2000

A trusted connection framework for multilevel secure Local Area Networks

Wilson, Jeffery Dwane.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A TRUSTED CONNECTION FRAMEWORK FOR MULTILEVEL SECURE LOCAL AREA NETWORKS

by

Jeffery Dwane Wilson

June 2000

Thesis Advisor: Cynthia E. Irvine
Second Reader: Timothy Levin

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE June 2000	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE A Trusted Connection Framework for Multilevel Secure Local Area Networks	5. FUNDING NUMBERS
6. AUTHOR(S) Jeffery D. Wilson	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

The Naval Postgraduate School is developing a Multilevel Secure Local Area Network (MLS LAN) that incorporates commercial-off-the-shelf client workstations to provide multiple users with simultaneous secure access to stored data of different sensitivity levels. The MLS LAN uses a Trusted Computing Base Extension (TCBE) in the LAN's client workstations to extend the TCB from the trusted server across the network to these workstations. Connections between elements of the LAN are under TCB control and are conducted by way of several new communications protocols.

Using a realistic System Requirements Document and a High Level Protocol Analysis, this thesis presents a framework of communications protocols that will enable the components of the MLS LAN to securely interact. The framework first presents a communications channel protocol that protects all data transmitted on the network. Following this, three other protocols are described that enable MLS LAN users to safely login and negotiate a secure session, access Application Protocol Servers that provide services such as e-mail or WWW services, and to use typical LAN-based office automation services. Finally presented is an analysis of both TLS and IPSec, which provides evidence that IPSec is best suited to provide MLS LAN communications protection.

SUBJECT TERMS Multilevel Security, Trusted Path, High Assurance, Network Client-Server	15. NUMBER OF PAGES 178
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20.
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18 298-102

Approved for public release; distribution is unlimited

A TRUSTED CONNECTION FRAMEWORK FOR MULTILEVEL SECURE LOCAL AREA NETWORKS

Jeffery Dwane Wilson
Lieutenant Colonel, United States Marine Corps
B.S., Bluefield College, 1982

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
June 2000

Author: [redacted]
Jeffery D. Wilson

Approved by: [redacted]
Cynthia Irvine, Thesis Advisor

[redacted]
Timothy Levin, Second Reader

[redacted]
Dan Boger, Chairman
Department of Computer Science

ABSTRACT

The Naval Postgraduate School is developing a Multilevel Secure Local Area Network (MLS LAN) that incorporates commercial-off-the-shelf client workstations to provide multiple users with simultaneous secure access to stored data of different sensitivity levels. The MLS LAN uses a Trusted Computing Base Extension (TCBE) in the LAN's client workstations to extend the TCB from the trusted server across the network to these workstations. Connections between elements of the LAN are under TCB control and are conducted by way of several new communications protocols.

Using a realistic System Requirements Document and a High Level Protocol Analysis, this thesis presents a framework of communications protocols that will enable the components of the MLS LAN to securely interact. The framework first presents a communications channel protocol that protects all data transmitted on the network. Following this, three other protocols are described that enable MLS LAN users to safely login and negotiate a secure session, access Application Protocol Servers that provide services such as e-mail or WWW services, and to use typical LAN-based office automation services. Finally presented is an analysis of both TLS and IPSec, which provides evidence that IPSec is best suited to provide MLS LAN communications protection.

TABLE OF CONTENTS

I. INTRODUCTION 1

A. BACKGROUND 1

1. Purpose 1

2. Project Overview 5

3. MLS LAN Project Goals 10

4. Thesis Goals 10

B. CHAPTER OVERVIEW 11

1. Introduction 11

2. The MLS LAN Systems Architecture 11

3. Protected Communications Channel Security 12

4. Overview of the MLS LAN Connection Framework 12

5. Conclusions and Recommendations 13

C. APPENDIX OVERVIEW 13

1. Appendix A: The NPS CISR MLS LAN Systems Requirement Document 13

2. Appendix B: The NPS CISR MLS LAN Protocol Requirements Document 13

3. Appendix C: The MLS LAN Connection Protocol Framework Document 13

II. NPS MLS LAN SYSTEMS ARCHITECTURE 15

A. THE MLS LAN PROJECT ARCHITECTURE OVERVIEW 15

1. System Definition and Accreditation 15

a. The Bell and LaPadula (BLP) Model 16

b. The Biba Integrity Model 18

2. Component Description 19

a. The Trusted Computing Base 19

(1) Protected Channel Initiator 22

(2) Session Database Server 23

(3) Trusted Computing Base Extension Server 23

(4) Trusted Computing Base Extension 26

(5) MLS LAN Connection Protocols 26

b. MLS LAN Network Application Protocol Services 28
(1) Secure Session Server 29
(2) Application Protocol Server 29
c. MLS LAN Workstation 30

III. PROTECTED COMMUNICATIONS CHANNEL SECURITY 31
A. OVERVIEW 31
B. TRANSPORT LAYER SECURITY PROTOCOL 32
1. Client/Server Hello Messages 35
2. Key Generation Messages 36
C. INTERNET PROTOCOL SECURITY 38
1. Security Policy 39
2. Security Protocols 42
3. Key Negotiation and Management Structure 45
D. SECURITY OPTIONS APPLICABILITY 47
1. Application Client/Server Modification 47
2. Security Policy 47
3. Domain of Interpretation 48

IV. OVERVIEW OF COMMUNICATIONS PROTOCOL FRAMEWORK 51
A. THE MLS LAN PROTECTED COMMUNICATIONS CHANNEL PROTOCOL 51
1. Overview 51
2. Logical Placement of MLS LAN IPSec 52
3. IPSec Security Policy for the MLS LAN 52
4. IPSec Key Management for the MLS LAN 54
5. MLS LAN PCC Processing 54
B. TCB-TCBE CONNECTION PROTOCOL 55
1. Overview 55
2. TCB and TCB Extension Server States 56
a. TCBE States 56
b. TCB Extension Server States 57
3. TCB-TCBE Connection Protocol Datagrams 57
a. Payload Datagram 58
b. Command Datagram 58

4. TCB-TCBE Connection Protocol Processing 60
C. SESSION STATUS PROTOCOL 63
1. Overview 63
2. TCB Extension Server and Session Database Server States 63
a. TCB Extension Server States 63
b. Session Database Server States 64
3. Session Status Protocol Datagrams 64
a. Request Datagram 64
b. Response Datagram 65
4. Session Status Protocol Processing 66
a. Use of Session Status Protocol by TCB Entities other that TCB Extension Server 66
b. Use of the Session Status Protocol by the TCB Extension Server 66
D. TCBE-TO-SESSION SERVER CONNECTION PROTOCOL 68
1. Overview 68
2. TCBE and Secure Session Server States 69
a. TCBE States 69
b. Secure Session Server States 69
3. TCBE-to-Session Server Connection Protocol Datagrams 69
4. TCBE-to-Session Server Connection Protocol Processing 69
V. CONCLUSIONS 71
A MLS LAN Project Development 71
1. Previous Efforts 71
a. NPS Thesis: Secure Local Area Network Services for a High Assurance Multilevel Network, by Susan BryerJoyner and Scott Heller, March 1999 71
b. NPS Thesis: Design of a High Assurance, Multilevel Secure Mail Server (HAMMS), by James Downey and Dion Robb, September, 1997 71
c. NPS Thesis: Analysis for a Trusted Computing Base Extension Prototype Board, by Bora Turan, March, 1997 72
2. Engineering Team Effort 72
a. Mission and Format 73
b. Leadership and Team Composition 73
c. Documentation 74

B. FUTURE WORK 74
1. Limitation of Session Sensitivity Levels 74
2. Acceptance of a Non-TCBE-Equipped Workstation 75
3. Non-TCBE Equipped Workstations Access to Application Protocol Servers 75
4. Session Domination Algorithm 76
5. Protected Channel Initiaitor 77
6. Distributed Session Database 77
7. Session Time Control Mechanism 77
8. TCB-TCBE Trusted Path Connectivity 78
9. MLS LAN Domain of Interpretation 78
C. CONCLUSIONS 78
APPENDIX A. MLS LAN SYSTEM REQUIREMENTS DOCUMENT 81
APPENDIX B. MLS LAN PROTOCOL HIGH LEVEL ANALYSIS DOCUMENT 97
APPENDIX C. MLS LAN CONNECTION FRAMEWORK DOCUMENT 113
LIST OF REFERENCES 153
INITIAL DISTRIBUTION LIST 157

LIST OF FIGURES

Figure 1.1 The Reference Monitor Concept 8

Figure 2.1 MLS LAN Components 20

Figure 2.2 XTS-300 Systems Architecture 22

Figure 2.3 TCB Extension Server Interactions 27

Figure 2.4 MLS LAN Connection Protocols 28

Figure 2.5 Secure Session Server / Application Protocol Server Interaction 30

Figure 3.1 TLS Protocol Stack [Ref.13] 33

Figure 3.2 TLS Record Protocol Operation [Ref. 15] 34

Figure 3.3 TLS Handshake Protocol Message Exchange [Ref. 16] 37

Figure 3.4 IPSec Implementation Architecture [Ref. 20] 42

Figure 3.5 Authentication Header Diagram [Ref. 20] 43

Figure 3.6 ESP Packet in Transport Mode [Ref. 19] 45

ACKNOWLEDGEMENTS

I would first like to thank my wife Tracy, and our children, Benjamin and Stephanie for their enduring patience with me as I completed this program. The work at the school and the effort placed on this thesis took many hours of family time, for which they gratiously gave.

I am also thankful to my Thesis Advisor and Second Reader. Dr. Cynthia Irvine's exceptional knowledge in the areas of computer systems engineering, software engineering, and most significantly, computer security were the impetus for my successful completion of this thesis. My Second Reader, Tim Levin, will probably never know how much his calm, reassuring manner brought focus to the myriad of issues which stumped me throughout my research.

Finally, I must thank the members of the MLS LAN engineering design team. Their many hours spent discussing, and sometimes arguing, the smallest details of network and protocol design produced a significant product. Of particular note, were the efforts of Dave Shifflett. Dave's unending patience and tutelage taught me to think as a computer scientist and I will forever be in his debt.

I. INTRODUCTION

The primary objective of this thesis is to investigate and define a communications framework for a multilevel secure, high assurance network. To sufficiently define this framework, a network security architecture must be proposed. This thesis will, therefore present both the proposed network security architecture and the communications framework.

