-B Restricting the performance of backups and restores

Requirement:	EMSs that provide backup or restore capabilities SHALL only permit backup and restore operations while not in the Activated state.
Applies to:	EMS
Test Reference:	Part 3:5.2 "Functional Testing"
Discussion:	Backup and restore operations should not be performed while EMSs are in the Activated state because backup operations could disrupt system functionality (e.g., locking files that the system needs to access) and restore operations could alter system functionality, vote data, etc.
Source:	[NIST05] Security Control SC-2
Principle(s)/ Guideline(s):	*Security: Data Protection* *The voting system protects sensitive data from unauthorized access, modification, or deletion.* **Justification:** Ensuring the EMSs do not perform backups while in an Activated State may keep the data safe from disruption.

5.5.3-C Authenticity and integrity of backup information

Requirement:	EMSs that perform backups SHALL create digital signatures, message authentication codes, or hashes for their backups so that their authenticity and integrity can be verified in the future.
Applies to:	EMS
Test Reference:	Part 3:5.2 "Functional Testing"
Discussion:	This requirement allows EMSs to verify the authenticity and integrity of backups before restoring them.
Source:	[NIST05] Security Control CP-9
Principle(s)/ Guideline(s):	*Security: System Integrity*

	The voting system maintains and verifies the integrity of software, firmware, and other critical components. **Justification:** The digital signatures, message authentication codes, or hashes are used to maintain integrity and allow for integrity checking of the EMS backups.

5.5.3-D Verifying backup authenticity and integrity

Requirement:	EMSs that perform restores SHALL verify the authenticity and integrity of backups before restoring them.
Applies to:	EMS
Test Reference:	Part 3:5.2 "Functional Testing"
Discussion:	
Source:	[NIST05] Security Control CP-10
Principle(s)/ Guideline(s):	*Security: System Integrity* *The voting system maintains and verifies the integrity of software, firmware, and other critical components.* **Justification:** The EMS verifies the integrity before restoring. This aligns with this guideline.

5.5.4-D Periodic malware scanning

Requirement:	EMSs SHALL be scanned for common known malware at least once every 24 hours during operation, including malware specifically targeted at voting systems.
Applies to:	EMS
Test Reference:	Part 3:5.2 "Functional Testing"
Discussion:	This identifies any current infections on the electronic device caused by common known malware. This requirement augments [VVSG2005] I.7.4.2 by specifying scanning of removable media for common known malware.
Source:	[VVSG2005] I.7.4.2
Principle(s)/ Guideline(s):	*Security: Detection/Monitoring* *The voting system employs mechanisms to protect against malware.* **Justification:** This requirement regulates how often the EMS is scanned for malware. The time intervals play a role in protecting against malware.

5.5.4-E Real-time malware scanning

Requirement:	EMSs SHALL perform real-time scanning for common known malware.
Applies to:	EMS
Test Reference:	Part 3:5.2 "Functional Testing"
Discussion:	This prevents files infected with common known malware from being executed or otherwise loaded within the electronic device. This requirement augments [VVSG2005] I.7.4.2 by specifying real-time scanning for common known malware.
Source:	[VVSG2005] I.7.4.2
Principle(s)/ Guideline(s):	*Security: Detection/Monitoring* *The voting system employs mechanisms to protect against malware.*

Justification: This requirement discusses the type of scanning performed on the EMS. This scanning is used to protect against malware in real-time.