RDB received 3/15/05

NUCLEAR ENERGY INSTITUTE

James W. Davis
DIRECTOR, OPERATIONS
NUCLEAR GENERATION DIVISION

12/16/04
69FR75359
(16)

March 14, 2005

Rules and Directives Branch,
Office of Administration
U. S. Nuclear Regulatory Commission
11555 Rockville Pike
Rockville, MD 20555-0001

SUBJECT: Comments on Draft Guide 1130

The following comments on Draft Regulatory Guide 1130, *Criteria for Use of Computers in Safety Systems of Nuclear Power Plants,* are submitted by the Nuclear Energy Institute (NEI)[1] on behalf of the industry.

The detained comments in Enclosure (1) focus on the cyber security aspects of the guide. In general, comments address two broad areas of concern, scope and clarity. A number of organizations have been working on cyber security requirements and it is important that Section C.2 of the guide be consistent with the industry's security program, both in physical security and cyber security.

There is the overarching concern that this regulatory guide is not the right vehicle for detailing cyber security requirements. There is no question that features that facilitate cyber security need to be incorporated in the design and operation of computers in safety applications. However, three years of industry work has shown that this is a complex area, with many ways to provide adequate protection and a high potential for unintended consequences. We are concerned that section C.2 of DG-1130 has not undergone the rigorous process of development and stakeholder interactions needed. It is recommended that section C.2 be removed from this regulatory guide, and appropriate guidance be developed through a process that involves extensive stakeholder interaction or through an established standards setting organization, such as IEEE.

---

[1] NEI is the organization responsible for establishing unified nuclear industry policy on matters affecting the nuclear energy industry, including regulatory aspects of generic operational and technical issues. NEI's Members include all utilities licensed to operate commercial nuclear power plants in the United States, nuclear plant designers, major architect/engineering firms, fuel fabrication facilities, materials licensees, and other organizations and individuals involved in the nuclear energy issue.

SISP Review Complete

E-RIDS = ADM-03
Add - S. Aggarwal (SKA)

1776 I STREET, NW SUITE 400 WASHINGTON, DC 20006-3708 PHONE 202.739.8000 FAX 202.785.4019 www.nei.org

Template - ADM-013

Rules and Directives Branch
March 14, 2005

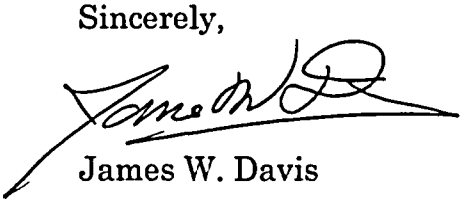
As written, Section C.2, Security, inappropriately establishes broad based cyber security program requirements that go well beyond the design and use of a computer in a safety application. Changes are required to better focus the guide on cyber security and its application to the use of computers in safety applications. The guide should allow use of existing programs, were applicable, but not dictate program requirements for other applications.

In many cases the specific wording does not provide the clarity needed and will result in significant implementation issues. The diversity and changing nature of digital technology provides a unique challenge in achieving clarity of requirements. Prescriptive requirements that may be appropriate today will be inadequate as technology changes. Based on industry experience in developing cyber security program requirements, it is important to clearly identify the intended outcome of a particular requirement. There may be a variety of acceptable methods of achieving that outcome.

We strongly recommend that the NRC staff conduct a public meeting, after Section C.2 has been redrafted. This could significantly aid in achieving the clarity needed in the cyber security requirements of the guide. The meeting should also address whether this guide is the appropriate mechanism for establishing these requirements.

If you have any questions on this issue, please contact me at 202-739-8105 or e-mail at jwd@nei.org.

Sincerely,



James W. Davis

Comments on DG-1130
Criteria for Use of computers in
Safety Systems of Nuclear Power Plants
March 1, 2005

The industry is concerned that the Cyber Security Requirements presented in this Draft Guide have the potential, as written for having significant unintended consequences. It is strongly recommended that prior to issuing the guidance of Section 2 that there be further interactions with interested parties so that the design requirements support, instead of conflicting with, general concepts for managing cyber security at a power reactor site.

