A Process for the Development of a Security Architecture for an Enterprise Information Technology System

Dr. Charles L. Smith, Sr.[1]
SenCom Corporation
1215 Jefferson Davis Highway, Suite 305
Arlington, VA 22202
1-800-471-0258
csmith@sencomcorp.com

ABSTRACT

A Team was directed to develop a set of security architecture rules (Rules Base) and its inferred Security Architecture. This architecture is to be embedded in an existing enterprise Information Technology (IT) architecture, which is referred to as the Reference Architecture in this paper. The cause for this study was a result of concerns regarding threats from internal and external entities (viz., users, devices, or application programs) to the computer and communications system, namely, network devices, information, operating systems, firmware, and applications. The rules identified for this task are for sensitive information assuming a security level of C2 (the Orange Book security category). This Rules Base, which had to be vendor and technology independent, served as the basis for a generic Security Architecture that will be embedded in the dynamic Reference Architecture. The Rules Base was created as a consensus effort among the team members, which included government, military, and contractor personnel. This effort included the following activities:

- developing a security Rules Base,
- developing an implementation plan for the interpretation of the rules so that a generic Security Architecture could be created, and
- creating the Security Architecture itself.

This paper briefly describes the first two of these activities.

1.0 Introduction

As part of the task of satisfying the needs of a particular enterprise IT community, this task was coordinated among many studies for upgrading and architecting the computer-communications system that provides support to these users.

A *systems engineering* approach was used for developing the Enterprise Security Architecture. This approach allowed for establishing a close link with *project management* because the roles of the two disciplines have considerable overlap. In the 3 June 1998 memorandum from J. S. Gansler on the Single Process Initiative to the Joint Chiefs of Staff, Gansler stated that “Civil

[1] This effort is neither promoted nor approved by any organization or agency of the Department of Defense (DoD). The concepts and methodology presented here are the results of ideas, opinions, and analyses of SenCom Corporation only.

military integration, eliminating the distinction between doing business with the government and other buyers, is critical to meeting our future military, economic, and policy objectives. The transition of the DoD to a Performance-Based Business Environment, maximizing the use of commercial items and practices, is a key step toward achieving civil military integration." An interpretation of this statement is that the military must utilize commercial business policies and products to achieve its objectives. This was an element of our approach to the security architecture development task.

If one were to implement a security architecture in a Performance-Based Business Environment, it would require at least the following actions:

- development and approval of a set of security performance measures,
- identification of the data required to quantify these measures,
- collection of the identified data,
- accurate quantification of the performance measure values, and
- dissemination and use of these performance measure values (by project managers) during the operation of the upgraded enterprise IT system as defined in the recommended system and security architecture that result from this study.

The implementation and use of appropriate systems engineering techniques will help to ensure that the correct security system is defined and built in response to the stated needs of the enterprise IT system users in an effective and efficient manner.

1.1 The Problem

Every IT system is vulnerable to attack. Security policies and products will reduce the likelihood that an attack is successful, but there is no such thing as a completely secure system. Also, there usually is no overarching security policy for the enterprise IT system, and similarly, there usually are no overarching security requirements for the enterprise IT system. Often, different organizations within an enterprise (e.g., a company or military service) investigate security issues independently, meaning that no overall enterprise IT policies nor requirements exist. Thus, one of the difficulties of creating a Security Architecture is that there are no overarching security policies and requirements.

The approach used for the creation of an enterprise IT System Architecture was to develop the architecture using close interaction with, and feedback from, the users. This approach resulted in the generation of a Reference Architecture (i.e., an overarching enterprise IT system architecture) from an *engineering perspective*. This Reference Architecture included some aspects of security but did not identify nor define them very well.

The essence of Security Architecture development is the identification of:

- IT threats,
- vulnerabilities of the IT system,
- alternative security mechanisms for countering the threats, and
- what (generically) is needed to mitigate these threats to the system.

SenCom Corporation (Dr. Charles L. Smith, Sr.) was given the task to provide team direction of this effort as co-chair of the development group. A key member of the team was Mr. Gary Murphy, Hadron, Inc., a nationally recognized security expert.

For a trusted network to be constructed from components that can be built independently (as they are in an open system (one of our security rules)), the Security Architecture must completely and unambiguously define the security functionality of components and subnetworks as well as the interfaces between or among the components and subnetworks. This architecture must be evaluated to determine that a network constructed to its specifications can in fact be trusted, that is, it must be evaluated under these interpretations. The physical realization of this architecture should result in a trusted system.

A succinct definition of the security issue for the enterprise IT system is as follows:

> **Develop a set of rules (Rule Base) given the Reference Architecture. These rules must be high level so that they can be used to create a generic Security Architecture that is compatible with the Reference Architecture.**

A simple illustration of the issues involved here is shown in figure 1. The computer network system has potential vulnerabilities in its hardware and software subsystems and there are many threats that are geared to take advantage of these vulnerabilities and are shown with the large arrows aimed at the hardware and software vulnerabilities.