A. BACKGROUND

1. Purpose

Almost all of the organizations in the United States Government and corporate America can place information they use and maintain into two distinct categories. The first, and probably most widely used, contains documents and data that are considered by the originating organization to be "non-proprietary" in nature. Information in this category is not regarded to be vital to the security or integrity of the organization's productivity and therefore is available to the public for use. The other category contains information that is considered to be "proprietary" in nature. These proprietary documents contain some information that, if released to their competitors, could cause some level of damage to the organization's productivity. The information in the "non-proprietary" category could comfortably be assigned a single label of "releasable" or "open to the public" as everyone considers all of the information similarly accessible. For proprietary information, however, additional label considerations are usually required to delineate the specific level of damage that may occur were the information be inadvertently released. The most recognized example of this is the military's security classification system. Information considered to cause "exceptionally grave damage to the national security" if

released is labeled "Top Secret" while information considered to cause only "serious damage to the national security" is labeled "Secret" [Ref. 1]. The actual determination of what is "grave" or "serious" is based upon the Government's National Security Policy and is assigned by the individual or organization that creates the information. While the basis for label determination is beyond the scope of this work, it does illustrate the necessity for multiple levels of data security within a given organization.

The question is then, how does the organization provide their authorized users access to both proprietary and non-proprietary labeled information while simultaneously ensuring the information's protection? Unfortunately there is no easy answer. Most organizations use one of four security modes of operation to accomplish this task. The easiest to implement is known as "Dedicated" mode. A dedicated network security solution involves the creation of mutually exclusive "stovepipe" networks that are configured to handle only a single level of data security. To gain access to a specific network each user must be cleared and have a "need to know", or requirement, for all information on that network. This solution creates two distinct problems for the organization. The first is the requirement to deploy redundant hardware and network configurations throughout the organization to support each of the unique data security levels. This significantly increases the cost of the organization's information technology (IT) structure. The second, and probably more significant, is the duplication of effort in the areas of system administration and infrastructure management.

A second operational mode is known as "System High". The system high solution labels all of the organization's proprietary information to its highest sensitivity level. This solution effectively forces everyone to be cleared to the same high level and

have a "need to know" for at least some of the information contained on the network. This solution does reduce the redundancy in the hardware implementation; however, the organization loses much of the security level granularity that most likely was the impetus for the security segregation of the information in the first place. This solution also adds complexity to the organization's ability to interoperate with other networks. All information created on the network, even if it is considered non-proprietary relative to another organization, must be transmitted out at "system high". This creates a huge problem for the receiving organization in the handling of this new "classified material".

The third mode of operation typically used is known as the "Compartmented Mode". This solution is similar to the "Dedicated Mode". All information and users are given the same sensitivity level, but the system provides a number of non-hierarchical compartments to confine access and segregate the information. The use of compartments allows the organization to grant access to proprietary information, not only on the basis of its security value, but also on the user's need to access that specific compartment. While this does not alleviate the need for redundant hardware architectures, e.g., the organization must still have separate systems for each sensitivity level, it does provide the organization with a robust solution for the required security segregation.

The most versatile, and complex of the operational modes is the "True Multilevel Security Mode". This solution enables an organization to maintain a single network that is sufficient to verifiably restrict access to only that data for which the user is both cleared and has the requirement to see, even though the network contains data at multiple sensitivity levels. A true Multilevel Secure (MLS) network can eliminate the architectural and administrative redundancy found in the previous solutions while

providing a well-defined structure for data sensitivity and data integrity differentiation. [Ref. 2] The most profound difficulty with this operational mode is the lack of a reasonably priced commercially available MLS network solution in today's marketplace.

In 1997, The Naval Postgraduate School, Center for Security and Information Security (INFOSEC) Studies and Research (NPS CISR) began to evaluate a possible solution for this problem. The research team envisioned the development of a network that incorporated the use of a small number of high cost servers, previously verified to provide high assurance in stand-alone systems, as the foundation for their multilevel assurance and protection. The client workstations connecting to the network were envisioned as inexpensive, "diskless" personal computers. Access to network information would be exclusively controlled by the network's security infrastructure or "Trusted Computing Base" that enforced the organization's security policy. The result of this vision is a system that is both multilevel secure and reasonably priced.

Once developed, this network would be suitable for evaluation using a defined criterion such as the Department of Defense Trusted Computer Security System Evaluation Criteria, DoD 5200.28-STD (TCSEC) [Ref. 3] or its successor, the Common Criteria for Information Technology Security Evaluation Version 2.1 [Ref. 4]. These documents provide standard security criteria for computer systems and specify technical methodologies, which can be used to evaluate the system's ability to support the security policy. The NPS CISR plan fell squarely into the "Multilevel Secure" class of systems that the TCSEC defined as "system[s] containing information with different sensitivities that simultaneously permits access by users with different security clearances and need-to-know, but prevents users from obtaining access to information for which they lack

authorization." [Ref 3.] From this definition the NPS CISR network plan became known as the Multilevel Secure Local Area Network Project or MLS LAN Project.

2. Project Overview

Any multilevel networking solution proposed for real world use must provide the required functionality, but it must also ensure that the information being secured can only be accessed by those users to whom the organization has granted access. This implies that the organization must first define a policy concerning access to the system's information. Sterne, [Ref. 5], breaks this notion down into three distinct areas. The first is the notion of establishing security policy objectives, or the "statement of intent to protect an identified resource from unauthorized use". Once this has been defined, the organization can develop a "set of laws, rules, and practices that regulate how an organization manages, protects and distributes resources to achieve [these] specified security policy objectives". These rules are known as their "Organizational Security Policy". From this is developed an "Automated Security Policy" that addresses the set of restrictions placed on computing systems to prevent violation of the Organizational Security Policy.

In order to enforce the protection called for in the security policy, the MLS LAN solution has to provide a couple of principle guarantees. First, it must be able to maintain absolute control over the mechanism that provides the data to the users. This mechanism, like a security guard on a vault, cannot be by-passed and must be absolutely trustworthy to enforce the rules given by the organization that hired him. In a computer system this means that all of the code used in the development of the protection mechanism, and by extension, the rest of the security related processes must be tied directly to the

enforcement of the security policy. As will be seen, the accepted standard evaluation criterion requires the use of formal and informal models of the security policy to create this association. In addition the finished processes must be shown to be free of malicious or un-validated code. This is no easy task, as the all code must be validated as it is written or modified to ensure that the development team has accurately designed the process to meet all of the functional, as well as assurance requirements. More importantly however, the code must be verified that it does these functions correctly, without subversive entries such as "back doors" or Trojan Horses that could later be used to undermine the system's security." Second, the MLS networking solution must verifiably ensure the identity and coinciding security factors associated with each user accessing the network. The solution must provide the user with the assurance that he is, in fact, connecting to the authentic network information he needs. The ability to distinctly identify both users and information facilitates the protection mechanism's ability to control access to both the network and protect the organization's information from uninvited users.

The MLS LAN Project was developed to provide a trusted network system that is both necessary and sufficient to satisfy the above requirements and allow for independent evaluation under an accepted standard criterion. Currently, the TCSEC is the Department of Defense's principle "metric with which to evaluate the degree of trust that can be placed in a computer system for the secure processing of classified and other sensitive information" [Ref. 3]. Ratings for computer systems are broken down into four divisions, each with internal "Evaluation Class" ratings.

Division D describes computer systems that fail to meet the security requirements for any of the higher evaluation classes. There are no classes in Division D. Division C has two classes and introduces the concept of using a Trusted Computing Base (TCB) to enforce "Discretionary Access Control" (DAC). A Trusted Computing Base is an abstraction for the collection of elements of a computer system that pertain to the organization's security rules or policy. Its aegis encompasses all security-relevant aspects of the system, for example, policy enforcement mechanisms, any auditing (retrieval and analysis), identification and authentication, and the interface for security administration. The introduction of DAC allows the system to separate users from information on a discretionary "need-to-know" basis.

Division B has three evaluation classes and, in addition to DAC policy enforcement, introduces the concept of Mandatory Access Control (MAC). Mandatory access control is designed to maintain separation between different security levels of the accessing agent or "subjects" and the files or data to be accessed, known as "objects". To accomplish this, "Sensitivity Labels" are assigned to each subject as they are added to the system and objects as they are created. Disclosure of information to a subject is granted based upon a comparison between the subject and object sensitivity labels, e.g., the subject's sensitivity label must be equal to or greater than that of the requested object [Ref. 6].

Division B also introduces the requirement for a clearly defined and documented security policy model. The security policy model states the policy to be applied using either an informal statement (Informal Model) or formal language with proven assertions (Formal Model). Previous evaluation Divisions required only a statement of the

manufacturer's "philosophy of protection" and how this would be applied to the TCB. Another important inclusion is the need to satisfy the requirements of the "Reference Monitor" Concept. "The reference monitor enforces security by forcing all subjects (e.g., processes and users) who wish to access an object (e.g., files or portions of memory) to do so only through the monitor itself. Thus it *monitors* all *references* to objects by subjects"[Ref. 6]. A depiction of the reference monitor concept is given in figure 1.1. The monitor based on a set of rules governing access grants the right to use objects. The key, however, to the successful use of the reference monitor is that its design must follow three specific principles:

- It "must be tamperproof".
- It "must always be invoked".
- It "must be small enough to be subject to analysis and tests, the completeness of which can be assured". [Ref 6.]

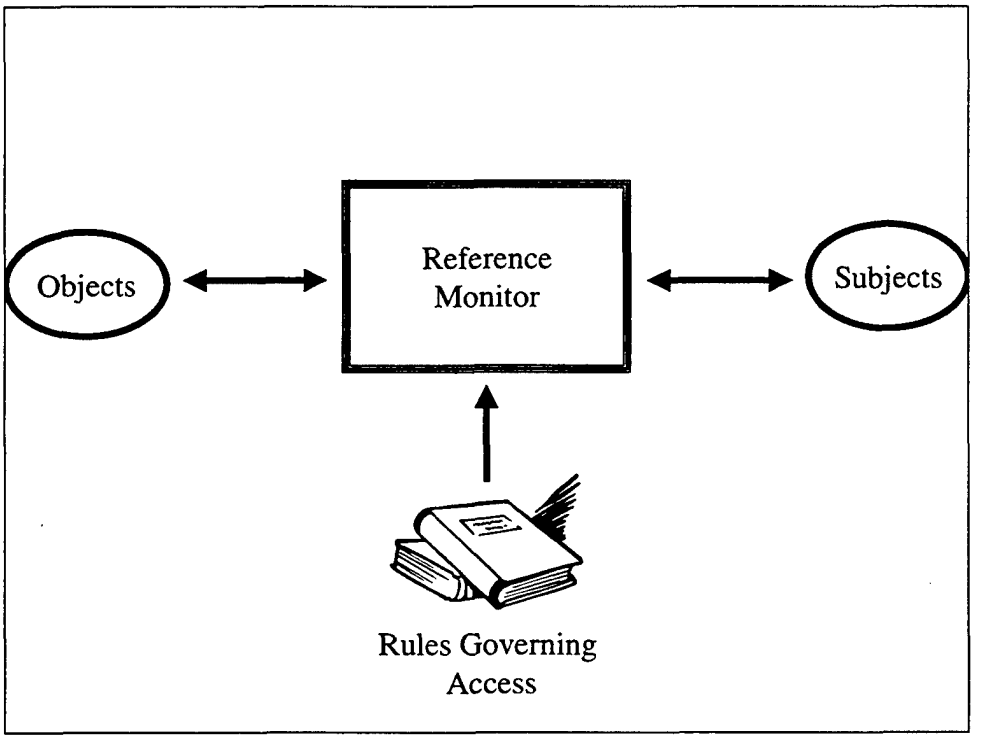

Figure 1.1 The Reference Monitor Concept

Another significant difference between Division B and Division C is the requirement to establish a "Trusted Path" between the user and the Trusted Computing Base. This requirement ensures that "communications via this trusted path shall be activated exclusively by a user of the TCB and shall be logically isolated and unmistakably distinguishable from other paths" [Ref. 3.]. The purpose for this is to assure both the TCB and user that they are communicating with each other through an "isolated and distinguishable" path. The trusted path can then be safely used for authentication operations, session renegotiation, or any other security related operations needed between the user and the TCB.