General Comments:

1. Currently, a number of industry organizations are developing comprehensive guidelines for the improvement of computer security: NIST, ISA, NEI, etc. Activities related to the protection of computer networks are also coordinated on a federal level. It is not clear how the guide is related to any of these efforts.

2. DG 1130 is based solely upon a life-cycle approach and is too restrictive. Other approaches, including risk-based, are available and should be allowed. Cyber vulnerabilities could be evaluated in relation to the impact to the plant and use of plant specific PRA

3. NUREG 6847 describes a cyber security assessment methodology for evaluating cyber security risks/vulnerabilities. The Reg Guide does not appear to even mention and some of the requirements in Section C conflict with the methodology.

4. The overall scope of "system safety and security" encompasses more than digital safety systems. As more computer-based control systems are implemented, the issue of system safety and security must address the non-safety computer systems in addition to the safety systems. For example, security vulnerabilities in computer-based control systems could lead to initiating events that would challenge the safety systems. This content would be outside the scope of IEEE Std. 7-4.3.2 and should be addressed by another standard, which would then be endorsed by another Regulatory Guide.

5. Security applies to hardware and software with the draft guides focus mainly on software. Security attributes for hardware need to be addressed (physical access control, modems, connectivity to external networks, data-links, open ports, etc.).

6. These guidelines would appear to preclude the use of any off-the-shelf software or software that had been initially developed for another purpose. There should be a process for qualifying this type of software without having to redesign it. The invoking of security on development of the base software is not possible since that

software has already been developed and approved; only the application software could have security. The NRC staff has issued SERs on a variety of software, such as Triconex, Teleperm, and Common Q, which endorses these prepackaged off the shelf systems. It is not possible to retrofit these cyber security guides to those systems, the already issued NRC SERs should be sufficient in these cases. Most appendix B vendors do not have the access controls that would be required by these guidelines. In many cases appropriate security measure verification could be conducted as part of the V and V phase with controls applied to the verified product.

7. The additional security requirements imposed by DG-1130 are very general and do not provide guidance appropriate for a standard. From the general context of these requirements, it is not clear what should be performed, at a minimum, to ensure security, or what types of threats/attacks require protection. For example, section 2.4.2 states "... There should be provisions against the incorporation of hidden functions in the application development software or the system software that could support potential unauthorized access".

8. While the topic of security applies to both hardware and software, the content of the draft guide focuses mainly on software. Security attributes for hardware need to be addressed as well as providing adequate consideration for COTS (Commercial off-the-self) based alternatives. The consideration of hardware aspects would include physical access control, modems, connectivity to external networks, data-links, open ports, etc.

9. Additionally, several sections of the DG refer to the use of testing for assurance that security requirements are met (see section 2.5.2 for example). In general, testing cannot be used to check for viruses, worms, Trojan horses, bomb codes, or back door codes. Scanning for known viruses and worms is possible, for a specific operating system environment. Reviews and design control / configuration management methods must be applied to prevent bomb codes and back door codes.

Section B Discussion

Paragraph 1-5 no comments

IEEE Std 7-4.3.2-2003 does not provide guidance regarding security measures for computer-based system equipment and software systems. Consequently, the NRC has modified this draft regulatory guide to include Regulatory Positions 2.1 – 2.9, which provide specific guidance concerning computer-based (cyber) safety system security.

> **Comment:** The IEEE Standard has gone through a rigorous development process. Since the guide does not address cyber security, it is hard to understand why this Regulatory guide is the right place to add cyber security requirements. How will the NRC staff ensure that the same level of rigor is used in developing the new requirements outlined in Section C of this

document? The issue is not whether cyber security needs to be addressed, but the right vehicle for the NRC to establish cyber security requirements.