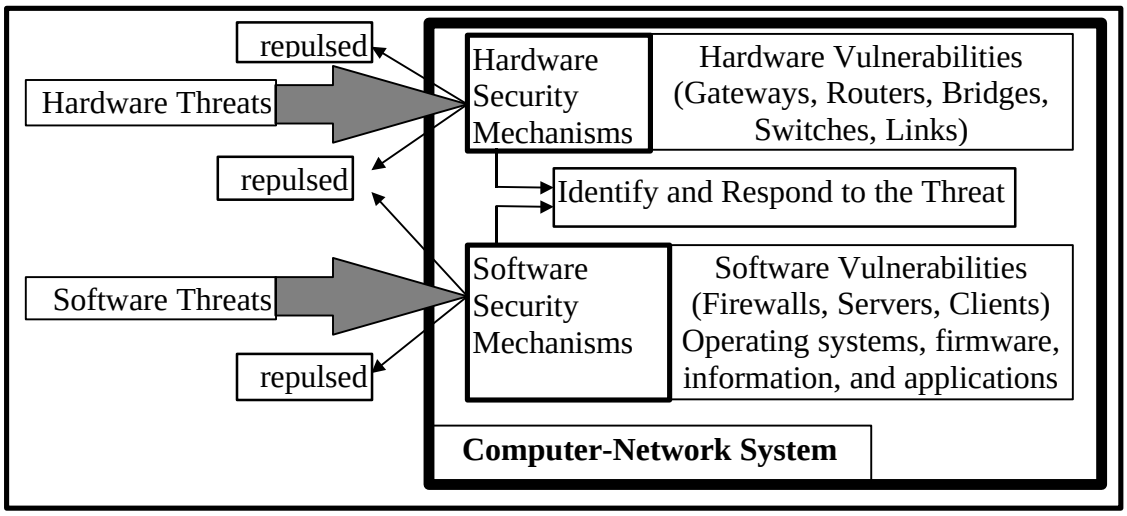

Figure 1. A Simple Illustration of a Computer-Network System with Threats and Countermeasures

By placing security mechanisms between the threats and the vulnerabilities, the threats can be repulsed, at least nearly all of them. However, some may get through, thus in addition, the threats should be monitored so that the threatening person or application can be identified. Note that

even an inept user who has no malicious motives but whose activities have negative results can be identified and revoked from further usage of the system.

There are many threats to any computer-network system and there are also many alternative security mechanisms to counter these threats. Each threat can be applied at a particular point (i.e., location, level, layer, segment, area, etc.) within an IT architecture. For each threat-location one can apply one or more security mechanisms to counter this threat and the implementation of a sequence of security mechanisms inside an IT architecture can reinforce the overall security assurance of the architecture. That is, the deeper inside an enterprise a sender's message must go to reach it intended recipient, the more firewalls it must successfully penetrate. This means that there are multiple chances for the system to block an incoming illegal or malicious message.

It is feasible that by defining and attempting to enforce a large set of security policies, standards, guidelines, and procedures, an enterprise may overly complicate its security architecture definition (for example, if these items are not consistent with one another). In this case, the implementation of the architecture becomes nearly impossible because the builder is inundated with too many requirements, some of which may be conflicting.

For this study, we decided to use the name "rules" instead of the other names, such as policies, guidelines, or procedures, that are more commonly used. We arduously attempted to ensure that none of these rules were in conflict with any other rules.

1.2 Relevance to C^2

The importance of IT security is becoming more apparent with each passing day. Just like the Year 2000 problem, IT security is usually ignored until the dangers are so apparent that finding solutions can no longer be postponed. Today an enterprise system architecture is not considered complete until it includes security mechanisms, both hardware and software, for countering the known threats to the organization developing the system.

Since the fundamental objective of some IT systems is to provide support to Command and Control (C^2), the approach or methodology used here for creating a security architecture for the system should be of interest to other C^2 users, analysts, designers, and builders. This paper offers a new emerging technology approach to the very important aspect of a C^2 systems architecture, namely, a process for developing a security architecture.

To ensure that this paper is unclassified and not sensitive to content disclosure, the techniques discussed herein are not oriented toward the exact nature of the reference IT system architecture but describe the development process itself.

1.3 Purposes

The funding organization had two objectives for this effort:

1) development of a security architecture, and
2) creation of a procedure for developing a security architecture.

The security architecture development process is needed for graduation of an organization from an “Initial Organization” to a “Repeatable Organization” in terms of its capabilities to create architectures. This paper contains a description of that development process.

