History of NIST 800-53 Controls and SwA

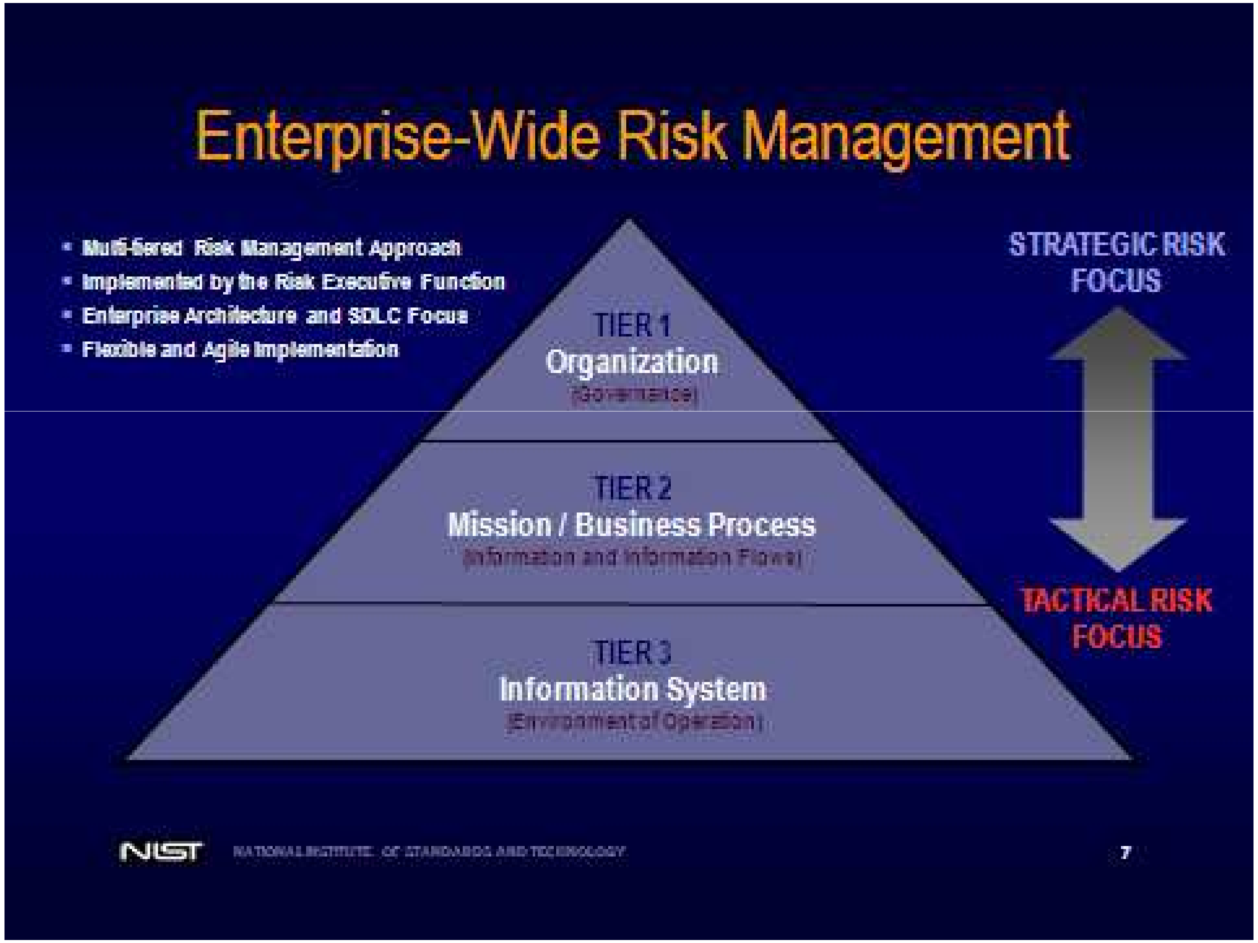
Michele Moss
SwA Working Groups
December 16, 2010

Booz | Allen | Hamilton

Prior to 800-53 Revision 3, controls that explicitly address application security principles included RA-5

- NIST SP 800-53, Control RA-5 Vulnerability Scanning
  - …..Vulnerability analysis for custom software and applications may require additional, more specialized techniques and approaches (e.g., web-based application scanners, source code reviews, source code analyzers…….