The highest rating provided for by the TCSEC is Division A, which has only one evaluation Class, A1. The functional requirements for Class A1 rated systems are equivalent to Class B3; however, these systems must undergo a much more rigorous and extensive regime of formal design specifications, proofs, and verification [Ref. 6]. The MLS LAN Project solution is to be designed to satisfy the requirements for a TCSEC Class B3 rating. The TCSEC was the first criteria developed to directly address the specific security features, and assurance requirements for multilevel systems, however it does not specifically extend to networks. In 1987, the National Computer Security Center published the Trusted Network Interpretation (TNI) to provide this association [Ref. 7]. This thesis will use each of these documents as the basis for its descriptive overview of the MLS LAN's system security, assurance, communications integrity and transmission security features.

3. MLS LAN Project Goals

The MLS LAN Project is an effort to provide government and commercial organizations with a cost effective, multilevel networking solution by leveraging existing high assurance technology. The ultimate goal of the project is to demonstrate a prototype network design that offers the ability to provide concurrent high assurance access for network users to data at multiple sensitivity levels through the incorporation of inexpensive commercial personal computers and software. The intended design of the network is to integrate the security features of a previously evaluated Class B3 high assurance server, the Wang Government Services Incorporated XTS-300™, with the conveniences of up-to-date operating systems and the latest commercial office automation software. The current plan for the MS LAN network architecture is to provide this functionality using the universally accepted TCP/IP protocol suite to allow our multilevel networking functionality to be layered on top of any chosen technology used in the lower layers of the OSI model. When completed, the MLS LAN will provide a cost effective multilevel solution within an easy-to-use office environment.

4. Thesis Goals

The MLS LAN is comprised of multiple components; each providing essential functions to ensure the network maintains absolute control over all accesses to its data, information, and services. Additionally, the LAN must provide verifiable protection against disclosure and modification of information during its transmission on the network's communications channels. To accomplish these objectives two things must be completed. First, the components of the MLS LAN must be described with respect to their design requirements and their incorporation into the proposed architecture. Second,

the method with which these components communicate with each other must be chosen in light of security and purpose. This thesis will describe, through a high level overview, each of the current MLS LAN components; their functionality; and the rationale behind the requirements assigned to them in the MLS LAN Project System Requirements Document [Appendix A]. This thesis will also establish a communications framework for these components as they provide network functionality to the users. The thesis will study the connectivity requirements as outlined in the MLS LAN Project Protocol High Level Analysis Document [Appendix B] and propose solutions for each.

B. CHAPTER OVERVIEW

1. Introduction

Chapter I discusses the purpose and goals of this thesis in the context of the problem that the MLS LAN Project has addressed and describes the how the project, in its entirety, proposes a solution. This chapter also provides an overview of the following chapters and appendixes: Chapter II – The MLS LAN Systems Architecture; Chapter III – Protected Communications Channel Security; Chapter IV – Overview of the MLS LAN Connection Framework; Chapter V – Conclusions and Recommendations; Appendix A – The MLS LAN Systems Requirements Document; Appendix B – The MLS LAN Protocol High Level Analysis; The MLS LAN Connection Framework Document. Each of these Chapters is sketched below.

2. The MLS LAN Systems Architecture

The MLS LAN is comprised of three primary components as outlined in Appendix A. The principle component is the network *Trusted Computing Base (TCB)*, which provides a penetration resistant security perimeter for MLS LAN operations. The

TCB is partitioned among the MLS LAN components to ensure the network as a whole enforces the overall network security policy. The *Network Application Protocol Services* provide functionality for access to available software, file transfer, electronic mail, or remote printing. Finally, the MLS LAN requires a *network computer or workstation* that can be employed by the user to access MLS LAN resources and functionality [Appendix A]. Chapter II will describe the makeup of each of these components, their functionality and how the network as a whole is constructed.

3. Protected Communications Channel Security

MLS LAN is required to protect all communications channels used by the network against disclosure and modification of the information transmitted. This is accomplished through the use of a protected communications channel established by the TCB. There are several options for the logical placement of the encryption mechanism that secures this channel. Chapter III will provide an overview of these options and evaluate their applicability for use in the MLS LAN Project.

4. Overview of the MLS LAN Connection Framework

The MLS LAN connection framework provides an overview of the parameters for initiation, security and communications establishment between two or more components of the MLS LAN. Chapter IV will describe the processing involved with each connection protocol used to establish a single-level session and to conduct operations on the LAN. The description will provide an overview of each connection in terms of the data required by each component, the data structures required for transmission and the usable states and transitions required for data transfer.

5. Conclusions and Recommendations

Chapter V contains the conclusions made for the use of the proposed architecture and connection framework as defined in the thesis. This chapter will also make recommendations pertaining to future research for aspects of the MLS LAN Project.

C. APPENDIX OVERVIEW

1. Appendix A: The NPS CISR MLS LAN System Requirements Document

The bulk of the research into the definition of the MLS LAN's architecture and its components was conducted as an engineering team effort. The appendix is the result of a collaboration to define the true requirements and functionality required of the MLS LAN Project.

2. Appendix B: The NPS CISR MLS LAN Protocol Requirements Document

As with the Systems Requirement Document, this appendix was also developed by an engineering team. This document outlines the requirements levied on each of the connection protocols of the MLS LAN.

3. Appendix C: The MLS LAN Connection Protocol Framework

The connection framework appendix provides a descriptive overview of the datagram format and packaging, as well as the state options and transitions for each protocol used in MLS LAN connection.

THIS PAGE INTENTIONALLY LEFT BLANK

II. NPS MLS LAN SYSTEMS ARCHITECTURE

A. THE MLS LAN PROJECT ARCHITECTURE OVERVIEW

1. System Definition and Accreditation

The purpose of the MLS LAN Project is to design a trusted network system. A network system is the "entire collection of hardware, firmware, and software necessary to provide a desired functionality" [Ref. 7]. This Chapter is not intended to define the entire network system, but to provide an overview of the major components that comprise the MLS LAN architecture. The Trusted Network Interpretation (TNI) defines a component as, "any part of a system that, taken by itself, provides all or a portion of the total functionality required of the system. A component is recursively defined to be an individual unit, not useful to further subdivide, or a collection of components up to and including the entire system". [Ref. 7] This view of system components is germane to the architectural overview because of the way the MLS LAN envisions its accreditation and evaluation.

There are two predominant views for how a trusted network system can be evaluated. The first looks at the policy enforcement provided by the trusted network components as a single entity. The network implements a reference monitor and has a single "Network Trusted Computing Base" (NTCB). A single accrediting authority then generally accredits the entire system. The second view, known as the "Interconnected Accredited AIS View", is more distributed in nature. It "recognizes that parts of the network may be independently created, managed, and accredited" [Ref. 7]. An interconnected network system consists of "multiple systems (some of which may be trusted) that have been independently assigned operational sensitivity ranges (the highest

and lowest sensitivity levels of information that may be simultaneously processed on that system). In this view, each network system is individually accredited to handle sensitive information at either a single level or over a range of multiple levels" [Ref. 7]. The MLS LAN Project intends to use the interconnected accreditation process to facilitate the evaluation of its modular design and to enable individual accreditation of the MLS LAN regardless of its future connectivity to other secure networks such as the DoD's Secure Internet Protocol Routed Network (SIPRNET).

The evaluation criterion for a trusted network system requires a statement of the security policy that is enforced. In addition, a Class B3 system must provide a formal Security Policy Model, which proves the assertion that the TCB and its implemented reference validation mechanisms correctly enforce the system's security policy [Ref. 4]. The MLS LAN incorporates two such security models, one for non-disclosure or secrecy and another for non-contamination or information integrity. The following subsections outline these models.

a. The Bell and LaPadula (BLP) Model

The Bell and LaPadula Model [Ref. 8] is a mathematical model describing the allowable paths for information flow in a secure system where it is important to maintain secrecy [Ref. 9]. The model uses the concept of a finite-state machines to define the security requirements for computer systems to concurrently handle data at different sensitivity levels. This is useful in systems where a machine may be required to handle, for example, both Top Secret and Confidential information at the same time. The BLP model describes the allowable communications in the system which prevent programs processing top secret data from leaking their information into the confidential data and

prevents the confidential users from accessing the top secret data. This model has been adopted by most DoD MLS systems and provides an abstract formal treatment of what is known as the military security policy [Ref. 9].

The components defined in the BLP model consist first of a set of subjects *S*. The term subject is used to describe an active entity in a computer system, such as users, processes or executable programs. The next component is a set of objects *O*. The term object refers to the passive entities in a computer system, such as files, directories, or databases. The third component is a set of modes of access *A* (e.g., *read, write, execute, append*)[1]. The final component is the set of security levels *L*.

The term "dominance", characterized by the symbol ≥, is "used to limit the sensitivity and content of information a subject can access" [Ref. 9]. It can be said that *o dominates s* ($o \geq s$) if the hierarchical security rank assigned to *o* is at least as high as that of *s*. For instance, Secret dominates Unclassified because, using the DoD hierarchical classification structure, a Secret Security level is higher than an Unclassified Security level. Under the BLP model, therefore, a state is considered secure if for each triple consisting of ($s \in S$, $o \in O$, $a \in A$), the following two properties are satisfied [Ref. 6]:

- The "*Simple Security Property*" or "*no-read up property*" – This property is used to prevent a subject from reading an object when the security level assigned to the subject does not dominate the security level of the object, e.g.,
  read permitted iff $s^L \geq o^L$

[1] It should be noted that in this context I use the notion that "read" and "execute" denote a read-only, an "append" denotes a write-only and a "write" denote a capability to read and write.