1.4 Approach

The usual environment is either 1) the case of needing a security architecture for an existing or legacy system (i.e., an upgrade) or 2) the case of needing a completely new IT architecture that includes the security aspects. A logical approach often taken for these common cases is to develop security policies, then describe the resulting security requirements, and then create a security infrastructure and architecture independent of any Reference Architecture. We did not use this process since our situation was different, because we had been given a Reference Architecture that was an upgrade of the legacy system. So the actual process we followed is shown in figure 2.

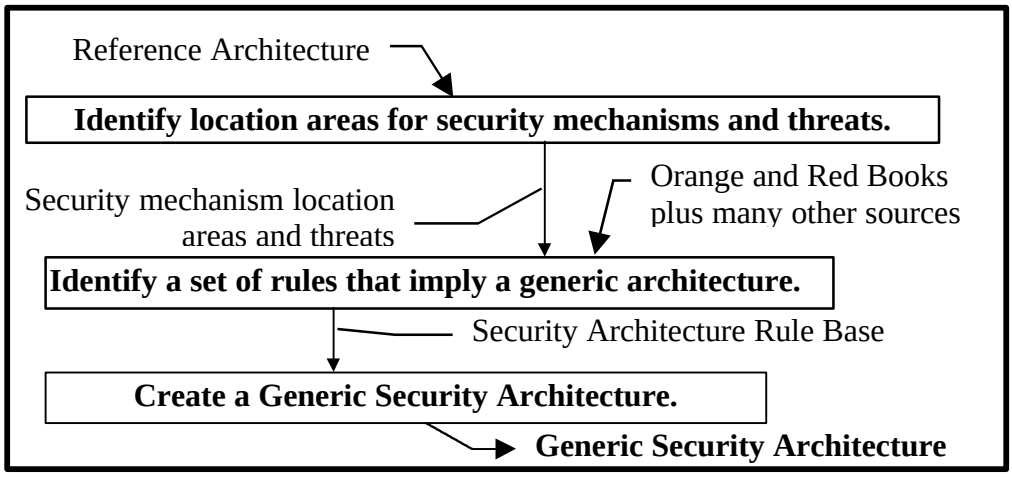

Figure 2. An Approach to Creating a Generic Security Architecture

The Reference Architecture, with an embedded preliminary security architecture, had been created from an *engineering perspective* by:

- using knowledge of available IT technologies,
- closely interacting with relevant IT users,
- interfacing with government security personnel, and
- incorporating “the best” security approaches developed by others.

The thrust of our task therefore was a need for an *architectural perspective* that would provide the elements of an architecture that were not available from the engineering approach, namely,

- a set of security rules,
- high-level security requirements, and
- an approach that would serve as a guide for developing an overarching security architecture.

An overarching security architecture must satisfy the various enterprise IT users. A high-level description of the security architecture development process that we used was the following:

Given a Reference Architecture, we;

1) identified the threats, vulnerabilities, and potential security mechanisms;
2) created a security Rules Base;
3) generated and used a glossary of relevant terms (to ensure that all involved analysts were using the same terms with the same meanings);
4) developed an implementation process for interpreting the rules in terms of a security architecture;
5) developed a Security Architecture;
6) compared the Security Architecture with the Reference Architecture; and
7) modified, as required, the Reference Architecture to ensure that it contained the embedded Security Architecture attributes.

We called this process a *pragmatic approach*. Most of the rules we identified resulted from statements in the Orange Book [DoD 85] and the Red Book [NCSC 87]. The Orange and Red Books say that the minimal C2 requirements include the following:

- Security policy (define the form of the Discretionary Access Control (DAC) policy that is enforced in the network to prevent unauthorized users from reading the sensitive information entrusted to the network),
- DAC,
- Object reuse,
- Accountability (identification and authentication, audit),
- Assurance,
- System integrity,
- Life-cycle assurance (i.e., security testing), and
- Documentation (Security Features User's Guide, Trusted Facility Manual, Test Documentation, and Design Documentation).

A set of rules for the system security architecture was defined but the complete set is not given here since these rules are sensitive and identify the funding organization. The Security Architecture and the concomitant rules included concepts for three perimeters, see figures 3 and 4:

1) an outside or enterprise Perimeter,
2) intermediate Perimeters (intermediate level), and
3) the organizational or interior Perimeters (interior level).

We understood that each intermediate organization or interior organization could implement an overlay security capability of its own that would raise its security level to a higher echelon that met their specific requirements. So it is important to understand that the Security Architecture that we defined was intended to provide a basic capability to block or track most IT traffic, both incoming and outgoing, but not necessarily all.