Clause 5.9 of IEEE Std 7-4.3.2-2003, "Control of Access," refers to the applicable requirements in IEEE Std 603-1998 and states, "The design shall permit the administrative control of access to safety system equipment. These administrative controls shall be supported by provisions within the safety systems, by provision in the generating station design, or by a combination thereof." For digital computer-based systems, controls of both physical and electronic access to system software and data should be provided to prevent changes by unauthorized personnel. Controls should address access via network connections and via maintenance equipment. Additionally, the design of the plant data communication systems should ensure that the systems do not present an electronic path by which unauthorized personnel can change plant software or display erroneous plant status information to the operators. Annex E to IEEE Std 7-4.3.2-2003 provides useful information for establishing communication independence of plant equipment and systems.

Computer-based systems must be secure from electronic vulnerabilities, as well as from physical vulnerabilities, which have been well addressed. Security of computer-based system software relates to the ability to prevent unauthorized, undesirable, and unsafe intrusions throughout the life cycle of the safety system. Computer-based systems are secure from electronic vulnerabilities if unauthorized access and use of those systems is prevented. The security of computer-based systems is established through (1) designing the security features that will meet user security requirements in the systems, (2) developing the systems without undocumented codes (e.g., back door coding, viruses, worms, Trojan horses, and bomb codes), and (3) installing and maintaining those systems in accordance with the users' security program.

Comment: The addition of the term "as well as for physical vulnerabilities" generates confusion as to the scope of this guidance. There is a physical security plan and requirements for protection of safety systems, digital or otherwise. It appears that the intent is to discuss cyber security issues which can involve both physical access and logical access to the component. This section needs to be rewritten to narrow the focus and remove potential conflicts with other regulatory requirements.

Comment: Expand discussion to include station administrative procedures. Existing station administrative procedures may already provide this requirement and should be recognized as appropriate. **Proposed text:** "The security of computer-based systems is established through (1) designing the security features that will meet user security requirements in the systems, (2) developing the systems without undocumented codes (e.g., back door coding, viruses, worms, Trojan horses, and bomb codes), and (3) installing and maintaining those systems in accordance with the station administrative procedures and the users' security program."

Regulatory Positions 2.1 – 2.9 (presented in Section C of this draft regulatory guide) provide specific guidance concerning safety system security. The effectiveness of the security functions implemented in the software safety system should be confirmed during verification and

validation (V&V) and in the configuration management of the safety system software in each lifecycle phase.

Comment: "The effectiveness of the security functions... should be confirmed during verification and validation and in the configuration management ... in each life cycle phase". In the life cycle used as an example in this NuReg, V&V is not mentioned in every phase. How would you test effectiveness as part of configuration management? The testing is part of the V&V process, site and factory acceptance testing, failure analysis/FEMA/FTA, etc. (in this NuReg case, it is part of 2.6 Installation and Checkout Phase), but it is not part of configuration management.

Section C. Regulatory Position

Comment: The use of the word "user" throughout section C.2 appears to refer to licensee on some occasions. For example, "The user should establish a security quality assurance program and a security configuration management program as part of its security program". On other occasions it sounds more like the end-user of the application (e.g. The users and developers should define the security functional and performance requirements ... user documentation for the software and hardware, installation and acceptance, user operation and execution, and user maintenance.) The document should be clear when referring to end users vs. the overall licensee. The same issue exists with "developer".

Comment: In many sections, only software is mentioned and does not address the associated hardware. The cyber Security issues apply to both hardware and software that make up a given safety related system. As such, it is important to address hardware in various subsections.

Comment: Focus of this Section C should be specific cyber security requirements that should be addressed during the different phases of the digital SR system life cycle. The Cyber Security programmatic aspects should not be part of this Reg Guide since the Reg Guide's objective is specifying technical requirements and not programmatic requirements

2. Security

This regulatory position uses the waterfall lifecycle phases as a framework for describing specific digital safety system security guidance. Lifecycles other than the waterfall lifecycle may be used. The digital safety system development process should address potential security vulnerabilities in each phase of the digital safety system lifecycle. The typical waterfall lifecycle consists of the following phases:

- Concepts
- Requirements

- Design
- Implementation
- Test
- Installation and Checkout
- Operation
- Maintenance
- Retirement

Comment: "Life cycles other than the waterfall may be used". The amplification of the different phases only applies to the Waterfall model. There is no statement to the effect that the utility may address the concepts as appropriate, i.e. not be tied to Waterfall wording that is not appropriate for other models, e.g. incremental model.

