Global Information Assurance Certification Paper

Determining System Boundaries for Federal IT Systems

Submitted by Brian Sizemore

GIAC Security Essentials Certification
Practical Assignment

Version 1.4b
Option 1
May 3, 2003

Abstract

The following document describes how to determine the system boundaries for Federal information technology systems. These boundaries determine the scope of responsibility for all aspects of system security, from the implementation of technical controls to testing requirements for a system's certification. Business continuity strategies, documentation of system security controls, and interface requirements all depend upon the size and complexity of the system. The only way to determine what should be included in any of these security tasks is to first establish and document the system boundaries.

Fully establishing a system's boundaries requires four steps: defining the system type and security requirements, establishing the physical boundaries, determining the logical boundaries, and lastly, documenting the system interconnections and rationales behind those interconnections. Once completed, system management will own a definitive scope of the system, one that fully defines both what is included and excluded from the system, as well as justifies why components were set external to the system's responsibilities. The system boundaries provide the solid foundation for all security activities for the system.

Purpose

Defining appropriate system boundaries is one of the most basic, yet vital steps of securing information technology (IT) systems. During this step, the scope of the security task is established, determining at a fundamental level where the responsibility starts and stops for a given system. Defining and establishing these boundaries is the necessary initial step for all matters of system security, whether it be for the creation of contingency plans, the placement and testing of security controls, the conducting of the system's risk assessment, or the eventual certification and accreditation of the system itself.

The following document provides a step-by-step process for establishing the system boundaries for a Federal IT system. Commercial IT systems may also use this methodology, but many of the source publications for this process come from Federal agencies, and the Federal standards and requirements cited may not apply for a commercial system.

Impacts on Certification and Accreditation

An area impacted most significantly by the setting of system boundaries is the certification and accreditation (C&A) of a system. All Federal Major Applications and General Support Systems (see "System Type and Security Requirements" for definitions) must undergo system certification and accreditation before they may begin system operation, and must additionally be recertified and reaccredited every three years. Because of its influence on the complexity and

cost of a system's C&A, some further explanation of the certification and accreditation process and the specific impact of the system's boundaries follows.

To begin with, one needs definitions for certification and accreditation. Certification is defined as "the comprehensive evaluation of the technical and non-technical security controls of an IT system to support the accreditation process that established the extent to which a particular design and implementation meets a set of specified security requirements."[1] As part of a system's development life cycle, the development team implements, documents, and tests the system's security controls. Certification is the process of independently validating that these controls conform to the required Federal and Departmental standards.

Accreditation is "the authorization of an IT system to process, store, or transmit information, granted by a management official."[2] The management official that authorizes system operation is known as the Designated Approving Authority (DAA). This authorization is as much about the DAA officially granting the system approval to operate, however, as it is the DAA officially assuming responsibility for the system. The DAA must understand and assume ownership of any remaining risk inherent to the system when granting the approval to operate.

Organizations must perform certification and accreditation for multiple reasons, both practical and regulatory, despite it being a lengthy and often costly process:

- **Risk management:** The risk assessment and certification process documents any unmitigated risks to the system. By presenting these potential risks to management, system owners make informed decisions whether to mitigate or accept these risks.
- **Independent verification of security controls:** The certification process uses an external review group to examine the system's security controls. The certifier's independence negates any prejudices towards the project, and provides an unbiased examination and recommendation regarding the system's overall security.
- **Security management program:** System owners must know the overall level of security in systems for which they are responsible. In order to make qualified decisions regarding the system's capabilities and controls, as well as remain confidant that the system operates in a safe and secure manner, the system owner needs to understand the security controls in place and the risks still remaining.
- **Federal requirements:** OMB Circular A-130, Appendix III requires Federal agencies plan for security, verify that appropriate officials are

[1] NIST 800-37, *Guidelines for the Security Certification and Accreditation of Federal Information Technology Systems*, p.2
[2] NIST 800-37, p.1

assigned security responsibility, and authorize system processing prior to operation and periodically thereafter.

The very first step in the C&A process is setting the boundaries of the system to be certified and accredited. Without these boundaries, it is not possible for the DAA to know for what he or she is assuming responsibility. In addition, the very scope of the boundaries will have great impact on the C&A process. While expanded boundaries may provide an increased level of authority over a greater portion of the business process, this authority also increases the cost of testing and implementing security controls. Expanded boundaries may even decrease the level of detail of overall testing or protection for the system due to lack of sufficient funding to cover the entire system. Conversely, tighter boundaries allow for more detailed testing of the included aspects of the system, but mean the system owner has control over less of the business process. System management must keep these tradeoffs in mind when deciding upon the system boundaries.