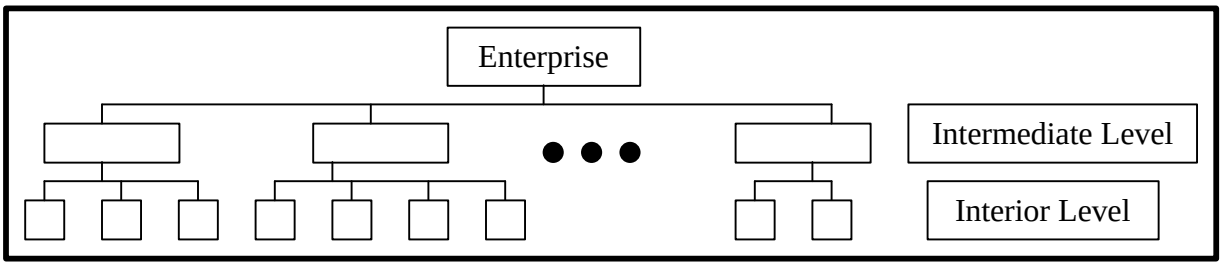

Figure 3. A Three Level Hierarchy for an Enterprise

It was essential that the final set of rules be relevant to each of the seven Open Systems Interconnection (OSI) levels. The Security Architecture must correspond to all OSI levels, be compatible with the Reference Architecture, and be extensible to meeting the needs of all IT users within the enterprise.

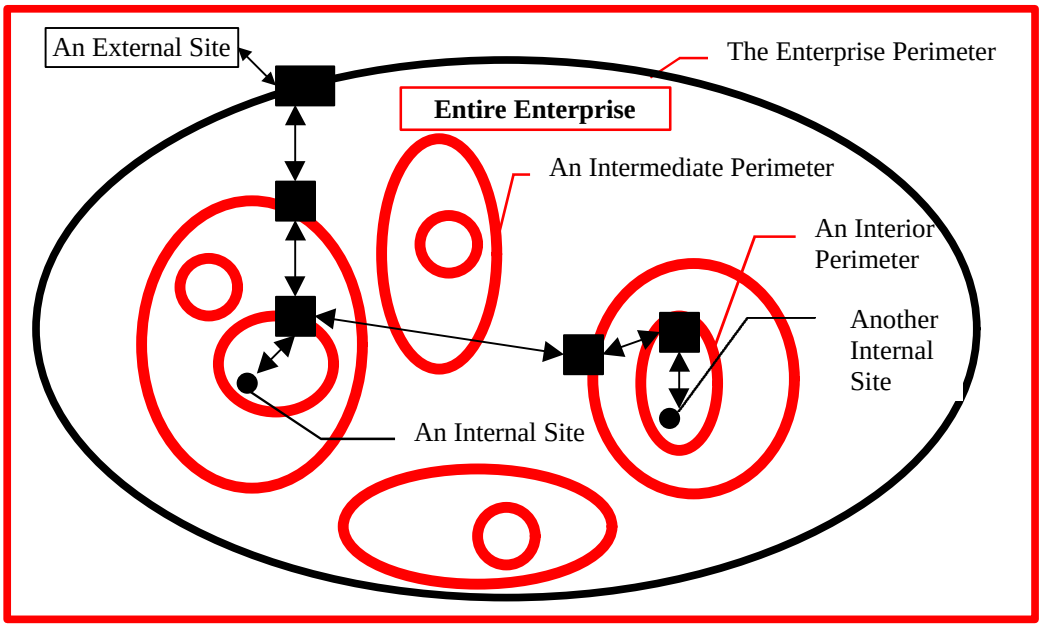

Figure 4. The Security Perimeters

1.5 Some Definitions

Some relevant definitions required to properly understand the security architecture process are listed here, such as:

An Architecture - is a set of models and the accompanying words needed to understand these models. A *Reference Architecture* is the architecture of the overall system whereas *a security architecture* is the subset architecture that provides for security of the overall system. There are three perspectives, operational, technical, and systems that are defined as follows:

1) an *Operational Architecture* provides a description of the tasks and activities, operational elements, and information flows to accomplish or support a military operation;

2) a *Technical Architecture* (the primary objective of this study) provides a minimal set of rules governing the arrangement, interaction, and interdependence of system parts or elements, whose purpose is to ensure that the system, as defined, satisfies a specified set of requirements as stipulated by the users; and
3) a *Systems Architecture* (a secondary objective of this study) provides a description of physical systems and interconnections (and their attributes) providing for, or supporting, warfighting functions.

The descriptions are both verbal and graphical. Other definitions are as follows.

Audit - The recording of network activity and later examination of the records to identify user network accesses and activities. The act of gathering usage information on a computer system with the intent of detecting and deterring penetration of the system, and revealing usage that identifies misuse.

Audit Control - The detection of security policy violations, assessing system vulnerabilities, and analysis of ongoing activities to help system administrators determine preventative and corrective security measures.

Authentication - The process of proving that a subject is who or what the subject claims to be. It is a measure used to verify the eligibility of a subject and the ability of that subject to access certain information. It protects against the fraudulent use of a system or the fraudulent transmission of information. There are three classic ways to authenticate oneself: something you know, something you have, or something you are.