The lifecycle phase-specific security requirements should be commensurate with the risk and magnitude of the harm resulting from unauthorized access, use, disclosure, disruption, or destruction of the digital safety system.

The user should establish a security quality assurance program and a security configuration management program as part of its security program. The security quality assurance program and security configuration management program can be incorporated into the existing quality assurance and configuration management programs.

Comment: As written "establish a security quality assurance program" is much too broad a statement. It literally would be requiring a QA program for much more than the digital components covered by this guide. This should be rephrased to require application of quality assurance and configuration management techniques required as part of the general program to the cyber security features applied to the component.

The Quality Assurance organization should conduct periodic audits to determine the effectiveness of the digital safety system security program.

Comment: Here also the focus needs to be shifted to including the cyber security aspects of the component within the quality assurance program required for other aspects of the design and use.

Regulatory positions 2.1 – 2.9 describe digital safety system security activities and recommendations for the individual phases of the waterfall lifecycle.

2.1 Concepts Phase

In the concepts phase, the user and developer should delineate safety system security features that should be implemented to meet the desired system security capabilities. During this activity, the system architecture is selected and the desired safety system security functional capabilities are allocated to hardware, software, and user interface components.

Comment: "In the concepts phase, ... should delineate.. security features ... to meet the desired system security capabilities." This is occurring too early. In the concept phase, the security capabilities are only first being determined. This delineation should be delayed until the requirements phase.

The user and developer should perform security risk analyses to identify potential security vulnerabilities in the relevant phases of the system and software life cycle. The results of the analysis should be used to establish security requirements for the system (hardware and software).

Remote access to the safety system software functions or data from outside the technical environment of the plant (e.g., from the administrative or engineering buildings or from outside the plant) that involves a potential security threat to safety functions should not be implemented.

Comment: "Remote access ...should not be implemented." This is pretty drastic and provides no exceptions. The term "potential security threat" will be hard to interpret. In one sense, it either makes the paragraph a non-statement, in that you would not allow any access that represented a potential threat with unacceptable risk. In another sense it could be read to be a zero risk criteria, conflicting with the preceding paragraph. With appropriate security features, this should be allowed, as it is direct conflict with the way most utilities are heading. Future technology may allow connectivity; requirement should ensure that connectivity is analyzed.

Comment: "Remote access to the safety system software functions or data for outside the technical environment of the plant (e.g., from the administrative or engineering buildings or from outside the plant) that involves a potential security threat to safety functions should not be implemented." In fact, remote access that represents a security threat should not be implemented, unless the risk can be mitigated, no matter where the access exists. There are security features associated with the protected and vital areas that should be credited in the design of a digital system, which is what the paragraph appears to be saying. The definition of "technical environment" in a manner inconsistent with security requirements adds unnecessary complication.

2.2 Requirements Phase

2.2.1 *System Features*

The users and developers should define the security functional and performance requirements; interfaces external to the system; and the requirements for qualification, human factors engineering, data definitions, user documentation for the software and hardware, installation and acceptance, user operation and execution, and user maintenance.

Comment: Add system configuration as another item to be defined by system users and developers. Add "; system configuration" after the words "performance requirements" in the first sentence.

The security requirements are part of the overall system requirements. Therefore, the V&V process of the overall system should ensure the correctness, completeness, accuracy, testability, and consistency of the system software and hardware system requirements, which include security requirements.

Comment: This paragraph includes V&V requirements beyond those related to cyber security and are redundant of those in other parts of the IEEE standard. The intent of this paragraph is to ensure that any security requirements added to the system are evaluated in the V&V phase. However as written it requires a number of V&V steps related to all system software and hardware requirements. Although this testing is warranted, and called for in other parts of the standard, it should not be the objective of a section that relates to security. The intent of this change is not to change the V&V requirements related to security, but get the focus of this paragraph back on to security. **Proposed Text:** The security requirements should be part of the overall system requirements. Therefore, the V&V process should include those security requirements that have an impact on the correctness, completeness, accuracy, testability, and consistency of the system software and hardware system.