Methodology

Having established the importance of setting boundaries, it is now time to discuss the actual methodology of creating them. In Debra Herrmann's *A Practical Guide to Security Engineering and Information Awareness*, she describes setting an IT system's boundaries as, "comprised of four activities:

1. Determining what is being protected and why
2. Identifying the system
3. Characterizing system operation
4. Ascertaining what one does and does not have control over"[3]

Each of these steps follows a natural progression, and should take place as early in the system's development life cycle as possible. Researchers at MIT and Stanford determined it costs up to seven times as much to implement security features after deployment as compared to during system design.[4] By setting the boundaries early in a system's life cycle, system developers know what security controls they have responsibility for installing into the system, and which they can expect interfacing systems to control. By knowing how a system should be protected, and who exactly is responsible for those protective measures, a development team can significantly reduce the costs of securing a system while at the same time producing a system possessing effective security controls from its earliest stages.

While Debra Herrmann provides an effective description of how to create effective system boundaries in the commercial world, many Federal publications provide guidance which would bolster her methodology. The National Institute of

[3] Herrmann, Debra. *A Practical Guide to Security Engineering and Information Awareness*, p.67
[4] Berinato, Scott. "Finally, A Real Return on Security Spending." CIO Magazine, 2/15/2002

Standards and Technologies (NIST) is responsible for providing all manners of guidance for securing unclassified Federal IT resources. While several of the NIST publications provide a general description of system boundaries, however, no assistance is provided for how to actually establish those boundaries using the Federal definitions and standards in place. This document will merge the commercial guidance with the Federal resources, providing an effective tool in establishing a system's boundary in a Federal government environment. The four activities below closely parallel those described by Ms. Hermann:

1. Define system type and security requirements
2. Establish physical boundaries
3. Determine logical boundaries
4. Document system interconnections and rationales

By completing all these activities, system management can create a defensible illustration of the scope of a system's responsibility.

System Type and Security Requirements

The first step to defining system boundaries is defining the system itself. NIST 800-12, *An Introduction to Computer Security: The NIST Handbook*, defines a system as, "the entire collection of processes, both those performed manually and those using a computer (e.g., manual data collection and subsequent computer manipulation), which performs a function. This includes both application systems and support systems, such as a network."[5] When establishing a system's boundaries, one must first make sure that the system's core function or process has been defined. Without this anchor, there is nothing to prevent boundaries to waver, nor anything to assist in justifying a subcomponents inclusion or exclusion.

An additional categorization for Federal systems is whether a system is a Major Application (MA), a General Support System (GSS), or an application receiving security support from another MA or GSS. An MA is defined as, "an application that requires special attention to security due to the risk and magnitude of the harm resulting from the loss, misuse, or unauthorized access to or modification of the information in the application."[6] A GSS is defined as "interconnected information resources under the same direct management control which shares common functionality… and provides support for a variety of users and/or applications."[7] Examples of a GSS include LANs, communications networks, data centers or shared application integration tools. A system classified as an MA or GSS requires additional security controls and oversight. By making this classification, system owners have a better understanding of the type of security requirements the system must fulfill.

[5] NIST 800-12, *An Introduction to Computer Security: The NIST Handbook*, p. 29
[6] Appendix III to OMB Circular A-130, Section A.2.d
[7] Appendix III to OMB Circular A-130, Section A.2.c

Having defined the system, one must also determine why and how the system needs protection. Not all systems share the same sensitivity or criticality risks. While a financial database may require heavy protection against breaches in confidentiality or integrity, it may only provide updates on a monthly basis. Meanwhile, a website at the IRS may have no confidential data whatsoever, but absolutely requires 24/7 access during tax season. System developers must determine these requirements before purchasing security solutions; else money may be misspent on lower priority solutions.

As a method for determining the sensitivity and criticality requirements for a system, a sensitivity/criticality assessment should take place. During this assessment, system developers, designers, and owners should determine and rate the overall sensitivity and criticality of the system.