NIST Special Publication 800-39 Enterprise-Wide Risk Management: Organization, Mission, and Information Systems View

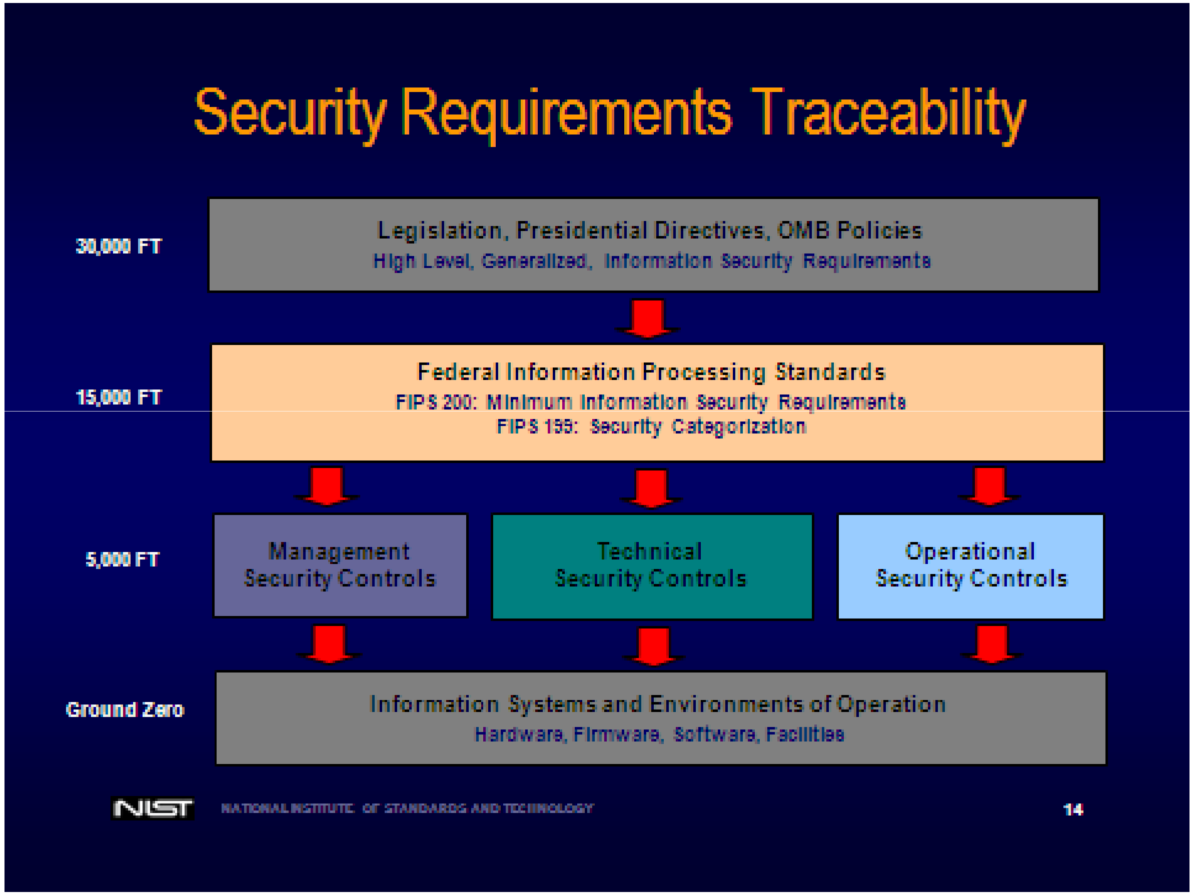


Enterprise-Wide Risk Management

- Multi-tiered Risk Management Approach
- Implemented by the Risk Executive Function
- Enterprise Architecture and SDLC Focus
- Flexible and Agile Implementation

TIER 1
Organization
(Governance)
TIER 2
Mission / Business Process
(Information and Information Flows)
TIER 3
Information System
(Environment of Operation)
STRATEGIC RISK FOCUS
TACTICAL RISK FOCUS
NIST
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
7

Security categorization provides repeatable input into security requirements by defining minimum security controls



Security Requirements Traceability

30,000 FT

Legislation, Presidential Directives, OMB Policies
High Level, Generalized, Information Security Requirements

15,000 FT

Federal Information Processing Standards
FIPS 200: Minimum Information Security Requirements
FIPS 199: Security Categorization