"In the military model, this property says that the security class (clearance) of someone receiving a piece of information must be at least as high as the [security] class (classification) of the information [Ref. 9].

- The (Confinement or Star) "* *Property*" or "*no-write down property*" – This property is used to prevent a subject from writing to an object when the security level assigned to the object does not dominate the security level of the subject, e.g.,
  write permitted iff $o^L \geq s^L$

In the military model, the * property prevents a user operating in a Secret Session from writing a document that is classified Confidential.

b. *The Biba Integrity Model*

The Biba model [Ref. 10] is intended to address the control of information flow with respect to data integrity or non-contamination. The Biba model introduced two basic properties that are very similar to the BLP model, however its perspective is orthogonal to that of the BLP model rules.

The components defined in the Biba model also consist of a set of subjects *S*, a set of objects *O*, and a set of modes of access *A* (*read, write, execute, append*). However, instead of levels of security, Biba uses the set of integrity levels *L*. Integrity is maintained if for each triple consisting of ($s \in S$, $o \in O$, $a \in A$), the following two properties are satisfied [Ref. 6]:

- The "*Simple Integrity Property*" or "*no-write up property*" – This property is used to prevent a subject from writing to an object when the integrity level assigned to the subject does not dominate the integrity level of the object, e.g.,
  write permitted iff $s^L \geq o^L$

The basic purpose of the simple integrity property is to prevent a low integrity, or unreliable subjects from modifying high integrity objects.

- The (Integrity Confinement) "* *Integrity Property*" or "*no-read down property*" – This property is used to prevent a subject from reading an object when the integrity level assigned to the object does not dominate the integrity level of the subject, e.g.,

  read permitted iff $o^L \geq s^L$

The integrity property prevents a high integrity, or reliable subject from accessing a low integrity or unreliable object. This ensures that highly reliable subjects run only highly reliable software.

The two models, BLP addressing inappropriate disclosure, and Biba addressing integrity, are used in conjunction to provide an appropriate dominance relationship for the MLS LAN and can be used as a joint enforcement mechanism for both secrecy and integrity throughout the network.

2. Component Description

The MLS LAN is comprised of three components. The principle component is the Trusted Computing Base (TCB), which provides a penetration resistant security enforcement mechanism for MLS LAN operations. The second component, the Network Application Protocol Services provides the functionality required for network access to available application software, file transfer, electronic mail, or remote printing. Finally, the MLS LAN will use a network computer or workstation that can be employed by the user to access any required network functionality. The components of the MLS LAN are depicted in figure 2.1.

a. The Trusted Computing Base

The Trusted Computing Base (TCB) in a network, like a stand-alone system, consists of all of the security-relevant portions of the network. But, unlike the stand-alone system, a network configuration may distribute the security mechanisms to

various components in the system. This distribution is referred to in [Ref. 7] as a "Partitioned Network TCB". The MLS LAN TCB components are presently built upon the Wang Government Services, Inc. XTS-300™ systems architecture. This systems architecture affords the XTS-300 security kernel complete control of the

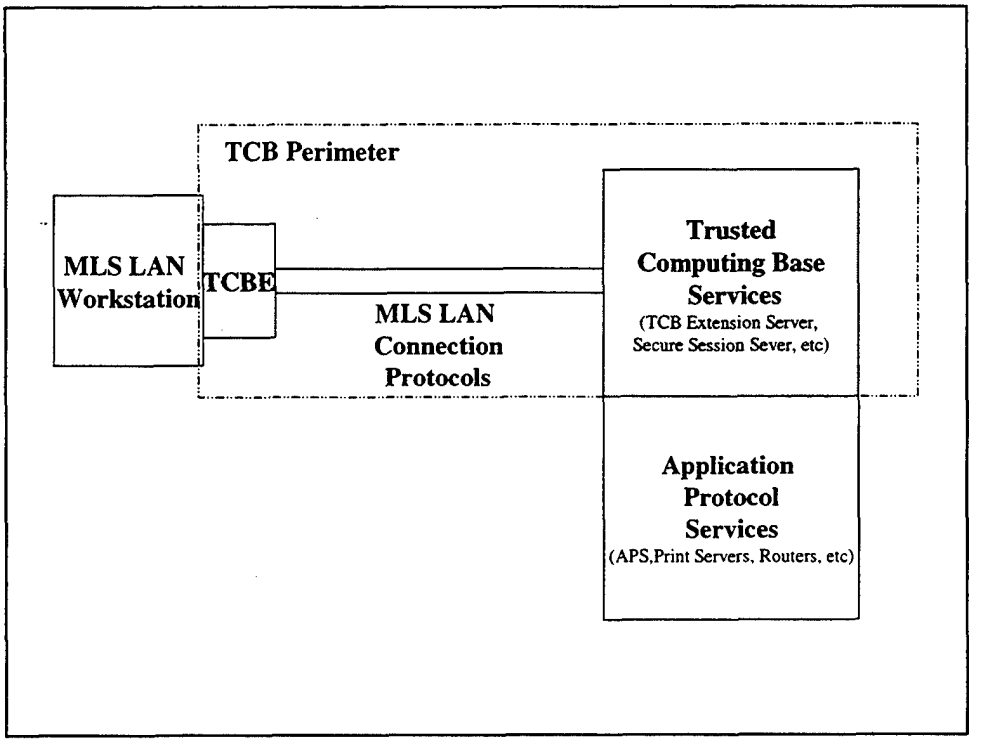

Figure 2.1 MLS LAN Components

MLS LAN trusted user-developed code. This user-developed code is installed to extend the TCB to the workstations, create secure session application connections, and protect communications. The XTS-300 uses a four-ring structure or hardware abstraction, with each ring defining a level or domain in which a process can execute. Protection during process execution is afforded through the ring structure by isolating the security domains in hardware thus preventing system processes from tampering with each other. The XTS-300 defines these domains as four primary software components: The Security

Kernel, Trusted System Services (TSS), Trusted Software and Commodity Application System Services (CASS) and Untrusted Applications.

Ring 0, the Security Kernel domain, is the most privileged. It contains the Reference Validation Mechanism and provides basic operating system services such as MAC and DAC policy enforcement for process and device objects, resource management, process handling, and interrupt handling. Ring 1, the Trusted System Services domain is controlled by the security kernel and provides "networking, I/O, file system management, and file system object discretionary access policy enforcement for both trusted and untrusted processes" [Ref. 11]. Ring 2, Trusted Software and CASS, is shared by the trusted software such as the STOP operating system or user-developed trusted code and the untrusted CASS. Trusted software functions allow system operators and administrators to perform security related housekeeping or other privileged tasks not supported by the STOP components. Ring 3, Application Domain, is reserved for user processes and is the least privileged. An abstract depiction of the XTS-300 architecture is provided in figure 2.2 [Ref. 11].

The XTS-300 supports many of the MLS LAN TCB requirements outlined in Appendix A, such as Secure Attention Key (SAK) recognition and processing, user access identification and authentication (I & A), session control and TCP/IP configuration management [Ref. 12]. The MLS LAN trusted processes that reside in Ring 2 provide for MLS LAN specific procedures such as the extension of the TCB to the TCBE and the provision of communications protection. These trusted processes, which are controlled by the XTS-300 hardware and software (Rings 0 &1), the TCB Extension

hardware, and the protocols defined for connecting two MLS LAN components comprise the subcomponents of the MLS LAN TCB.

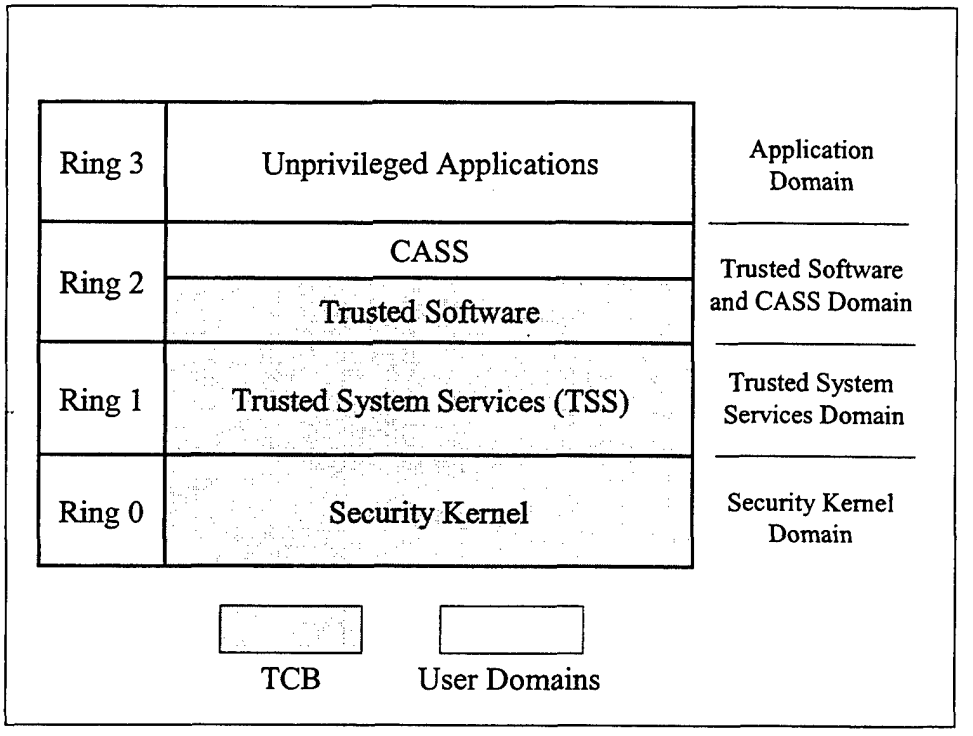

Figure 2.2 XTS-300 System Architecture

(1) Protected Channel Initiator. This trusted process is responsible for the creation of the Protected Communications Channel (PCC) between two MLS LAN components. The initiator process will enforce a "two-way" mutual hardware authentication between the two connecting entities and provide security and integrity protection on all transmitted data. The Protected Communications Channel provides the secure conduit through which all other connection protocols operate and provides the basis for extending the TCB from the XTS-300 to the distributed components, e.g., TCBE or other source hosts. Effectively, these protected extensions allow us to view the distributed TCB as one logical TCB, from a security perspective. The use of this channel also provides fault tolerance protection in the event of component loss, as the communications between the two PCC connected entities will cease, but the overall network will not be affected. The protocol framework for this channel is discussed in Chapter IV, however the design of the Protected Channel Initiator process is left to future work.