Sensitivity can be thought of as the magnitude of impact to the value of the data held within a system after an incident. Per NIST standards, a system's sensitivity is defined as a function of three areas: confidentiality, integrity, and availability. NIST 800-26, *Security Self Assessment Guide for Information Technology Systems*, defines these areas as follows:

> "**Confidentiality**—The information requires protection from unauthorized disclosure.
>
> **Integrity**—The information must be protected from unauthorized, unanticipated, or unintentional modification. This includes, but is not limited to:
>
> - Authenticity—A third party must be able to verify that the content of a message has not been changed in transit.
> - Non-repudiation—The origin or the receipt of a specific message must be verifiable by a third party.
> - Accountability—A security goal that generates the requirement for actions of an entity to be traced uniquely to that entity.
>
> **Availability**—The information technology resource (system or data) must be available on a timely basis to meet mission requirements or to avoid substantial losses. Availability also includes ensuring that resources are used only for intended purposes."[8]

The sensitivity rating uses a qualitative ranking (High/Medium/Low). OMB Bulletin 90-08 provides the following definitions for how to determine the appropriate rating for the system:

[8] NIST 800-26, *Security Self Assessment Guide for Information Technology Systems*, p 5

> **"High**—a critical concern of the system
>
> **Medium**—an important concern, but not necessarily paramount in the organization's priorities.
>
> **Low**—some minimal level of security is required, but not to the same degree as the previous two categories."[9]

While sensitivity defines the value of the system's data, criticality refers to what happens if system processing were interrupted or subject to fraud or abuse. Criticality should also be assigned one of three values:

> **Mission Critical**—Systems whose failure would prevent the Department or Agency from accomplishing its core missions or objectives. Mission Critical systems are also usually included in the Department's Critical Infrastructure Protection (CIP) Plan.
>
> **Mission Important**—Systems whose failure would seriously impede the accomplishment of core missions or objectives in the short term, and may prevent their accomplishment in the long term.
>
> **Mission Supportive**—Systems whose failure would impact Departmental efficiency or effectiveness in accomplishing their core missions or objectives, but would not necessarily prevent them.

Establishing the sensitivity and criticality ratings of the system creates a foundation as to what kind of protection the system requires. Higher confidentiality ratings mean stronger encryption requirements, and better authentication and authorization controls. Higher integrity ratings equate to strong data validation techniques and controls to mitigate user entry errors. Higher availability ratings translate to more stringent service level agreements with support systems and facilities, expanded contingency planning, and stronger redundancy built into the system architecture. As each of these controls can increase security costs, it is vital to determine the requirements as early as possible in the development life cycle. As mentioned previously, building security controls into development is always less expensive than adding these same controls onto a completed system.

[9] OMB Bulletin No. 90-08, Section II.B

Physical Boundaries

After establishing the role of the system and its fundamental security requirements, the next step is determining the actual components that make up the system. These components should all relate or support the system's core business process. Components may consist of hardware, software, environmental support, personnel, or any other supporting function relating to the IT process.

In order to determine what should be included within a system's boundaries, NIST has provided some basic guidelines. NIST 800-18, *Guide for Developing Security Plans for Information Technology Systems*, provides the following assistance:

> "A system, as defined by this guideline, is identified by constructing logical boundaries around a set of processes, communications, storage, and related resources. The elements within these boundaries constitute a single system requiring a security plan. Each element of the system must:
>
> - Be under the **same** direct management control;
> - Have the **same** function or mission objective;
> - Have essentially the **same** operating characteristics and security needs; and
> - Reside in the **same** general operating environment."[10]

NIST 800-37, *Guidelines for the Security Certification and Accreditation of Federal Information Technology Systems*, offers another rule of thumb: "The Designated Approving Authority (DAA) typically has budgetary and operational control over the system being certified and accredited."[11] With this guidance in mind, one can begin segregating what should and should not be set within the boundaries.

One item to note is that the guidance from both publications revolves around control. The system owners must have the capability to effect changes in all areas that fall within the boundaries of their systems. The DAA must have the authority to accept the risks involved with approving a system for operation, or, likewise, the authority to decide that the risks are too great and deny the system from operational status. If no authority exists over a certain component of the system, that component should be excluded from the system's boundaries and be added to the boundaries of another, more appropriate, system.