5,000 FT

Management Security Controls

Technical Security Controls

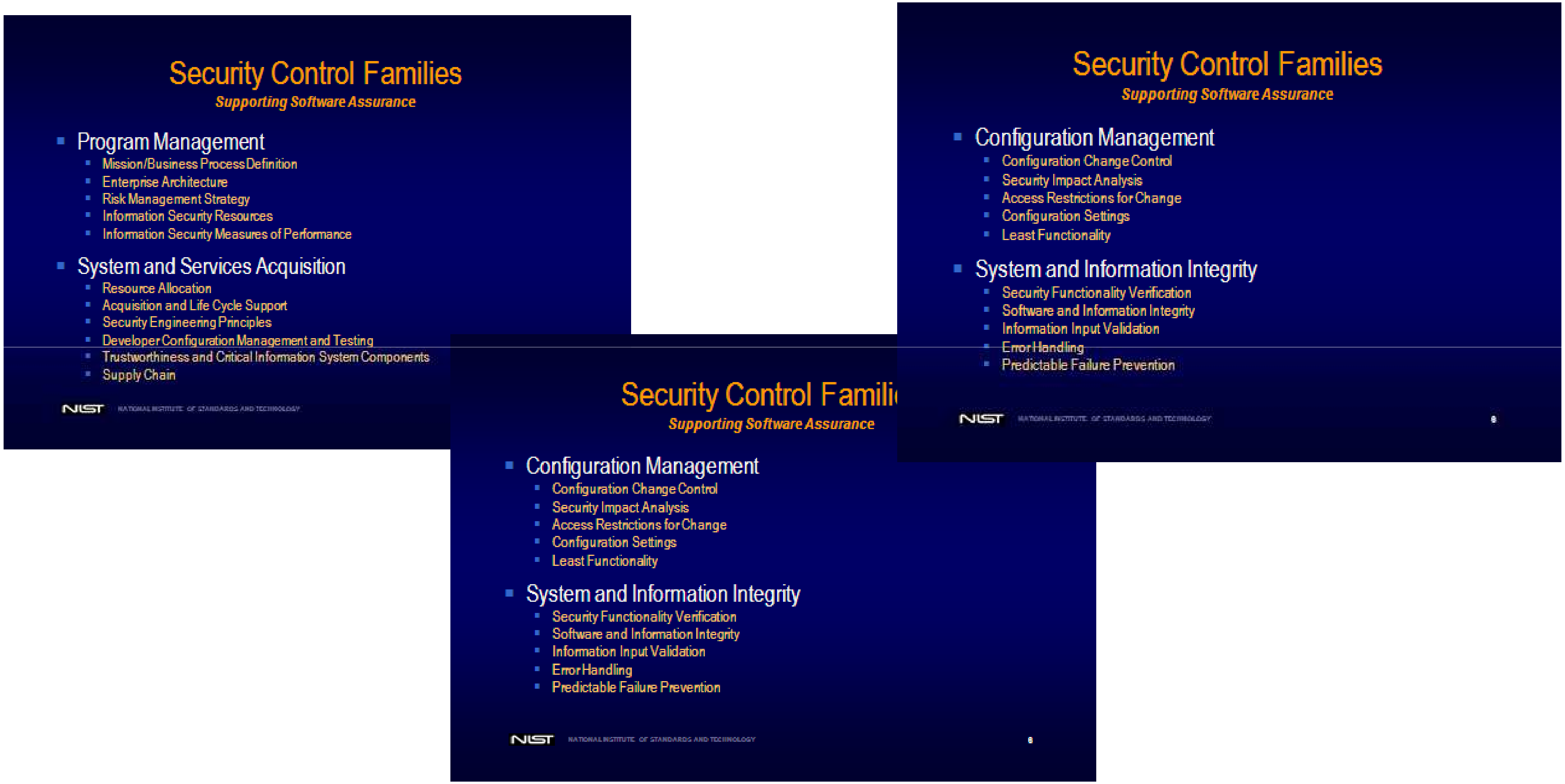
Operational Security Controls

Ground Zero

Information Systems and Environments of Operation
Hardware, Firmware, Software, Facilities

NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY 14

Security Control Families in NIST 800-53 that support Software Assurance (presented by Ron Ross at the June 2010 SwA WGs)



Security Control Families

Supporting Software Assurance

- Program Management
  - Mission/Business Process Definition
  - Enterprise Architecture
  - Risk Management Strategy
  - Information Security Resources
  - Information Security Measures of Performance
- System and Services Acquisition
  - Resource Allocation
  - Acquisition and Life Cycle Support
  - Security Engineering Principles
  - Developer Configuration Management and Testing
  - Trustworthiness and Critical Information System Components
  - Supply Chain

NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY

Security Control Families

Supporting Software Assurance

- Configuration Management
  - Configuration Change Control
  - Security Impact Analysis
  - Access Restrictions for Change
  - Configuration Settings
  - Least Functionality
- System and Information Integrity
  - Security Functionality Verification
  - Software and Information Integrity
  - Information Input Validation
  - Error Handling
  - Predictable Failure Prevention

NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY 8

Security Control Famili

Supporting Software Assurance

- Configuration Management
  - Configuration Change Control
  - Security Impact Analysis
  - Access Restrictions for Change
  - Configuration Settings
  - Least Functionality
- System and Information Integrity
  - Security Functionality Verification
  - Software and Information Integrity
  - Information Input Validation
  - Error Handling
  - Predictable Failure Prevention

NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY 8

SwA Foundations in NIST Special Publications

- NIST SP 800-64 (2008) – Table 2-1 Key Security Roles and Responsibilities In the SDLC

“Software Developer – The developer is responsible for programmatic coding regarding applications, software, and Internet/intranet sites, including “secure coding,” as well as coordinating and working with the Configuration Management (CM) manager to identify, resolve, and implement controls and other CM issues “