Requirements specifying the use of pre-developed software (e.g., reuse software and commercial off-the-shelf software) should minimize the vulnerability of the safety system (e.g., by minimizing the number of pre-developed software functions used by the safety system to the extent necessary or using existing security functions of the pre-developed software).

Comment: "requirements specifying the use of pre-developed software should minimize the vulnerability of the safety system" is good, but "by minimizing the number of pre-developed software functions used by the safety system..." doesn't make sense. We want to re use pre-developed software functions to the MAXIMUM, both for safety and cost considerations. If the pre-developed software functions have already been tested and have operating experience behind them, wouldn't it be prudent to use them.

2.2.2 *Development Activities*

The developer should delineate its security policies to ensure the developed products (hardware and software) do not contain undocumented code (e.g., back door coding), malicious code (e.g., intrusions, viruses, worms, Trojan horses, or bomb codes), and other unwanted and undocumented functions or applications.

Comment: "... should delineate its security policies to ensure the developed products do not contain undocumented code " How can a policy do this alone?

A policy will not prevent this; it can only minimize the chances. It takes it and a combination of other activities to "ensure". It is unclear who the developer is meant to be in this case. It really seems that what is needed is a development process that ensures,....

2.3 Design Phase

2.3.1 *System Features*

The safety system software security requirements identified in the system requirements specification should be translated into specific design configuration items in the software design description. The safety system software security design configuration items should address control over (1) access to the software functions, (2) use of safety system services, (3) data communication with other systems, and (4) the list of personnel who may access and use the system.

Comment: There concerns of clarity of intent in some of this guide. What is the linkage between the list of four items in paragraph one and "access control" discussed in paragraph three? A very important concept is contained in, "Access control" methods "should be based on the results of risk analysis." Access control usually is considered to have two components, physical access and logical access. Does it apply to all elements in paragraph one? Use of the term without clearly defining the scope of the statement is going to lead to problems. Additionally, item four in the list, "(4) the list of personnel who may access and use the system..." seems to be unnecessary. Access is discussed in item one and use in item two. A list may not be necessary in all cases. For example, one logical access path may be from a control board in the control room. Maintaining a list of qualified ROs and SROs would not improve security.

Design configuration items incorporating pre-developed software into the safety system should be specified such that vulnerability of the safety system security is minimized.

Comment: What is intended is unclear. "Design configuration items incorporating pre-developed software ... should be specified ...such that vulnerability... is minimized". How could that minimize security vulnerabilities?

Comment: Add a discussion concerning hardware and split the section into part "a" and "b" for clarity. Add "a." before the second paragraph. Add the following new paragraph at the end of the section:

> b. The safety system hardware design should consider system architecture that includes external connectivity, user interface, maintenance interface, development systems and interfaces,

networking architectures (if applicable), built-in communication devices, data-link requirements, data communications requirements, etc.

Access control should be based on the results of risk analyses. The results of the analyses may require more complex access control, such as a combination of knowledge (e.g., password), property (e.g., key, smart-card) and personal features (e.g., fingerprints), rather than just a password.

Comment: Clarify access to explicitly include both physical and logical access. Proposed text: Add "Physical and logical" at the beginning of the sentence.

2.3.2 *Development Activities*

The developer should delineate the standards and procedures that will conform with the applicable security policies to ensure the system design products (hardware and software) do not contain undocumented code (e.g., back door coding), malicious code (e.g., intrusions, viruses, worms, Trojan horses, or bomb codes), and other unwanted or undocumented functions or applications.

2.4 Implementation Phase

In the software implementation phase, the system design is transformed into code, database structures, and related machine executable representations. The implementation activity addresses software coding and testing, including the incorporation of reused software products.