Authorization - The granting of rights to a user, a program, or a process. This includes the granting of access based on specific access rights.

Availability - The property that ensures that a resource is accessible and usable upon demand by an authorized principal.

Controlled Access Point - This is a location in the network where external or internal messages are approved for delivery or blocked.

Intrusion Detection - The capability to detect an unauthorized user either through proactive examination or reactive analysis using sensors, software, hardware, etc. at a server down to a port level.

Non-repudiation - The property that enables the receiver of a message to prove that the sender did in fact send the message even though the sender might later desire to deny ever having sent it.

Other important definitions include the following: *Security* is used here in a sense of minimizing the vulnerabilities of assets and resources. An *asset* is anything of value in an enterprise computing system [ISO, 1988]. A *vulnerability* is any weakness that could be exploited to violate a system or the information it contains. A *point of vulnerability* is a location (software or device)

within a system that is susceptible to attack. A *threat* is a potential violation of security or a possible danger to the system; the danger might be a person on a personal computer with an active modem who is attempting to enter the system and do some harm or to discover secrets. *Computer security* refers to a complex set of procedural, logical, and physical mechanisms aimed at prevention, detection, and correction of certain kinds of misuses, together with the tools to install, operate, and maintain these mechanisms [ECMA, 1988].

A few additional definitions are needed to understand the processes to be described. There are three separate aspects of computer security: secrecy, accuracy or integrity, and availability [Russell and Gangemi, 1991]. A *secure computer system* must not allow information to be disclosed to anyone who is not authorized to access this information (also called data confidentiality). Secrecy and need-to-know should ensure that users access only information that they are allowed to see. *Users* are those people who actually sign on to use the computer system online. *Accuracy or integrity* means that the system must not corrupt the information or allow any unauthorized malicious or accidental changes to it. A related variant of accuracy is known as *authenticity* which provides a way to verify the origin of data by determining who entered or sent it, and by recording when it was sent and received. *Availability* means that the computer system's hardware and software are working properly and a system that has high availability is able to recover quickly and completely if a disaster should occur. The opposite of availability *is denial of service*. This means that the system users are unable to get resources that they need. The term *sensitive* is used to denote information that is not classified but is sensitive none the less.

The method for the listing of authorized users (individually or by group membership) shall provide for simplicity and ease of administering the following situations:

1) changing access privileges (i.e., adding or subtracting privileges),
2) deletion of accesses for users who no longer have rights (e.g., leave the organization or pass away), and
3) addition of accesses for users who are new to the system (e.g., new organizational employees).

This method should provide for the minimization of administrator errors (which can lead to attacks and successful intruder break-ins) through use of a method that can be automated for computerization. Currently, the three principal methods for access control are discretionary, mandatory, and role-based access control. Role-based access control probably offers the best alternative for authorizing access, and therefore minimizing mistakes because it can be automated, by administrators.

1.6 Study Results

A Security Rules Base was developed, an implementation process was created, and a Security Architecture was developed from the Rules Base. These achievements are documented in a draft report that was delivered to DoD at the end of March 1999.

2.0 Analysis

The Reference Architecture infrastructure consisted of routers, gateways, firewalls, Fiber Distributed Data Interface (FDDI) networks, Local Area Networks (LANs), Asynchronous Transfer Mode (ATM) networks, switches, servers, personal computers (PCs), and laptops. We used a graphical abstraction of the Reference Architecture to begin the process of deriving the Rules Base.

2.1 Security Rules Base Development

The process used for developing the Security Rules Base was as follows, we:

1) organized a security architecture team;
2) developed a Rules Base taxonomy;
3) identified the threats, vulnerabilities, and security mechanisms;
4) stated the assumptions needed for developing the Rules Base;
5) gathered and reviewed all relevant reference material;
6) created or modified the Rules Base and implementation process (who uses, how used, and categories for the rules);
7) reviewed the Rules Base (by the architecture team);
8) iterated back to steps 2, 3, 4, 5, 6, and 7 as required;
9) when team consensus to the Rules Base had been achieved, it was sent forward for higher level approval and/or modification; and
10) when the Rules Base and the implementation process had been approved, we used the Rules Base for creating the generic Security Architecture.

The members of the team were talented people who had experience with the development of security policy, security requirements, and security architectures. A team consensus arrived at a taxonomy of the Rules Base which is shown in figure 5.