- CAG Critical Control 7: Application Software Security

CM-7, RA-5 (a, 1), SA-3, SA-4 (3), SA-8, SI-3, SI-10

- Next Steps
  - Where can we find SwA Foundations in NIST SP 800-53 Rev 3?
  - Where are there gaps and how do we fill them

AC-4 INFORMATION FLOW ENFORCEMENT

- Control: The information system enforces approved authorizations for controlling the flow of information within the system and between interconnected systems in accordance with applicable policy.
  - ***(1) The information system enforces information flow control using explicit security attributes on information, source, and destination objects as a basis for flow control decisions.***
  - ***(8) The information system enforces information flow control using [Assignment: organization-defined security policy filters] as a basis for flow control decisions.***

CA-2 SECURITY ASSESSMENTS

- *Control:* The organization:

  Supplemental Guidance: The organization assesses the security controls in an information system as part of: (i) security authorization or reauthorization; (ii) meeting the FISMA requirement for annual assessments; (iii) continuous monitoring; and (iv) ***testing/evaluation of the information system as part of the system development life cycle process.*** The assessment report documents the assessment results in sufficient detail as deemed necessary by the organization, to determine the accuracy and completeness of the report and whether the security controls are implemented correctly, operating as intended, and producing the desired outcome with respect to meeting the security requirements of the information system.

SA-3 LIFE CYCLE SUPPORT

- Control: The organization:
  - *a****. Manages the information system using a system development life cycle methodology that includes information security considerations;***
  - ***b. Defines and documents information system security roles and responsibilities throughout the system development life cycle; and***
  - ***c. Identifies individuals having information system security roles and responsibilities.***

SA-4 ACQUISITIONS

- *Control: The organization includes the following requirements and/or specifications, explicitly or by reference, in information system acquisition contracts based on an assessment of risk and in accordance with applicable federal laws, Executive Orders, directives, policies, regulations, and standards:*

…requires software vendors/manufacturers to demonstrate that their software development processes employ state-of-the-practice software and security engineering methods, quality control processes, and validation techniques to minimize flawed or malformed software.

SA-7 USER-INSTALLED SOFTWARE

- Control: The organization enforces explicit rules governing the installation of software by users.
- Supplemental Guidance: If provided the necessary privileges, users have the ability to install software. **The organization identifies what types of software installations are permitted (e.g., updates and security patches to existing software) and what types of installations are prohibited (e.g., software whose pedigree with regard to being potentially malicious is unknown or suspect).**

SA-8 SECURITY ENGINEERING PRINICPLES

- Control: The organization applies information system security engineering principles in the specification, design, development, implementation, and modification of the information system.

- Supplemental Guidance:….Examples of security engineering principles include, for example: (i) developing layered protections; ***(ii) establishing sound security policy, architecture, and controls as the foundation for design; (***iii) incorporating security into the system development life cycle; (iv) delineating physical and logical security boundaries; (v) ***ensuring system developers and integrators are trained on how to develop secure software; (vi***) tailoring security controls to meet organizational and operational needs; and (vii) reducing risk to acceptable levels, thus enabling informed risk management decisions.

SA-11 DEVELOPER SECURITY TESTING

- ***Control: The organization requires that information system developers/integrators, in consultation with associated security personnel (including security engineers):***
  - ***a. Create and implement a security test and evaluation plan;***
  - ***b. Implement a verifiable flaw remediation process to correct weaknesses and deficiencies identified during the security testing and evaluation process; and***
  - ***c. Document the results of the security testing/evaluation and flaw remediation processes.***

- ***Control Enhancements:***

- ***(1) The organization requires that information system developers/integrators employ code analysis tools to examine software for common flaws and document the results of the analysis.***

- ***(2) The organization requires that information system developers/integrators perform a vulnerability analysis to document vulnerabilities, exploitation potential, and risk mitigations.***

- ***(3) The organization requires that information system developers/integrators create a security test and evaluation plan and implement the plan under the witness of an independent verification and validation agent.***

SA-13 TRUSTWORTHINESS

- Control: The organization requires that the information system meets [*Assignment: organization defined level of trustworthiness].*

- Supplemental Guidance: The intent of this control is to ensure that organizations recognize the importance of trustworthiness and making explicit trustworthiness decisions when designing, developing, and implementing organizational information systems. Trustworthiness is a characteristic or property of an information system that expresses the degree to which the system can be expected to preserve the confidentiality, integrity, and availability of the information being processed, stored, or transmitted by the system. Trustworthy information systems are systems that are capable of being trusted to operate within defined levels of *risk despite the environmental* disruptions, human errors, and purposeful attacks that are expected to occur in the specified environments of operation. Two factors affecting the trustworthiness of an information system include: (i) *security functionality (i.e., the security features or functions employed within the* system); and (ii) *security assurance (i.e., the grounds for confidence that the security functionality* is effective in its application)….