Comment: Expand the section to include integrated hardware and software implementation as well as hardware and communication configuration. Replace paragraph with: "In the system (integrated hardware and software) implementation phase, the system design is transformed into code, database structures, and related machine executable representations. The implementation activity addresses hardware configuration and set-up, software coding and testing, communication configuration and set-up including the incorporation of reused software products."

Comment: Organize the subsection in a clearer manner. Proposed text: Start Subsection 2.4.1, System Features, after the first paragraph to include only one paragraph (The developer should ensure that the security design configuration item transformations from the system design specification are correct, accurate, and complete.). Start Subsection 2.4.2., Development Activities, immediately after that

2.4.1 *System Features* The developer should ensure that the security design configuration item transformations from the system design specification are correct, accurate, and complete.

2.4.2 *Development Activities*

The developer should implement security procedures and standards to ensure against tampering with the developed software. The developer's standards and procedures should include testing, including scanning, to ensure against undocumented codes or malicious codes that might (1) allow unauthorized access or use of the system or (2) cause systems to behave beyond the system requirements. There should be provisions against the incorporation of hidden functions in the application development software or the system software that could support potential unauthorized access. If provisions cannot be implemented for pre-developed software, the use of such software should be justified considering potential security threats.

> **Comment:** "...to ensure against tampering ..." You cannot always develop procedures and standards to ensure against tampering. You can minimize the potential and/or write/implement the program to flag if tampering does occur.
>
> **Comment:** Scanning is not always meaningful. Proposed text: Add the words, "where appropriate" after the word "scanning" in the second sentence.
>
> **Comment:** Expand the thought to require that all hidden functions be addressed by a failure modes and affects analysis. Proposed text: Replace the sentence with the following: "The developer should account for any and all hidden functions embedded in the code, its purpose and impact on the client system. If possible, these functions should be disabled or removed, or as a minimum, they need to be addressed as part of the failure modes and affects analysis of the application code to prevent any unauthorized access."

The user and developer should review the possibility for deliberate modification of software to cause erroneous behavior of the software triggered by certain time or data constraints (e.g., viruses, worms, and Trojan horses).

> **Comment:** 2nd sentence "should include testing, including scanning" Scanning is very dependent on the platform and code being used, and may not be available for the specified code and complier. This is likely to be a difficult task with little assurance that the results will be comprehensive and successful in uncovering hidden problems given the size and complexity of most modern computer systems. Pure application code scanning may be partially successful, but many operating systems, machine code, and callable library function aspects of the system may not be able to be successfully scanned and are just as likely to be where avenues for exploitation exist.
>
> **Comment:** Section 2.4.2 "System Software" – This is likely to be proprietary and generally unavailable. It is likely that there is no reliable method to determine this for Operating System Software (i.e., Microsoft and other operating system suppliers do not provide access to the source code for operating systems and callable code libraries). In such cases, unless such software is modified by the application developer, the security effort should be limited to ensuring that the features within the software do not compromise the security requirements of the system.

2.5 Test Phase

The objective of testing software security functions is to ensure that the software security requirements and system security requirements allocated to software are validated by execution of integration, system, and acceptance tests. Testing includes software testing, software integration testing, software qualification testing, system integration testing, and system qualification testing.

> **Comment:** Need to address System Level Testing for integrated hardware and software and then lead into specifics on software testing. This model is ignoring the hardware configuration aspects of security which are going to be the major reasons for intrusion stemming from modems, open ports, unknown and unanalyzed network connectivity to IT LAN, etc. Test Phase introduction needs to address hardware, software and the system as a whole, including any external built-in features. **Proposed text:** Add a thought similar to the paragraph below at the beginning of the section:
>
> > Need to address System Level Testing for integrated hardware and software and then lead into specifics on software testing. This model is ignoring the hardware configuration aspects of security which are going to be the major reasons for intrusion stemming from modems, open ports, unknown and unanalyzed network connectivity to IT LAN, etc.
>
> **Comment:** Alternative wording--The objective of testing software security functions is to ensure that the software security requirements and system security requirements allocated to software are validated by execution of integration, system, and acceptance tests where practical and necessary. Testing includes system hardware configuration including all external connectivity, software testing, software integration testing, software qualification testing, system integration testing, and system qualification testing, and system Factory Acceptance Testing

2.5.1 *System Features*

The security requirements and configuration items are part of the overall system requirements and design configuration items. Therefore, testing security design configuration items is just one element of the overall system testing. The user and developer should test each system security feature to verify that the implemented system does not increase the risk of security vulnerabilities.