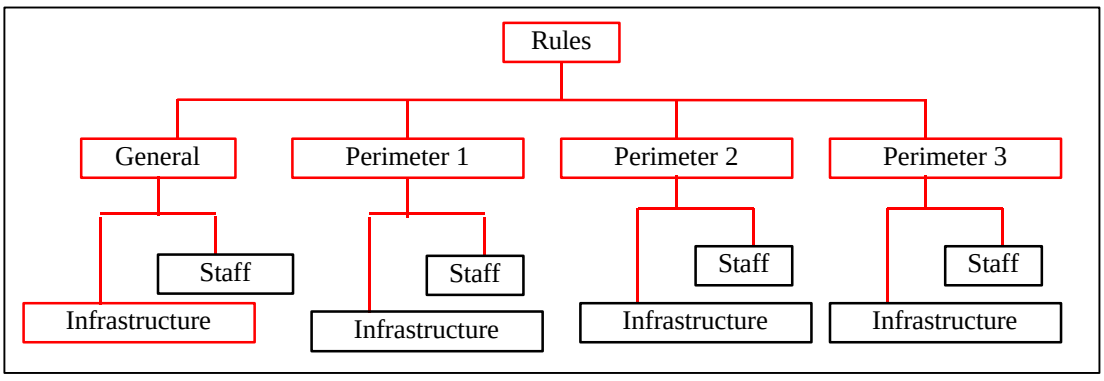

Figure 5. A Taxonomy for the Rules Base

We assumed that the rules would be used in the following manner:

1) *Who should use the rules*, that is, which rules are used by *network engineers* (those familiar with the OSI levels 1 through 4 systems), *system engineers* (those familiar with the OSI levels 5 through 7 systems; sometimes called a host engineer), or

security engineers (those familiar with the security aspects of the OSI levels 1 through 7 systems)?

2) *How should the rules be used* to guide the engineers, that is, how are the rules implemented (for Architecture Design, System Selection, or Organizational Design and Responsibilities)?
3) *To what parts of the system should the rules be applied,* that is, what subsets of the rules apply to particular infrastructure components (viz., hardware (HW), software (SW), HW and SW, Staff, SW and Staff, and All (HW, SW, and Staff))?

The Security Rules Base should have the following attributes:

- The Rules Base addresses all of the security issues that it should.
- The Rules Base does not address any security issues that it should not.
- The rules in the Rules Base are properly categorized.
- Each rule in the Rules Base is easy to understand.
- Each rule in the Rules Base is accurately stated with a minimum of words.
- No rule in the Rules Base is redundant with any other rule.
- No rule in the Rules Base is in conflict with any other rule.
- The rules in the Rules Base are stated at an appropriate level, i.e., no vendors nor technologies are mentioned nor is any particular implementation implied.
- The rules in the Rules Base are properly prioritized.
- One person is responsible for formulating, correcting, and maintaining the Rules Base.

The following assumptions were used as guidance during the process of developing these rules:

1) The rules are for a target system security capability.
2) The rules are derived from the Reference Architecture.
3) The rules assume that all information is at the Sensitive level.
4) The rules assume that security protection is at the C2 level.
5) The rules are for a minimal level of security capability for computer, communications, and operations methods and do not consider any other aspect of security such as physical security (except for the isolation rule) or electronic emissions protection.
6) The architecture inferred by these rules shall allow only traffic with proper authentication and authority and deny all other traffic.
7) Anyone who connects to the enterprise network must comply with these rules.
8) The rules are for a system where a separate subnetwork exists for each major organization within the enterprise.
9) The rules are for an architecture with a single point of entry/exit for each Perimeter (i.e., the entire enterprise, each intermediate organization, and each interior organization).
10) An authorizing and administrating organization shall be responsible for accepting the varying risks associated with components of the architecture infrastructure.
11) The range of the rules is limited to access control processes.
12) The rules are categorized according to the taxonomy shown in figure 5.
13) The rules do not include any vendor- or technology-oriented statements.
14) This Rules Base can be used to infer a generic security architecture, and this generic architecture encompasses the security aspects of the Reference Architecture.

15) The terms used in this Rules Base are defined in an appendix (Glossary) that will accompany the final report.
16) The purpose of this Rules Base is to provide guidance for designing a security architecture for the Reference Architecture that meets the needs of enterprise users.

2.2 Implementation Process

The rules in the Rules Base can be implemented according to the information contained in each row of the table. *General rules* are those rules that apply throughout the enterprise. *Rules for Perimeter 1* apply at the enterprise boundary. *Rules for Perimeter 2* apply at, and within, the boundary of any intermediate zone but outside Perimeter 3. *Rules for Perimeter 3* apply at, and within, the boundary around any interior organization.

The Security Architecture Team that developed the Rules Base will also update it, and will develop the implementation process and update that also, as revisions are required. The procedure is depicted in figure 6.