SA-13 TRUSTWORTHINESS (Continued)

- ….Appropriate security assurance can be obtained by: (i) the actions taken by developers and implementers of security controls with regard to the design, development, implementation, and operation of those controls; and (ii) the actions taken by assessors to determine the extent to which the controls are implemented correctly, operating as intended, and producing the desired outcome with respect to meeting the security requirements for the information system.

- Developers and implementers can increase the assurance in security controls by employing well defined security policy models, structured, disciplined, and rigorous hardware and software development techniques, and sound system/security engineering principles. Assurance is also based on the assessment of evidence produced during the initiation, acquisition/development, implementation, and operations/maintenance phases of the system development life cycle. ….

- Explicit trustworthiness decisions highlight situations where achieving the information system resilience and security capability necessary to withstand cyber attacks from adversaries with certain threat capabilities may require adjusting the risk management strategy, the design of mission/business processes with regard to automation, the selection and implementation rigor of management and operational protections, or the selection of information technology components with higher levels of trustworthiness.

SC-24 FAIL IN KNOWN STATE

- Control: The information system fails to a [*Assignment: organization-defined known-state] for* [*Assignment: organization-defined types of failures] preserving [Assignment: organization-defined system state information] in failure.*

- Supplemental Guidance: Failure in a known state can address safety or security in accordance with the mission/business needs of the organization. Failure in a known secure state helps prevent a loss of confidentiality, integrity, or availability in the event of a failure of the information system or a component of the system. Failure in a known safe state helps prevent systems from failing to a state that may cause injury to individuals or destruction to property. Preserving information system state information facilitates system restart and return to the operational mode of the organization with less disruption of mission/business processes.

SI-2 FLAW REMEDIATION

- Control: The organization:
  - a. Identifies, reports, and corrects information system flaws;
  - b. Tests software updates related to flaw remediation for effectiveness and potential side effects on organizational information systems before installation; and
  - c. Incorporates flaw remediation into the organizational configuration management process.

- Supplemental Guidance: The organization identifies information systems containing software affected by recently announced software flaws (and potential vulnerabilities resulting from those flaws) and reports this information to designated organizational officials with information security responsibilities (e.g., senior information security officers, information system security managers, information systems security officers). ….Organizations are encouraged to use resources such as the Common Weakness Enumeration (CWE) or Common Vulnerabilities and Exposures (CVE) databases in remediating flaws discovered in organizational information systems. By requiring that flaw remediation be incorporated into the organizational configuration management process, it is the intent of this control that required/anticipated remediation actions are tracked and verified….

SI-3 MALICIOUS CODE PROTECTION

- Control: The organization:
  - a. Employs malicious code protection mechanisms at information system entry and exit points and at workstations, servers, or mobile computing devices on the network to detect and eradicate malicious code:
    - - Transported by electronic mail, electronic mail attachments, web accesses, removable media, or other common means; or
    - - Inserted through the exploitation of information system vulnerabilities;
  - b. Updates malicious code protection mechanisms (including signature definitions) whenever new releases are available in accordance with organizational configuration management policy and procedures;
  - c. Configures malicious code protection mechanisms to:
    - - Perform periodic scans of the information system [*Assignment: organization-defined frequency] and real-time scans of files from external sources as the files are downloaded,* opened, or executed in accordance with organizational security policy; and
    - - [*Selection (one or more): block malicious code; quarantine malicious code; send alert to administrator; [Assignment: organization-defined action]] in response to malicious code* detection; and
  - d. Addresses the receipt of false positives during malicious code detection and eradication and the resulting potential impact on the availability of the information system.

SI-3 MALICIOUS CODE PROTECTION (Continued)

- Supplemental Guidance:

- Information system entry and exit points include, for example, firewalls, electronic mail servers, web servers, proxy servers, and remote-access servers. Malicious code includes, for example, viruses, worms, Trojan horses, and spyware. Malicious code can also be encoded in various formats (e.g., UUENCODE, Unicode) or contained within a compressed file. Removable media includes, for example, USB devices, diskettes, or compact disks. A variety of technologies and methods exist to limit or eliminate the effects of malicious code attacks. Pervasive configuration management and strong software integrity controls may be effective in preventing execution of unauthorized code. In addition to commercial off-the-shelf software, malicious code may also be present in custom-built software. This could include, for example, logic bombs, back doors, and other types of cyber attacks that could affect organizational missions and business functions. Traditional malicious code protection mechanisms are not built to detect such code. In these situations, organizations must rely instead on other risk mitigation measures to include, for example, secure coding practices, trusted procurement processes, configuration management and control, and monitoring practices to help ensure that software does not perform functions other than those intended. Related controls: SA-4, SA-8, SA-12, SA-13, SI-4, SI-7.