2.5.2 *Development Activities*

The developer should perform testing and scanning to ensure the developed products (i.e., hardware and software) do not contain undocumented code (e.g., back door coding), malicious code (e.g., intrusions, viruses, worms, Trojan horses, or bomb codes), and other unwanted and

undocumented functions or applications. Additionally, the developer should audit the configuration management processes to ensure that the software is developed in accordance with the appropriate configuration management procedures and standards.

Comment: Based on experience, about 99% of this phase for security aspects of the system consists of checking that the designed security features are correctly configured and enabled (i.e., the security design elements have been put in place). The testing of specific security code/features is likely to be unfeasible for many if not most of the security items/functions. For instance, setting up the adverse conditions to perform testing might require a hacker's software toolbox" and expert hacker's knowledge to produce the environment necessary to perform the test. Example test requirement: "Verify that the anti-virus software detects and eliminates viruses". Testing for this type of requirement may be undesirable as the testing itself could expose and potentially "infect" or alter the system.

Comment: Requirements should verify that proper anti-virus software has been installed

Comment: Add testing for potential compromise of system integrity. Proposed text: Add the following paragraph at the end: "The developer should perform testing to ensure that the system hardware architecture and external communication devices and configurations are such that they do not provide unauthorized unknown pathways and compromise system integrity. Attention needs to focus on built-in OEM features"

2.6 Installation and Checkout Phase

In installation and checkout, the safety system is installed and tested in the target environment. The system user should perform an acceptance review and test the safety system security features. The objective of installation and checkout security testing is to verify and validate the correctness of the safety system security features in the target environment.

Comment: Change title to " Site Installation, Checkout and Acceptance Testing" Phase

Comment: Clarify that the review and test includes physical and logical features. Add the words "physical and logical" between the words "safety" and "security" in the second sentence.

2.6.1 *System Features*

The user should ensure that the system features enable the user to perform post installation testing of the system to verify and validate that the security requirements have been incorporated into the system appropriately.

2.6.2 *Development Activities*

A user or licensee should have a comprehensive digital system security program. The security policies, standards, and procedures should ensure that installation of the digital system will not compromise the security of the digital system, other systems, or the plant. This may require the user to perform a security assessment, which includes a risk assessment, to identify the potential security vulnerabilities caused by installation the digital system. The risk assessment should include an evaluation of new security constraints in the system; an assessment of the proposed system changes and their impact on system security; and an evaluation of operating procedures for correctness and usability. The results of this assessment should provide a technical basis for establishing certain security levels for the systems and the plant.

2.7 Operation Phase

The operation lifecycle process involves the use of the safety system by the end user in its intended operational environment.

> Comment:: The purpose of IEEE std and Reg Guide is to provide technical requirements and not programmatic requirements. The programmatic requirements should not be specified in a technical document. Proposed text: The operation lifecycle process involves the use of the safety system by the end user in its intended operational environment. Delete 2nd paragraph and replace it with the following: During operations phase, the user should ensure that the system security is in tact by techniques such as periodic monitoring, review of system logs, real time monitoring where possible. Leave 3rd Para "as is".

The user should monitor and record access and use of the system to ensure that its digital system security policies are implemented properly. The monitoring should include real-time monitoring and periodic audits. The type of monitoring is determined by the risk analyses performed in earlier lifecycle phases. The audit should include the security of any equipment that is connected to the system for maintenance.

> Comment: "The user should monitor and record .."Monitor yes, record no. In nuclear terminology, there are many components within the plant subject to this NuReg. Not all have the capability to record access attempts, e.g. Bailey logic cards, digital valve position indicators, etc. and even if they did, can you imagine the extra work to go around and collect/check these records?