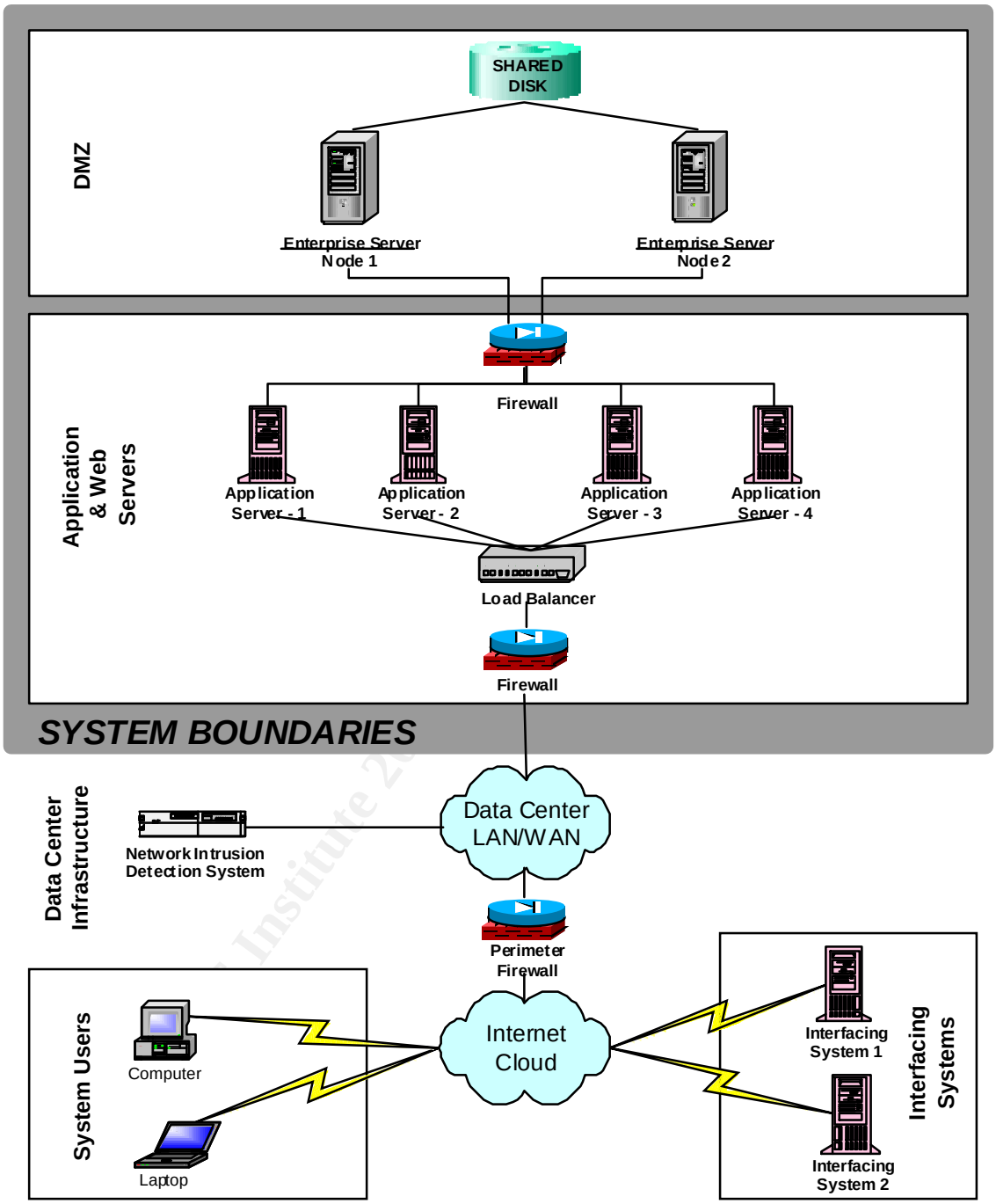
[10] NIST 800-18, *Guide for Developing Security Plans for Information Technology Systems*, p.5
[11] NIST 800-37, *Guidelines for the Security Certification and Accreditation of Federal Information Technology Systems*, p.37

The physical boundaries describe the actual physical location and arrangement of the system's assets. Once completed, system management will know what physical components are considered as part of the system. Once again, this step should be completed as early in the development life cycle as possible. At the initial stages of a system's life cycle, an exact system architecture may not yet be fully developed. In this case, a list of assumptions should be created, with each assumption revisited once the architecture has been completed.