SI-3 MALICIOUS CODE PROTECTION (Continued)

- Control Enhancements:
  - **(1) The organization centrally manages malicious code protection mechanisms.**
  - **(2) The information system automatically updates malicious code protection mechanisms (including signature definitions).**
  - **(3) The information system prevents non-privileged users from circumventing malicious code protection capabilities.**
  - **(4) The information system updates malicious code protection mechanisms only when directed by a privileged user.**
  - **(5) The organization does not allow users to introduce removable media into the information system.**
  - **(6) The organization tests malicious code protection mechanisms [*Assignment: organization-defined frequency] by introducing a known benign, non-spreading test case into the information system* and subsequently verifying that both detection of the test case and associated incident reporting occur, as required.**

SI-5 SECURITY ALERTS, ADVISORIES, AND DIRECTIVES

- Control: The organization:
  - a. Receives information system security alerts, advisories, and directives from designated external organizations on an ongoing basis;
  - b. Generates internal security alerts, advisories, and directives as deemed necessary;
  - c. Disseminates security alerts, advisories, and directives to [*Assignment: organization-defined list of personnel (identified by name and/or by role)]; and*
  - d. Implements security directives in accordance with established time frames, or notifies the issuing organization of the degree of noncompliance.

- Supplemental Guidance: Security alerts and advisories are generated by the United States Computer Emergency Readiness Team (US-CERT) to maintain situational awareness across the federal government. Security directives are issued by OMB or other designated organizations with the responsibility and authority to issue such directives. Compliance to security directives is *essential* due to the critical nature of many of these directives and the potential immediate adverse affects on organizational operations and assets, individuals,