The user should evaluate the impact of safety system changes in the operating environment on safety system security; assess the effect on safety system security of any proposed changes; evaluate operating procedures for compliance with the intended use; and analyze security risks affecting the user and the system. The user should evaluate new security constraints in the system; assess proposed system changes and their impact on system security; and evaluate operating procedures for correctness and usability.

2.8 Maintenance Phase

The maintenance phase is activated when the user changes the system or associated documentation. These changes may be categorized as follows:

- Modifications (i.e., corrective, adaptive, or perfective changes)
- Migration (i.e., the movement of software to a new operational environment)
- Retirement (i.e., the withdrawal of active support by the operation and maintenance organization, partial or total replacement by a new system, or installation of an upgraded system)

System modifications may be derived from requirements specified to correct errors (corrective), to adapt to a changed operating environment (adaptive), or to respond to additional user requests or enhancements (perfective).

2.8.1 *Maintenance Activities*

Modifications of the safety system should be treated as development processes and should be verified and validated as described above. Security functions should be assessed as described in the above regulatory positions, and should be revised (as appropriate) to reflect requirements derived from the maintenance process.

When migrating software, the user should verify that the migrated software meets the safety system security requirements. The maintenance process should continue to conform to existing safety system security requirements unless those requirements are to be changed as part of the maintenance activity.

2.8.2 *Quality Assurance*

If the safety system security functions were not previously verified and validated using a level of effort commensurate with the safety system security functional requirements, and appropriate documentation is not available or adequate, the user should determine whether the missing or incomplete documentation should be generated. In making this determination of whether to generate missing documentation, the minimum safety system security functional requirements should be taken into consideration.

The user should establish a security configuration management program as part of its security program. The security configuration program may be incorporated into the existing configuration management program.

> **Comment:** It is unclear what differentiates a "security configuration management program" from a typical configuration management program. Then there is a conflict between the first and second sentences. In the first sentence it's to be part of the security program. In the second sentence, it may be incorporated into the existing program. This section should be changed indicate that cyber security features should be maintained under a configuration management program.

> **Comment:** Purpose of IEEE std and Reg Guide is to provide technical requirements and not programmatic requirements. The programmatic requirements should not be specified in a technical document. Programmatic

Requirements need to be addressed in other regulatory standard or in an updated 10CFR 50, Appendix B QA Requirements. Technical Std needs to address technical Requirements only. Delete Sections C.2.8.2; C.2.8.3; C.2.8.4

2.8.3 *Incident Response*

The user should develop an incident response and recovery plan for responding to digital system security incidents(e.g., intrusions, viruses, worms, Trojan horses, or bomb codes). The plan should be developed to address various loss scenarios and undesirable operations of plant digital systems, including possible interruptions in service due to the loss of system resources, data, facility, staff, and/or infrastructure. The plan should define contingencies for ensuring minimal disruption to critical services in these instances.

2.8.4 *Audits and Assessments*

The user should perform periodic computer system security self-assessments and audits, which are key components of a good security program. The user should assess proposed safety system changes and their impact on safety system security; evaluate anomalies that are discovered during operation; assess migration requirements; and re-perform V&V tasks to ensure that vulnerabilities have not been introduced into the plant environment.

Comment: Change to reperform selected V&V tasks. It is not clear that all V&V tasks would need to be conducted at this point.

2.9 Retirement Phase

In the retirement lifecycle phase, the user should assess the effect of replacing or removing the existing safety system security functions from the operating environment. The user should include in the scope of this assessment the effect on safety and nonsafety system interfaces of removing the system security functions. The user should document the methods by which a change in the safety system security functions will be mitigated (e.g., replacement of the security functions, isolation from other safety systems and user interactions, or retirement of the safety system interfacing functions).

Comment: The "Retirement Phase" does not address the concern of cleansing the hardware once it is removed from service and sold to an external entity. The document should include guidelines for data cleaning. Also, one step re-formatting should NOT be sufficient. Disk destruction or complete overwrite should be required before