Typically, physical boundaries for an MA will include all system software, and any system hardware designated solely for that system's use, to include such items as application and database servers, routers, switches, load balancers, firewalls, host-based intrusion detection systems (IDS) and data storage units. Any shared hardware should have a prearranged delegation of responsibility amongst the systems sharing the equipment, with one system or even the GSS holding ultimate responsibility for the security of the hardware. A data center GSS would typically include a data center's infrastructure components (e.g. perimeter firewalls, routers, switches, network-based IDS), as well as the data center support systems (e.g. UPS, fire suppression systems, physical security controls), although some data centers may choose to split up responsibility to several interfacing GSSs.

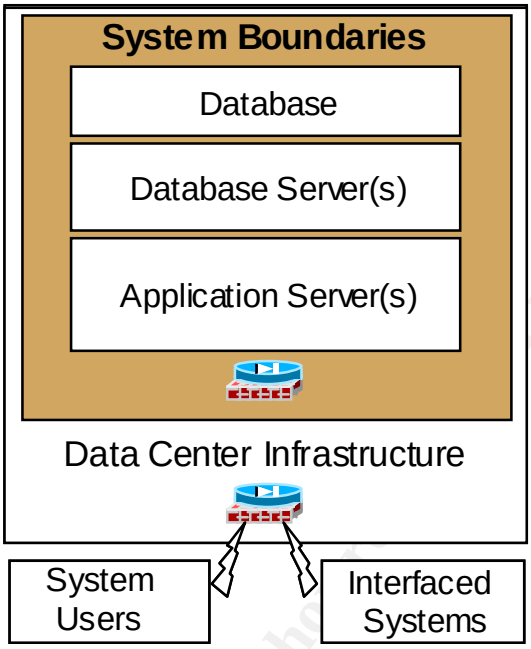
An important element in establishing physical boundaries is creating a system architecture diagram clearly labeling what is included vs. excluded from the system. Figure 1 below depicts an example system architecture with the system boundary demarcation included. In this example, the system boundaries begin at the firewall separating the application and database servers from its hosting data center. The data center's network, including its intrusion detection system and perimeter firewalls, are outside the system's responsibility, as well as the system's users and various interfacing systems.

Figure 1: Sample Physical Boundaries



Again, if an exact architecture cannot be created due to the stage in the system's life cycle, appropriate assumption should be made and documented, and a notional diagram created in lieu of an exact architecture. Figure 2 below shows a notional diagram of the same example system that still illustrates where the system's boundaries lie.

Figure 2: Sample Notional Physical Boundaries



Logical Boundaries

Whereas physical boundaries demonstrate who holds responsibility for physical assets of the system, logical boundaries illustrate who holds responsibility for the system's data. A network cable connecting two system's servers provides an example between the two. Physical boundaries concentrate on who is responsible for the protection of the actual physical cable. Logical boundaries focus instead on where transfer of responsibility occurs for protecting the data transmitted in that cable. Because of the focus on the transferring of responsibility, logical boundaries concentrate on the system's interconnections and interfaces.

The principal interconnections are those between a system and one of the two following groups of connectors:

- System users remotely accessing a system (e.g. via a web application on the users' personnel computers)
- Interfacing systems under the responsibility of different system owners.

Each group should be explored in full, discussing the protective controls in place for each interconnection, as well as where exactly responsibility for the data is transferred from the system to the external agent. Different points of discussion include (when applicable): the use of encryption (including ownership

responsibilities for the encryption tool), training, user support, and application integration tools.

The system owner should also formally document this transfer of responsibility for data. For system users, the most common tool used is a system's Rules of Behavior. The Rules of Behavior formally sets what security activities are expected from a user in order to access the system. Users should sign an agreement stating that they have read and understand the Rules before they receive system access. For interfacing systems, "OMB Circular A-130, Appendix III, requires agencies to obtain written management authorization before connecting their IT systems to other systems, based on an acceptable level of risk."[12] These written authorizations may be in the form of a Memorandum of Understanding (MOU), Service Level Agreement (SLA), or any similar document that describes the security controls protecting the interface. By creating these documents, no doubts remain as to who owns data at any point.

System Interconnections and Rationales

The last step in establishing system boundaries is documenting all the elements that have been excluded from the system boundaries and providing the rationale for their exclusion. This essentially is a list of all the system interconnections, including applications that interface with a system, and support systems providing utilities. By documenting why elements were placed external to the system boundaries, system managers can refer to this list when asked to justify why something should not be within a system's scope of responsibility. This is particularly necessary after a security incident or during disaster recovery – system owners will need to know not only what they are or are not responsible for, but also who is responsible for the interconnected systems.