(2) Session Database Server. The Session Database Server is a trusted process that manages the session status data for each user logged into the MLS LAN. Session status modification requests are permitted only from the TCB Extension Server. These requests are made using a specified Session Status Protocol. Other TCB entities may query the information in the database, using a this protocol, however no "write" or modification access is granted. The query allows session servers or other entities to receive a listing of the current session information on a user. Currently the database is maintained on a single XTS-300 source host, however in the future, this server could provide the database synchronization required to incorporate a distributed implementation of the database. The loss of communications between the TCB Extension Server and the SDS could allow unwarranted access to the MLS LAN. To prevent an insecurity, the MLS LAN requires some control mechanism that could prevent new connections to the MLS LAN and its services in this event. The development of this mechanism is left to future work.

(3) Trusted Computing Base Extension Server. The TCB Extension Server process was previously developed by the Naval Postgraduate School. Its purpose is to extend the TCB perimeter securely over the network to the requesting TCBE-equipped workstation. This process will be initiated only through the request for "secure attention" from a user. The Extension Server process is comprised of a single parent and multiple child processes that are responsible for accepting connections from the TCBE-equipped client workstations. The parent process will initially listen on an assigned port for incoming requests for secure attention. Once a request is received, the parent process will verify the identification and authentication of the requesting TCBE.

If the verification is successful, a child process is forked and the parent is able to relinquish control of the communications to the child. This frees the parent to listen for new connection requests. If the I & A is in error, the connection is terminated and no child is created [Ref. 13].

Each TCBE connection to the MLS LAN is therefore assigned an individual child TCB Extension Server process that will handle all of the security related operations necessary to establish and maintain a session on the MLS LAN. The current MLS LAN design enables the child process to present the user with menus, with which they may conduct all trusted path security-related operations such as "login" and "session negotiation". This process also controls the actions of the connected TCBE through specific TCBE state commands. The options, commands, and transitions used in this interaction are discussed in TCB-to-TCBE Connection Protocol section of Appendix C. At any time, the user may activate the Secure Attention Key (SAK) which will prompt the TCB Extension Server to interrupt the current running processes, verify the TCBE, and begin the user login or session negotiation process. The TCB Extension Server interaction is depicted in figure 2.3.

A design consideration discussed during the development of the MLS LAN system architecture was the preservation of the trusted path connection between the TCB Extension Server and the TCBE-equipped workstation. Can this connection be terminated following session negotiation or must it be maintained throughout the lifetime of the user's connection to the MLS LAN? The answer rests upon the responsibilities of the TCB. The Extension Server is required to update the TCB on all connection and sessions established on the LAN. In essence, it maintains the "fail-

secure" [Ref. 14] properties of the MLS LAN's distributed TCB by ensuring that information used by TCB entities to establish connections is current and correct. As mentioned previously, the Session Database Server maintains this information, but the Extension Server exclusively controls modification of the database. The Extension Server will modify the database upon initialization of a user session, a session change, a user logout or TCBE disconnection from the LAN. This methodology ensures that the session database is the current depiction of the MLS LAN. From this example, it is obvious that the Extension Server – TCBE trusted path must either be maintained following the initial session establishment to support session changes or that the path must be reestablished to effect changes.

During normal LAN operations, there seems to be no requirement for the Extension Server – TCBE trusted path. The user has set his session and is operating normally. The database is current and only a renegotiation with the Extension Server will change it. Application protocol requests from the user cause the application servers to query the information maintained by the Session Database Server. The information returned from the query enables the application protocol requests to be validated against the TCB's trusted session information. If a request is not commensurate with the user's current session, the Secure Session Server will deny access rather than compromise the system. Session level modifications are conducted simply by activating the SAK and reestablishing the Extension Server – TCBE trusted path to change session levels.

One of the protection mechanisms, however, sought for the MLS LAN is the ability of the TCB to maintain control over the user's LAN connection. The

intent is to enable the TCB to confirm that the user is actually still physically there. This future requirement presents an issue with respect to normal LAN operations and the use of the TCBE-to-TCB connection. With no continuous connection between the TCBE and the TCB Extension Server or a mechanism to limit the time that a user may remain in a session without the physical activation of the Secure Attention Key, how does the TCB know the user is still there? The solution to this issue is beyond the scope of this document and will be left to future work.

(4) Trusted Computing Base Extension. The TCBE is an enhanced network interface card (NIC) that is installed into the MLS LAN workstation to support a trusted path interface to the user. The current test platform has been prototyped utilizing the Intel™ i960 processor. The TCBE provides the MLS LAN with a verifiable high assurance entity that can be used to extend the TCB. It provides the user with the Secure Attention Key mechanism for Trusted Path initiation and will provide communications protection, through the establishment of a Protected Communications Channel, to components of the MLS LAN. The TCBE, through state commands from the TCB Extension Server controls the disk operating system and applications used on the workstation. Additionally, the TCBE ensures appropriate object reuse between session security levels.

(5) MLS LAN Connection Protocols. The TCB utilizes a number of specific connection protocols to establish a session and conduct operations on the MLS LAN. The most fundamental of these is the Protected Communications Channel (PCC) Protocol. The PCC protocol is used to establish the security conduit through which all other MLS LAN protocols must operate. Once the PCC is established, the TCBE

must "connect" to the TCB Extension Server for login and session negotiation. The TCB-to-TCBE Protocol is used for this purpose. During the session negotiation, the TCB Extension Server updates the user session information through the Session Database Server to reflect the parameters of his current session. The Session Status Protocol supports these operations. Once a session

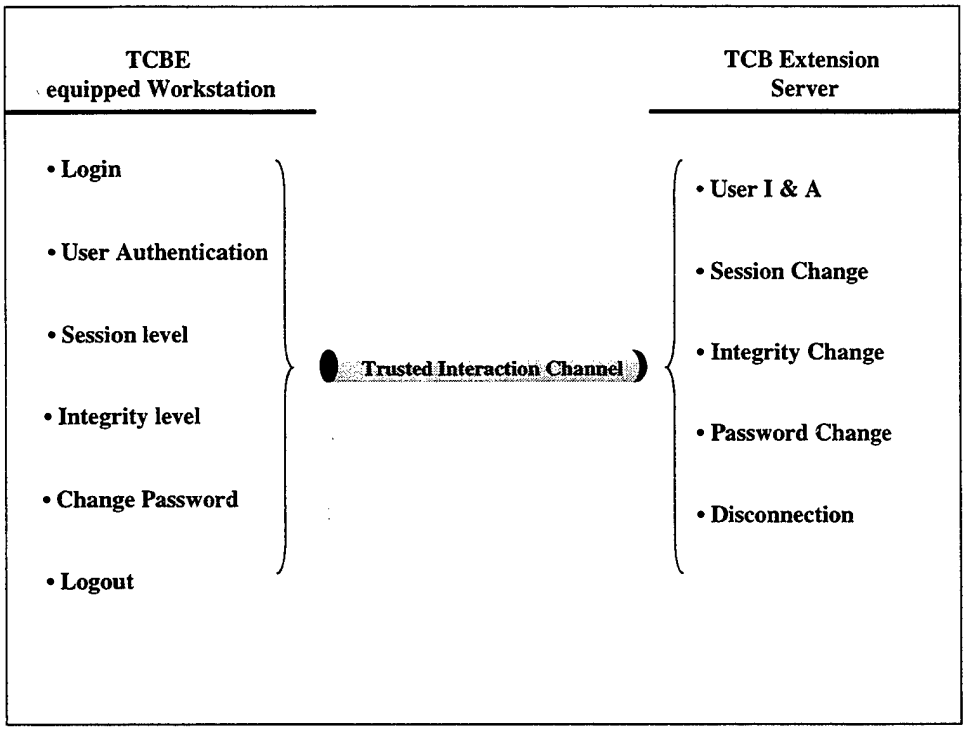

Figure 2.3 TCB Extension Server Interactions

is established, the user may require connectivity with a MLS LAN Application Protocol Server. These operations are conducted through the Secure Session Server. The Session Server Protocol supports requests for application protocol services. Prior to the Secure Session Server fulfilling the user's application protocol request, a listing is requested of the Session Database Server to verify the user's current session information. The Secure Session Server uses the Session Status Protocol to make this query. In the future there may be additional protocols defined for the MLS LAN to provide services to

workstations not utilizing a TCBE, however, these are not currently part of the framework. A depiction of the expected protocol usage is provided in figure 2.4. An overview of each of these protocols will be provided in Chapter IV and with a detailed description contained in Appendix C.

b. MLS LAN Network Application Protocol Services

The MLS LAN is designed to support the use of multiple simultaneous accesses to higher layer protocol services, such as HTTP, IMAP or FTP. The access to this information is controlled through the TCB in accordance with the security policy. A trusted process known as the Secure Session Server, validates and creates the connection. The Application Protocol Server (APS) is an untrusted application layer process that provides the service. These two subcomponents comprise the Network Application Protocol Services.

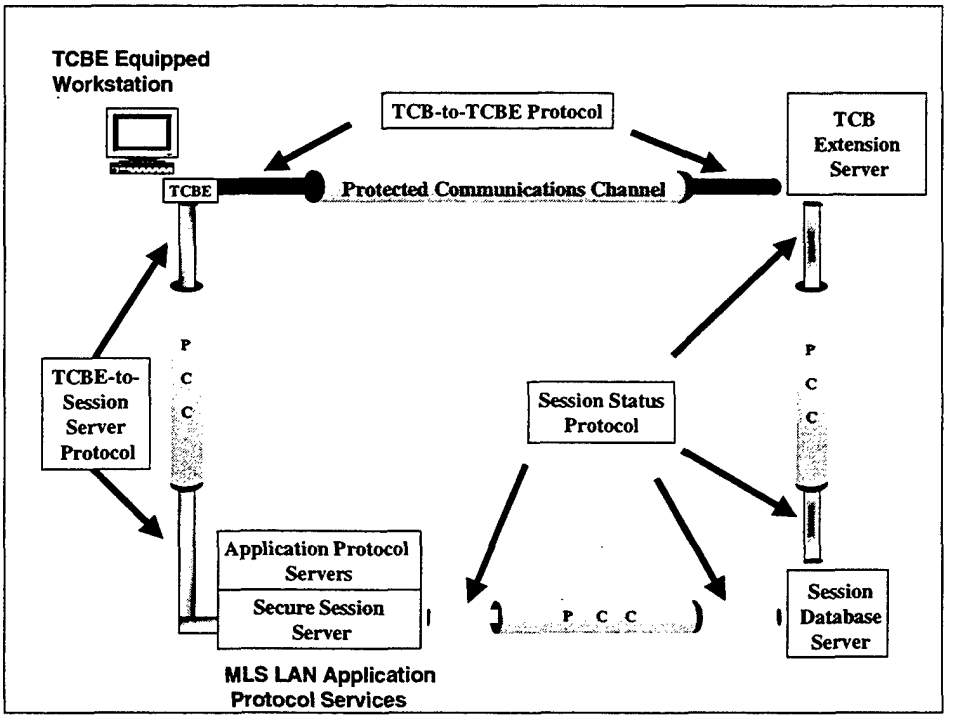

Figure 2.4 MLS LAN Connection Protocols

(1) Secure Session Server. The Secure Session Server process is comprised of a single parent and multiple child processes for each platform on which a given application protocol is hosted. These trusted processes reside in the Trusted Software portion of the XTS-300 architecture and are controlled by the Security Kernel. The Secure Session Server parent process is responsible for accepting connections from TCBE-equipped client workstations and establishing the TCP/IP protocol service for the user. The parent process will initially listen on an assigned port for incoming requests for protocol service. Once a request is received, the parent process will verify the user's MLS LAN session with the Session Database Server. If the verification is successful, a child process is forked and the parent is able to relinquish control of the communications to the child. This frees the parent to listen for new connection requests. If the database query is in error, the connection is terminated and no child is created [Ref.13].