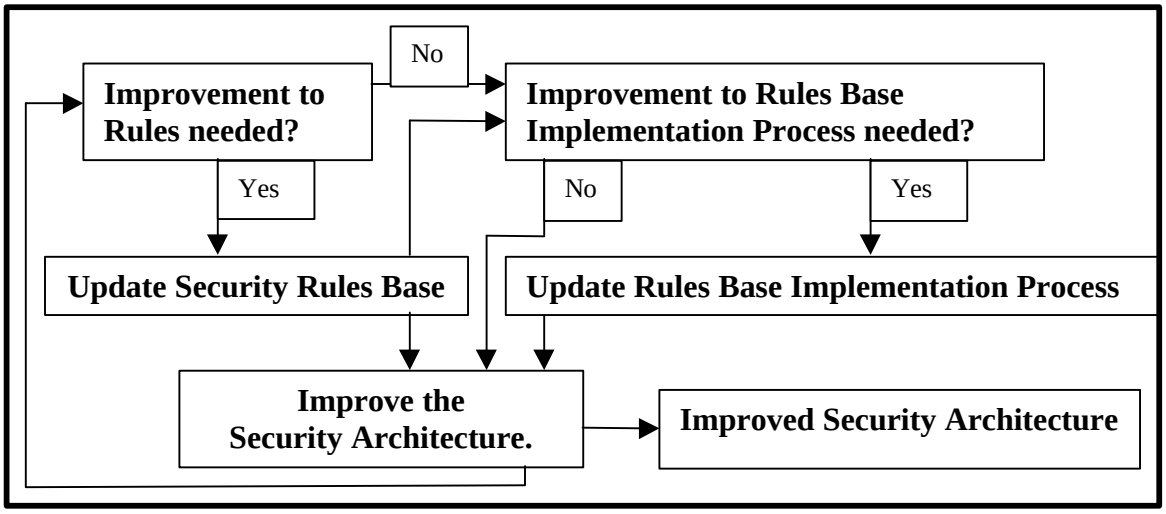

Figure 6. Updating the Dynamic Rules Base and Process

The Rules Base contained eighty-six rules that applied to the three perimeters (Perimeter 1, Perimeter 2, and Perimeter 3). Forty-four of these rules were generic and applied to each of the three perimeters, thirteen rules applied to Perimeter 1, fourteen applied to Perimeter 2, and fifteen applied to Perimeter 3. The manner in which the security architecture was created as a function of the rules base is shown in figure 7. The resulting *Generic Security Architecture*, identifies the three security perimeters and the security mechanisms, such as firewalls and traffic monitors.

The format used for describing the rules is shown in table 1. Interpretation of the enterprise security rules is as follows. The first column (Who Uses) tells who is responsible for applying the rule in that row. The second column (How Used) tells how this rule should be used. The third column (Category) tells what infrastructure or personnel element to which the rule applies. The last column (General or Perimeter Rule) contains the statement of the rule. The process for using

the table can best be described through an example. For the first row (which is all that is shown in this paper), a *network engineer* should use the rule (Open Architecture) for architecture design and the rule applies to the *hardware and software* aspects of the system. The other rows should be interpreted similarly.

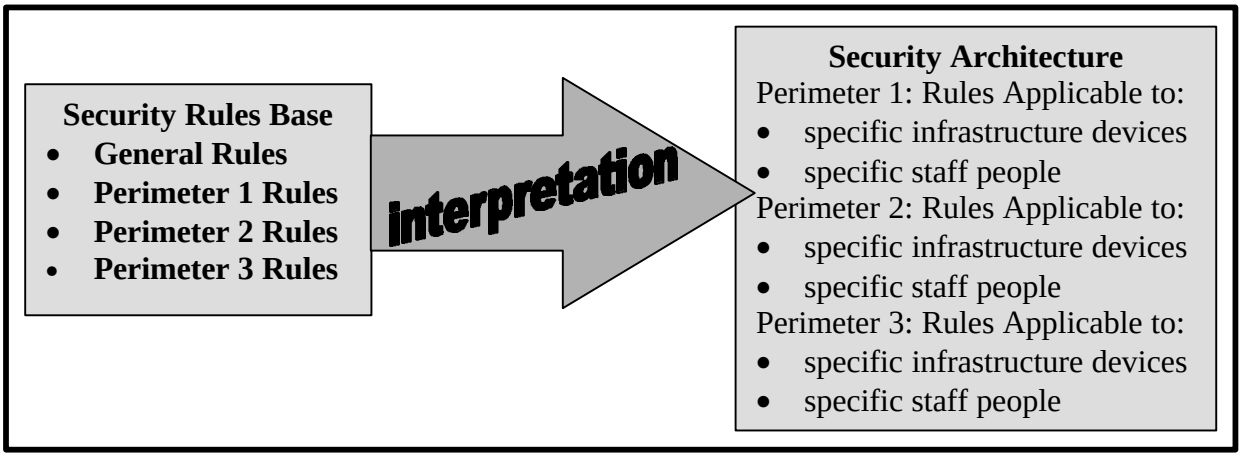

Figure 7. Interpretation of the Security Rules for the Security Architecture

Table 1. A Sample of the Enterprise IT Security Rules Base with Other Information

Who Uses	How Used	Category	General Rules
Network Engineer	**Architecture Design**	**HW/SW**	**Rule A.1.1 (Open Architecture)** - Ensure that all components of the network resulting from the architecture are interoperable, scalable, and portable, that is, they are components of a standards-based or open architecture. Preclude any systems that are proprietary.