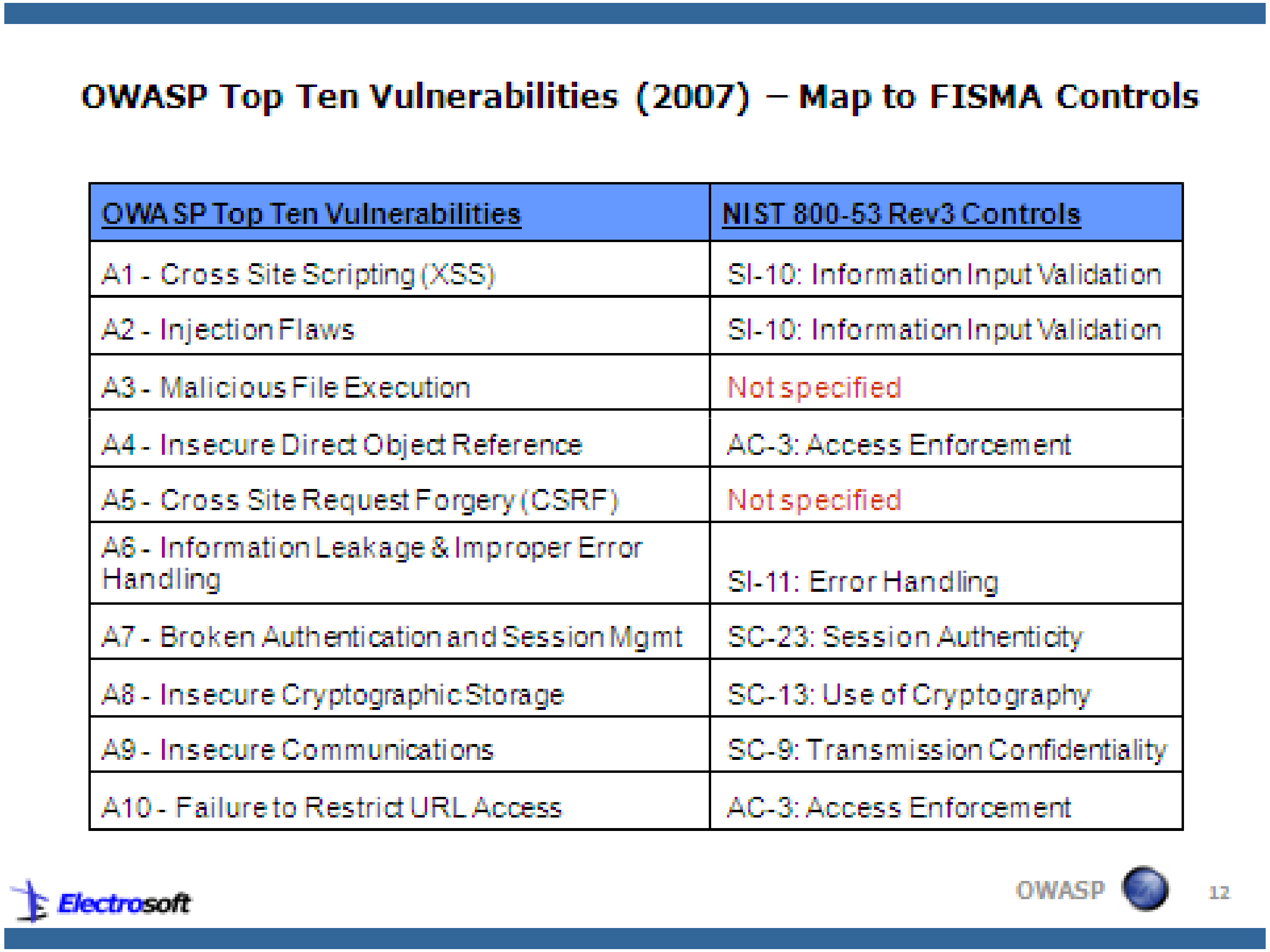
SI-10 INFORMATION INPUT VALIDATION

- Control: The information system checks the validity of information inputs.
- Supplemental Guidance: Rules for checking the valid syntax and semantics of information system inputs (e.g., character set, length, numerical range, acceptable values) are in place to verify that inputs match specified definitions for format and content. Inputs passed to interpreters are prescreened to prevent the content from being unintentionally interpreted as commands.

Opportunity Exists

- There is a foundation for SwA in NIST 800-53 rev 3.
  - How do we incorporate the foundation in our message
  - How do we communicate our resources in the context of controls

OWASP Examples for moving forward (1 of 2)



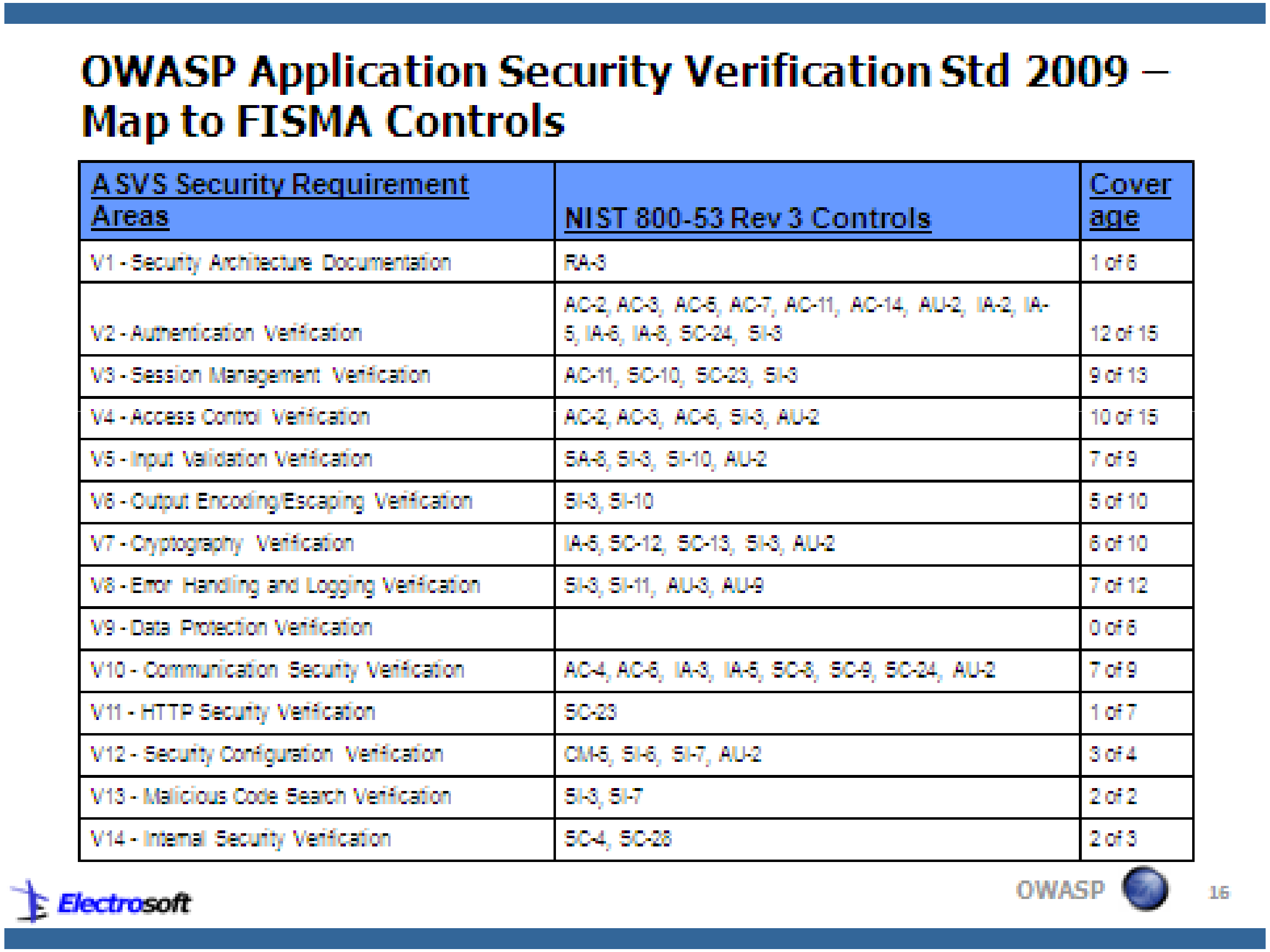
OWASP Top Ten Vulnerabilities (2007) – Map to FISMA Controls