Each protocol service request is therefore assigned an individual child Secure Session Server process that will handle all of the protocol transmissions to and from the APS. The child process is responsible for the creation of a unique Application Protocol Server process tied directly to the user through a handle created from the session data received from the Session Database Server (user name, session level). A depiction of the Secure Session Server/Application Protocol Server interaction is provided in figure 2.5.

(2) Application Protocol Server. The Application Protocol Server process is responsible for implementing the server portion of the application level protocol. This process will support only a single protocol and is untrusted with respect to the data stored on the server. The source code for these processes is intended to be an

implementation of the industry standard application protocol, with minor modifications where necessary for MLS LAN integration. Communications between the client workstation and the APS will be maintained exclusively through the Secure Session Server and are constrained by the underlying TCB [Ref. 13].

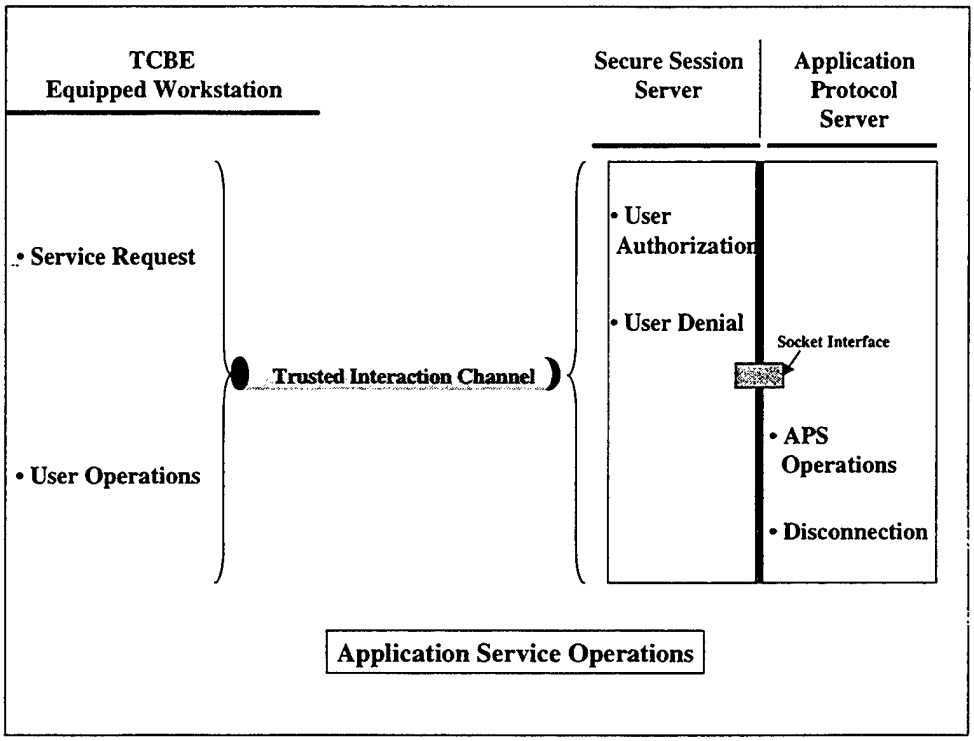

Figure 2.5 Secure Session Server / Application Protocol Server Interaction

c. MLS LAN Workstation

The MLS LAN client workstation is designed to be a commercially procured "thin client" diskless workstation. The workstation will operate under the control of the TCBE. Each workstation will support no more than one logged in user at a time. The workstation will support up-to-date commercial operating systems and application software. A future requirement for the MLS LAN will allow non-TCBE equipped workstations to connect to the LAN. This would permit "anonymous" access to selected application services.

III. PROTECTED COMMUNICATIONS CHANNEL SECURITY

A. OVERVIEW

The MLS LAN TCB is required to "provide protection against disclosure and modification of information on all communications channels used by the network" [Appendix A]. To accomplish this, digital communications encryption will be used. Generally, there are two common approaches used to provide this capability: Link Encryption and End-to-End Encryption.

Link encryption takes place at the physical layer of the Open Systems Interface (OSI) model through the use of special encryption devices connected at the point where the physical media exits each node. This technique would require that each MLS LAN workstation and source host be equipped with an additional encryption hardware device and symmetric encryption key. This, of course, would place a significant number of additional burdens on the TCB. The most significant of these is the management and dissemination of the appropriate keying material for these devices. How would the Security Manager change the device key at both the client workstation and source host each time a user changes his session level? Does each source host require an encryption device for each workstation connected to the LAN? Because of these issues, link encryption cannot be considered a viable option for Protected Communications Channel (PCC) implementation.

End-to-End encryption utilizes the higher layers of the OSI model to provide protection and therefore there are several options for the logical placement of the encryption. One method of encryption is to allow each individual application to apply its own security protection. This is known as Application-Level Security. With application-

level security only the user data portion of the TCP segment is encrypted and unfortunately requires the Commercial Off The Shelf (COTS) applications to be equipped with an encryption capability or to modify the application for this purpose. The use of application level encryption is insufficient for the MLS LAN as there is no way to enforce the requirements implicit in the reference monitor concept. Additionally, to require the MLS LAN to modify each application for appropriate security protection defeats the intended goal of the MLS LAN project [Ref. 15].

This leaves two other options for the logical placement of the encryption protection for the PCC: the Transport Layer or the Network Layer. Each of these OSI layers has a standard security protocol defined by the Internet Engineering Task Force (IETF). This chapter will provide an overview of these two protocols and evaluate their applicability for use in the MLS LAN.

B. TRANSPORT LAYER SECURITY PROTOCOL

The Transport Layer Security (TLS) protocol was designed to make use of the Transport Control Protocol (TCP) to provide privacy and data integrity on end-to-end communications between two client/server applications. TLS was originated by Netscape as the Secure Socket Layer (SSL) protocol and published as an Internet draft document. Subsequently the ITEF formed a working group to produce an Internet Standard that became Request For Comment (RFC) 2246, the TLS Protocol Version 1.0 [Ref. 16]. Currently, the most use of transport layer security is in the World Wide Web client/server transfer service provided by the hypertext transfer protocol (HTTP). Virtually all HTTP application clients and servers have been modified to recognize TLS,

however any end system can modify its higher layer protocol applications (e.g., FTP, IMAP, or SMTP) to incorporate TLS.

TLS introduces two new protocol layers above TCP to provide reliable end-to-end secure services as depicted in figure 3.1. These two layers allow independent programs to successfully exchange cryptographic parameters without knowledge of one another's code. The TLS protocol is written such that "the decisions on how to initiate TLS handshaking and how to interpret the authentication certificates exchanged are left up to the judgment of the designers and implementers of protocols which run on top" [Ref. 16].

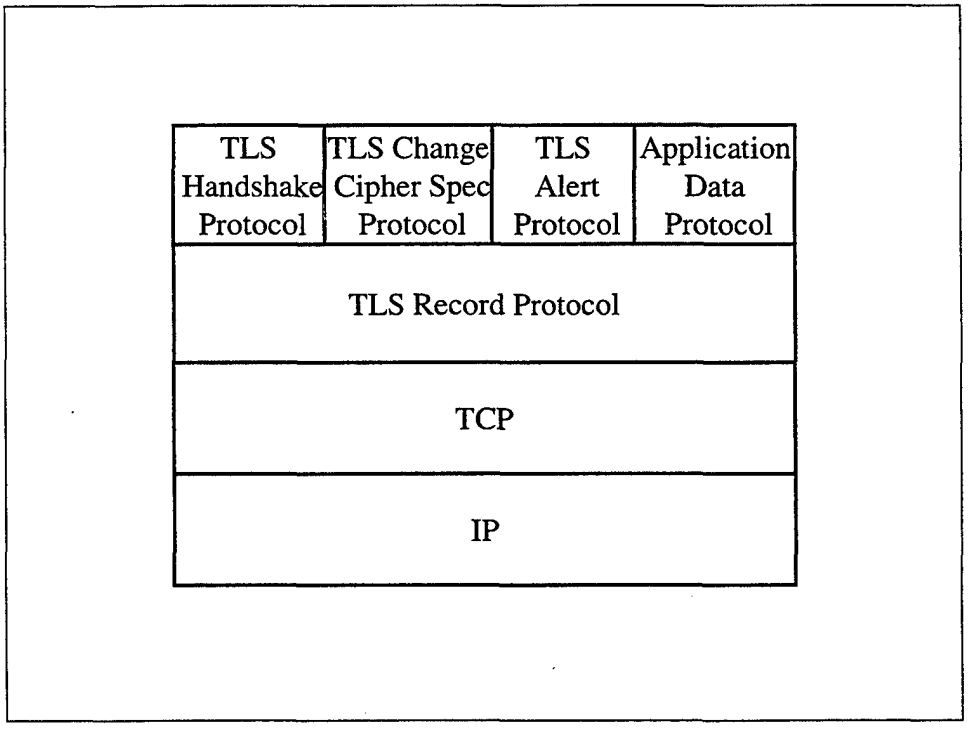

Figure 3.1 TLS Protocol Stack [Ref. 13]

The TLS Record Protocol layer provides higher layer protocols connection security that has two basic properties: confidentiality through the use of a negotiated symmetric key and reliability through the use of keyed Message Authentication Codes. To perform these functions, the Record Protocol Layer fragments, compresses, adds the

authentication code and encrypts the information from the four TLS defined higher layer record protocol clients [Ref. 15]. This process is depicted in figure 3.2.