The following table provides an example of the information desired:

Table 1: System Interconnections and Rationales

Reference	Full Name	POC	POC Phone	Rationale	Sunset
FAS	Federal Accounting Software	Jack Pratt	(202) 555-3321	FAS is owned by the Treasury Dept.	Maintain interface
SRD	Small Retiring Database	Bob Hope	(202) 555-9999	SRD is maintained by external owners	BND to replace SRD on 10/1/05

[12] NIST 800-47, *Security Guide for Interconnecting Information Technology Systems*, p.2-2

Reference	Full Name	POC	POC Phone	Rationale	Sunset
Utilities	All system utilities, including electrical, HVAC, water	Jane Doe	(202) 555-1212	Data center is responsible for all system utilities	N/A

System Security Officers should review and update this list, as well as all other system boundaries documentation, at least once a year. By keeping the system boundaries up to date, system management can maintain a solid foundation for the rest of the system's security controls.

Conclusion

Creating a thorough and conclusive set of system boundaries can be a daunting task, and one that many system owners would rather gloss over or skip entirely. A good set of system boundaries, however, provides the basis for all future system security activities. Without these boundaries, it is extremely difficult to know what to include in an activity, and impossible to justify security decisions to auditors or investigators.

For all matters of system documentation, setting the system boundaries is the first step of the process. During risk assessments and system audits, system boundaries define and help limit the scope of the investigation for the system. When preparing contingency plans, system boundaries help determine exactly what should be accounted for in case of an emergency. As mentioned earlier, system boundaries play a crucial role during certification and accreditation by setting boundaries of what controls will be validated during certification, and what the DAA will assume responsibility for during accreditation.

System boundaries, however, assist in operational activities beyond documentation and risk analysis. When adding security controls, system boundaries assist in determining who should pay for the control's implementation, maintenance and upgrades. Additionally, system boundaries play a crucial role in incident response activities, from knowing when an unauthorized user has made an intrusion across the system's border, to assisting determine the responsibility for mitigating any exposed vulnerabilities.

At an enterprise level, by creating solid system boundaries for all of a Department's major systems, Department managers can find both gaps of coverage as well as overlaps in security responsibility. Ideally, the boundaries of all the system at a Department should create a seamless blanket, with responsibility passed between systems at the adjoining borders. While unlikely to ever reach that ideal state, Departments can still make better enterprise decisions knowing where security responsibilities lie. In all, creating effective

system boundaries goes beyond OMB mandates or NIST guidance – it simply makes good security sense.

Appendix A: Bibliography

Berinato, Scott. "Finally, A Real Return on Security Spending." CIO Magazine, 2/15/2002, accessed online via http://www.cio.com/archive/021502/security.html (4/22/03)

Herrmann, Debra. A Practical Guide to Security Engineering and Information Awareness, CRC Press, 10/18/2001, accessed online via http://www.cccure.org/amazon/secengineering.pdf (4/14/03)

NIST 800-12, *An Introduction to Computer Security: The NIST Handbook*, (October 1995) accessed online via http://csrc.nist.gov/publications/nistpubs/800-12/handbook.pdf (4/15/03)

NIST 800-18, *Guide for Developing Security Plans for Information Technology Systems*, (December 1998) accessed online via http://csrc.nist.gov/publications/nistpubs/800-18/Planguide.doc (4/17/03)

NIST 800-26, *Security Self Assessment Guide for Information Technology Systems*, (November 2001), accessed online via http://csrc.nist.gov/publications/nistpubs/800-26/sp800-26.pdf (4/14/03)

NIST 800-37, *Guidelines for the Security Certification and Accreditation of Federal Information Technology Systems*, (Initial Public Draft, October 2002) accessed online via http://csrc.nist.gov/sec-cert/SP-800-37-v1.0.pdf (4/24/03)

NIST 800-47, *Security Guide for Interconnecting Information Technology Systems*, (September 2002), accessed online via http://csrc.nist.gov/publications/nistpubs/800-47/sp800-47.pdf (4/17/03)

Appendix III to Office of Management and Budget (OMB) Circular A-130, (11/28/00) accessed online via http://www.whitehouse.gov/omb/circulars/a130/a130appendix_iii.html (4/14/03)

Office of Management and Budget (OMB) Bulletin Number 90-08, accessed online via http://wwwoirm.nih.gov/itmra/omb90-08.html (4/21/03)