OWASP Top Ten Vulnerabilities	NIST 800-53 Rev3 Controls
A1 - Cross Site Scripting (XSS)	SI-10: Information Input Validation
A2 - Injection Flaws	SI-10: Information Input Validation
A3 - Malicious File Execution	Not specified
A4 - Insecure Direct Object Reference	AC-3: Access Enforcement
A5 - Cross Site Request Forgery (CSRF)	Not specified
A6 - Information Leakage & Improper Error Handling	SI-11: Error Handling
A7 - Broken Authentication and Session Mgmt	SC-23: Session Authenticity
A8 - Insecure Cryptographic Storage	SC-13: Use of Cryptography
A9 - Insecure Communications	SC-9: Transmission Confidentiality
A10 - Failure to Restrict URL Access	AC-3: Access Enforcement

Electrosoft

OWASP 12

OWASP Examples for moving forward (2 of 2)



OWASP Application Security Verification Std 2009 – Map to FISMA Controls

ASVS Security Requirement Areas	NIST 800-53 Rev 3 Controls	Coverage
V1 - Security Architecture Documentation	RA-3	1 of 6
V2 - Authentication Verification	AC-2, AC-3, AC-6, AC-7, AC-11, AC-14, AU-2, IA-2, IA-5, IA-6, IA-8, SC-24, SI-3	12 of 15
V3 - Session Management Verification	AC-11, SC-10, SC-23, SI-3	9 of 13
V4 - Access Control Verification	AC-2, AC-3, AC-6, SI-3, AU-2	10 of 15
V5 - Input Validation Verification	SA-8, SI-3, SI-10, AU-2	7 of 9
V6 - Output Encoding/Escaping Verification	SI-3, SI-10	5 of 10
V7 - Cryptography Verification	IA-5, SC-12, SC-13, SI-3, AU-2	6 of 10
V8 - Error Handling and Logging Verification	SI-3, SI-11, AU-3, AU-9	7 of 12
V9 - Data Protection Verification		0 of 6
V10 - Communication Security Verification	AC-4, AC-6, IA-3, IA-5, SC-8, SC-9, SC-24, AU-2	7 of 9
V11 - HTTP Security Verification	SC-23	1 of 7
V12 - Security Configuration Verification	CM-6, SI-6, SI-7, AU-2	3 of 4
V13 - Malicious Code Search Verification	SI-3, SI-7	2 of 2
V14 - Internal Security Verification	SC-4, SC-28	2 of 3

SwA WG Deliverables (1 of 3)

	P&P	A&O	WET	TTPE /MW	M/BC
DEV Pocket Guide Secure Coding (draft)	UD				
DEV Pocket Guide SwA Business Case & Return on Investment (outline)					UD
Assurance Process Reference Model	P				
Software Security Checklist for Supply Chain Risk Management	P				
DEV Pocket Guide "Key Practices for Mitigating the Most Egregious Exploitable Software Weaknesses"				Rev	
DEV Pocket Guide “Software Security Testing”	P				
DEV Pocket Guide “Requirements and Analysis for Secure Software”	UD				
DEV Pocket Guide “Architecture and Design Considerations for Secure Software”					

Key: P –Publicly Available, UD- Under Development, Rev – Under Revision

SwA WG Deliverables (2 of 3)

	P&P	A&O	WET	TTPE /MW	M/BC
Software Assurance Mobile Instruction (SAMI)]			P		
WET Linked In Wiki			P		
The Software Assurance Curriculum Project			P		
Enhancing the Development Life Cycle to Produce Secure Software, v2.0	P				
Proceedings from the Business Case Workshop and Making the Business Case TN					P
ACQ Guide		P			
ACQ Pocket Guide “Software Supply Chain Risk Management and Due Diligence”		P			
ACQ Pocket Guide “Contract Language For Secure Software”		P			
Assurance For CMMI	Rev				
SwA Common Body of Knowledge			P		

Key: P –Publicly Available, UD- Under Development, Rev – Under Revision

SwA WG Deliverables (3 of 3)

	P&P	A&O	WET	TTPE /MW	M/BC
Lifecycle Pocket Guide “Software Assurance in Education, Training & Certification”			P		
Towards and Organization for Software Security Principles and Guidelines			P		
Understanding Product Characteristics Throughout the SDLC	UD			UD	
Practical Measurement Framework for Software Assurance and Information Security					P
Making Security Measurable CVE, CWE, CAPEC				P	
SAMATE				P	
MAEC					

Key: P –Publicly Available, UD- Under Development, Rev – Under Revision

What other resources do we have?

- OWASP
- SAFECODE
- Microsoft
- Others