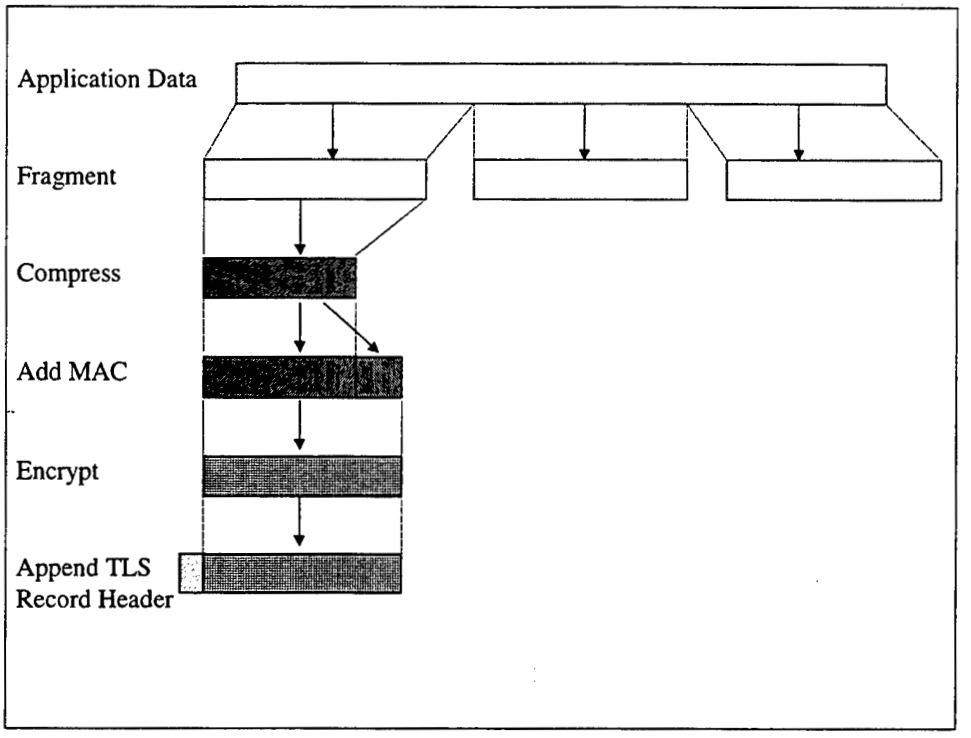

Figure 3.2 TLS Record Protocol Operation [Ref. 15]

The most complex of these higher layer protocols is the Handshake Protocol. It "consists of a suite of three sub-protocols which are used to allow peers to agree upon security parameters for the record layer, authenticate themselves, instantiate negotiated security parameters, and report error conditions to one another" [Ref. 16]. The other three upper layer protocols use the lower Record Protocol layer to pass application or control information between the client and server. The Change Cipher Spec Protocol consists of a single message, which causes the negotiated cipher suite to become the current encryption suite. The Alert Protocol conveys TLS-related alerts to the peer entity. If the alert is considered fatal, TLS will terminate the connection. Examples of alert messages are: Incorrect MAC, Bad Certificate, Certificate Expired, or a Handshake Failure.

"Servers and clients are required to forget any session-identifiers, keys, and secrets associated with a failed connection" [Ref.16].

The handshake protocol consists of a series of messages exchanged between the client and server that use public key cryptography or asymmetric algorithms to negotiate a symmetric master secret with which all other transmissions will be secured. Public key algorithms, invented by Witfield Diffie and Martin Hellman [Ref. 17] utilize two keys. A private key is created and protected by the owner, and a matching public key is published for others to use. When a message is encrypted with one of these keys it can only be decrypted by its matching key and since it is impossible to derive the private key from the public, the technique is considered to be computationally secure. An example of how the handshake messages establish the master secret is summarized as follows and depicted in figure 3.3:

1. Client/Server Hello Messages

The Client sends a "client_hello" message to which the server must respond with a "server_hello" message or a fatal error will occur and the connection will fail. The client_hello message contains:

- The client's TLS version number
- Cipher suite settings that the can be supported by the client.
- The requested session id if a previous session is to be used. If this is a new connection, this field is left blank.
- Compression methods supported by the client.
- A randomly generated value.

In response the server_hello message contains:

- The TLS version number that will be used.
- A specific cipher suite selected from the list provided by the client.
- The session id assigned to this connection.
- A specific compression method selected from the list provided by the client.

- A randomly generated value (different from the client's).

If the agreed-upon key method requires, the server will immediately follow the hello message with its public key certificate. Generally this will be an X.509v3 [Ref. 18] certificate. It must contain a key that corresponds to the key exchange algorithm selected or a fatal error will result. Following the server certificate, the server may send a "server_key_exchange" message. This message is sent only when the server certificate message does not contain enough data to allow the client to exchange a pre-master secret [Ref. 16]. The server_key_exchange message contains either an RSA [Ref. 19] or a Diffie-Hellman public key to encrypt the pre-master secret. The Diffie-Hellman key exchange provides a secure method to establish a shared secret between the parties of the exchange. The result of this exchange will be the pre-master key. The server can optionally request a certificate from the client by using the certificate request message.

Once the server has completed the above messages, it will send a "server_hello_done" message. This message conveys to the client that the server has passed all of the transactional information necessary to support the key exchange. After sending this message, the server will wait for a client response.

2. Key Generation Messages

The first message a client can send following the "server_hello_done" is the "client_certificate" message. If no suitable certificate is available, the client should send the message containing no certificates. If the server requires authentication, this may result in a fatal handshake error passed in an "alert" protocol message. The client will follow its certificate with the "client_key_exchange" message, which sets the pre-master key using either the RSA-encrypted secret or a Diffie-Hellman exchange. If the client

certificate has signing capability, the client will finalize the key exchange by explicitly verifying its certificate in a "client_certificate_verify" message.

The successful setting of the pre-master key and the authentication of the communicating peers will be followed by a "change_cipher_spec" protocol message. This message converts the new generated (pending) cipher specifications into the validated (current) encryption scheme. The client immediately sends a "finished" message using the new algorithms, keys and secrets. In response, the server will send its own "change_cipher_spec" message and "finished" message using the new encryption specifications [Ref. 16].

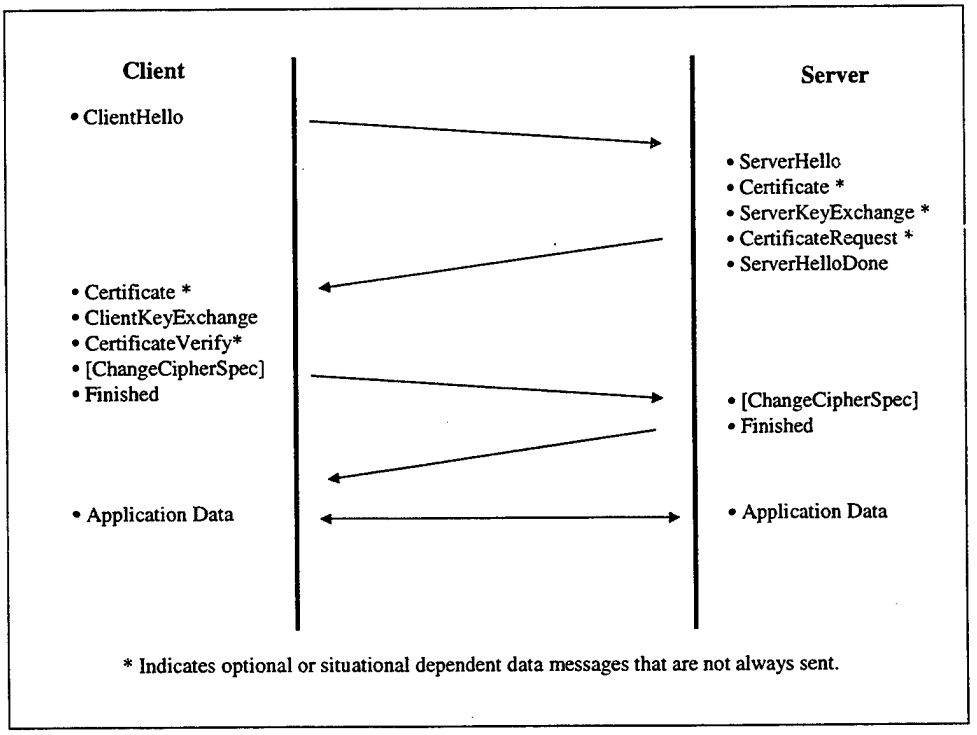

Figure 3.3 TLS Handshake Protocol Message Exchange [Ref. 16]

From this point on, application data may be passed to the lower layer of the Record Layer for secure transmission (see Figure 3.1). It must be noted that it is up to the higher layer application to be cognizant of the security requirements of their

transmissions, as TLS may not negotiate the strongest possible connection for their use. For example, the application must be aware that the security policy requires at least 3DES with a 1024 bit RSA key exchange to provide adequate protection for secret data. If the connecting server's highest encryption transform is DES, the application must recognize a security problem and terminate the connection. This adds significant complexity to the use of TLS in multilevel systems. Additionally, the client must specifically request that the server authenticate itself or the handshake protocol will skip this exchange [Ref. 15].

C. INTERNET PROTOCOL SECURITY

The Internet Protocol Security (IPSec) standard was designed to provide authentication, confidentiality and integrity across an untrusted network environment such as the Internet. IPSec operates in layer 3 or the Network layer of the OSI stack. The application of protection in the lower layers "reduces the explosion in the implementation of security protocols at the higher layer. If security is implemented at higher layers, each application has to design its own security mechanism" [Ref.20]. This flexibility allows IPSec to encapsulate "all and any kind of Internet traffic...[while allowing] per flow or per connection security" [Ref 20]. Unlike TLS, IPSec is not restricted to end systems, but can protect packets between two hosts, between network security gateways (e.g., routers and firewalls) or a combination of the two. This allows IPSec to individually handle each IP datagram based on the traffic to protect, the unique and appropriate encryption scheme for protection and to whom the traffic is to be delivered [Ref.20]. The overall architecture as outlined in [Ref. 22], defines three major components of the IPSec family. The first provides a method to represent and implement

the intended security policy. The second component includes the protocols that provide the confidentiality, authentication, and integrity to the IP packets and the third defines the key negotiation/management structure.

1. Security Policy

The mission of a security policy is to "ensure that network components support the basic principles of information security: protect information from unauthorized or accidental modification, destruction, and disclosure and ensure timely availability and usability of those data" [Ref. 23]. This is a bit broader definition than that of the Automated Security Policy presented in [Ref. 5], but the intent is provide a direct correlation between the information protection requirements and the mechanisms that provide the security. IPSec gives the user a "standard, robust, and extensible mechanism to provide security to IP and the upper layers (e.g. UDP or TCP) in direct support of the organization's unique security requirements" [Ref. 20]. This is accomplished through the use of a Security Policy Database (SPD).