The White Paper concomitant to the Presidential Directive PDD-63, May 22, 1998, states "Frequent assessments shall be made of our critical infrastructures' existing reliability, vulnerability, and threat environment because, as technology and the nature of the threats to our critical infrastructures will continue to change rapidly, so must our protective measures and responses be robustly adaptive."

Another rule that was developed by the security architecture team states the following:

Rule A.2.3 (Improvement Process) - [paraphrased]

- acquire feedback information on existing or potential problems,
- formally document the problems,
- use the information to identify alternative responses,
- select the appropriate response,
- implement the response,
- test the response, and
- approve or modify.

This rule meets the needs for a learning process identified in the PDD 63 white paper [WHTP 98].

2.3 Security Architectural Development Process

The process used for developing the Security Architecture was to use the rules and the defined implementation guidelines, as well as the Reference Architecture definition. At some point, it is anticipated that the resulting Security Architecture will be compared with the Reference Architecture, and several modifications may be required.

3.0 Summary

In this paper, we presented a security architecture development process that required the creation of a Security Rules Base and some guidelines for making use of this Rules Base during the generation of a generic Security Architecture. Because of the sensitivity of the rules base and the concomitant security architecture, these were not presented in this paper.

References and Bibliography

DoD 85 *Department of Defense Trusted Computer System Evaluation Criteria*, DOD 5200.28 STD, Department of Defense, December 1985. (The Orange Book)

NCSC 87 *Trusted Network Interpretation (of The Trusted Computer System Evaluation Criteria(TCSEC))*, National Computer Security Center (NCSC), NCSC-TG-005, 31 July 1987. (The Red Book)

DoD 88 *Security Requirements for Automated Information Systems (AISs)*, DOD 5200.28, Department of Defense (DoD), March 21, 1988.

OMB *Management of Federal Information Resources*, Office of Management and Budget (OMB) Circular No. A-130, December 12, 1985.

RUSS *Computer Security Basics*, Russell, D., and G. Gangemi, Sr., O'Reilly & Associates, 1991.

DISAa *Global Combat Support System (GCSS) Security Requirements*, Defense Information Systems Agency (DISA), 13 April 1998.

DISAb *Department of Defense Public Key Infrastructure Requirements*, Defense Information Systems Agency (DISA), PKI Working Group, 20 April 1998.

AFCA *The Information Protection Barrier Reef 12 Step Process: Barksdale AFB Case Study*, Headquarters Air Force Communications Agency (AFCA), Network Strategies Office, Scott AFB, Illinois, 19 August 1997.

ISO *Information Processing Systems - OSI RM: Part 2: Security Architecture*, ISO/TC 97 7498-2, International Organization for Standards, 1988.

ECMA *Security Framework (TR46)*, European Computer Manufacturers Association, June 1988.

CSC *Computer Security Requirements: Guidance For Applying The Department Of Defense Trusted Computer System Evaluation Criteria In Specific Environments*, CSC-STD-003-85, DoD Computer Security Center (CSC), 25 June 1985.

NIST *Computer Security Policy*, National Institute of Standards and Technology, 1994.

JTAA, *Joint Technical Architecture - Army, Interoperability and Soldier Support, Version 5*.5, Department of the Army, 23 December 1998.

OPPL Oppliger, Rolf, *Internet and Intranet Security*, Artech House, 1998.

PRES *Protecting America's Critical Infrastructures*, Presidential Decision Directive (PDD) 63, The White House, Office of the Press Secretary, May 22, 1998.

WHTP *The Clinton Administration's Policy on Critical Infrastructure Protection: Presidential Decision Directive 63, A White Paper*, The White House, Office of the Press Secretary, May 22, 1998.

Brief Biography for Dr. Charles L. Smith, Sr.

Dr. Smith has over twenty years experience as a DoD analyst and manager. He has worked with MITRE, ANSER, and many other DoD contractors. He has provided consulting services to Chrysler Corporation at their world headquarters, to the Federal Deposit Insurance Corporation, to the National Institute of Standards and Technology, and to many others. He is currently with SenCom Corporation, which is headquartered in Bedford, Massachusetts. Dr. Smith is performing analyses for a DoD computer network, creating a security architecture and a Web architecture that meet the needs of IT users. Dr. Smith has BS and MS degrees in Mathematics and a PhD in Information Technology and Systems Engineering. He has contributed many technical papers to DoD conferences and seminars. He is the author of a recent book from Ablex Publishing titled “Computer-Supported Decision Making,” April 1998. His paper today concerns a procedural methodology for creating a security architecture that meets the needs of a certain group of IT users.