Once the organization has determined which transmission links are to implement IPSec, a database is created to store this information. The Security Policy Database (SPD) is populated with attributes that can be extracted from the network and transport layer headers and used to determine the security services afforded to a packet. Each SPD entry has the following fields:

- Source Address
- Destination Address
- Name (This is a unique DNS name, X.500, or Distinguished Name used during key exchange negotiations)
- Protocol (e.g., FTP, HTTP, IMAP)
- Data Sensitivity Level

- Upper layer ports (This is the unique TCP port number assignment for the Upper layer protocol)

[Ref. 22]

The SPD entry uses this information as selectors to define one of three actions to take place for each packet:

- Discard the packet
- Bypass security on the packet - do not apply IPSec
- Apply IPSec to the packet. [Ref. 22]

The type of security services to be applied is designated using the concept of security associations (SA).

Security associations are essentially contracts between two communicating entities that outline the parameters required to securely transmit information. A SA is unidirectional and protocol-specific in nature. In other words, they describe the specific transmission state parameters (i.e., security protocol, transforms or encryption algorithms, key, key duration, etc.) that must be established from entity A to entity B in order to transmit securely. A separate and distinct SA must be defined to transmit from entity B back to entity A. SAs can be created manually through verbal or written agreements, or dynamically through an Internet standard key management protocol such as the Internet Key Exchange (IKE), provided by IPSec [Ref. 24]. Once an SA is created, a Security Parameter Index (SPI) is assigned which uniquely identifies the SA to the receiver. The SPI is a 32-bit identifier that accompanies the state information as it is entered into the host's Security Association Database (SADB).

An SADB is created for any entity that implements IPSec protocols. The SADB maintains all of the active SAs for both incoming and outgoing processing. The listed SAs in the database are indexed using the unique SPI and contain the parameters

previously negotiated to create the secure communications state between the two entities. These parameters include:

- Sequence number counter– a 32 bit field used to prevent replay attacks
- Sequence counter overflow – describes the action taken following an overflow
- Anti-replay Window – describes the size of the anti-replay sliding window.
- AH Authentication Algorithm – The algorithm and keys used in AH.
- ESP Encryption Algorithm – The encryption algorithm and keys used in ESP.
- ESP Authentication Algorithm - The algorithm and keys used in ESP.
- Lifetime – the duration of time that the SA is active.
- Mode – IPSec can be used in either "transport" or "tunnel" mode. This field designates the mode used.
- Tunnel destination – When using tunnel mode, this indicates the destination address of the outer header.
- Path MTU parameters -- When using the tunnel mode, this field maintains the fragmentation and hop count information. [Ref. 22]

Figure 3.4 depicts the logical policy entities that work together to evaluate every inbound and outbound IP packet to ensure the proper IPSec is applied. As an inbound IP datagram is received, its headers are evaluated against the selectors located in the SPD. If a "selector" designates that this packet must have IPSec applied, the SPD will query the SADB for the corresponding SA (or multiple SAs known as an SA bundle) described by the packet. If no SA is found the entity may dynamically create an SA based on IKE or the packet may be discarded. The SADB and SPI identify the unique security services that are then applied to the packet.

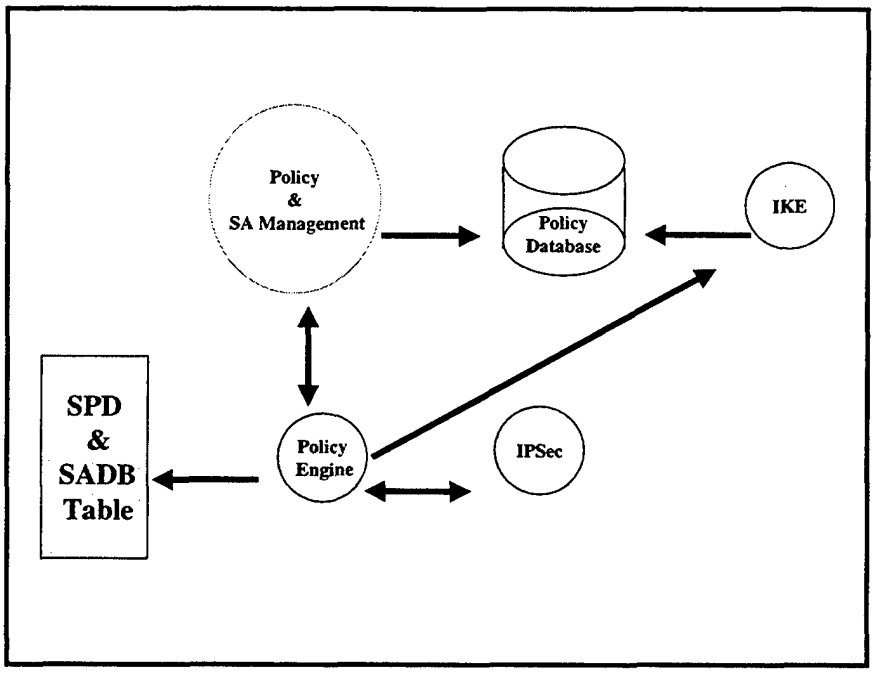

Figure 3.4 IPSec Implementation Architecture, from [Ref. 20]

2. Security Protocols

IPSec defines two specific protocols that provide security services. The first, Authentication Header (AH), provides data integrity, authentication and optional protection against replay attacks. The second, Encapsulating Security Payload (ESP) provides all of these services, but in addition, provides data confidentiality. Each of the protocols can be used in either the "tunnel" or "transport" mode providing multiple combinations of modes and protocols:

- AH in transport mode
- AH in tunnel mode
- ESP in transport mode
- ESP in tunnel mode

The AH protocol is a very simple and sophisticated method to provide data integrity, source authentication and replay attack protection. Due to its simple design, AH requires only an AH header (there is no trailer data) to identify the specific SA to which it applies and the transform it uses. The AH header is inserted into the datagram following the original IP header and before the data payload (Figure 3.5). The original IP header's IP

protocol field is changed to 51 to signify that an AH header follows. The current specifications for the protocol are defined in [Ref. 25].

The security provided with AH comes from its ability to use an authenticator to protect either the upper layer protocol in the transport mode or an entire packet in the tunnel mode. The authentication field holds the result of the integrity checking function. This field is set to zero prior to the integrity computation and then the result is added prior to transmission. The authentication algorithm or hash function (such as Hash Message Authentication Code – Secure Hash Algorithm – 96 bit, HMAC-SHA-96, or Message Digest 5 – 96 bit, HMAC MD5-96), is negotiated as part of the unique SA. IPSec allows for the incorporation of additional algorithm transforms to be defined.

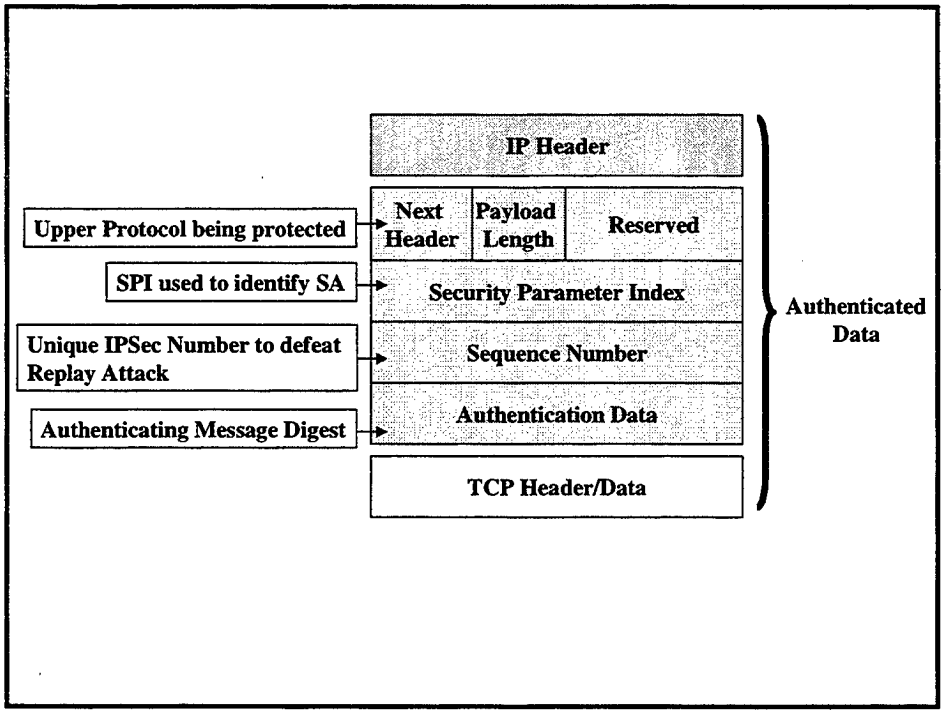

Figure 3.5 Authentication Header Datagram from [Ref. 20]

The Encapsulating Security Payload (ESP) protocol provides data confidentiality and authentication to the IP packets through the use of two encryption algorithms. One, the *encryptor,* is used to protect the data payload and the other, the *authenticator,* verifies

the integrity of the packet. ESP can be used in two manners. First, it can encapsulate only the upper-layer protocol data to provide an encrypted segment of the original IP packet's payload. This allows the original IP header information (and new ESP header/trailer) to be seen. Alternatively, ESP can wrap the entire original IP datagram within an ESP shell. This second option places a new IP header in front of the ESP packet and allows opaque transmission of data to tunnel through the Internet.

All encryption algorithms in ESP use a multiple of the block size of the cipher - or cipher block chaining (CBC) - to encrypt the data. Currently only the DES-CBC transform specification is required for all ESP implementations, but other transforms such as Blowfish-CBC, CAST-CBC or 3DES-CBC can be implemented as options [Ref. 20]. This method of encryption requires an initialization vector (IV) to "jump-start" the encryption process. The IV information is passed to the receiver in the ESP header following the SPI and sequence number similar to that of the AH (Figure 3.6). Additionally, padding may be required if the size of data being encrypted is not a multiple of the CBC block. The trailer contains the authentication digest to verify the integrity of the packet. This hash provides the necessary verification of the SPI, sequence number and IV, which need to be transmitted in plaintext to establish the SA [Ref. 20]. In the transport mode, the original header's IP protocol field is changed to 50 to signify that an ESP header follows and the "ESP header is inserted between the IP header and the upper-layer protocol header. In the tunnel mode, the entire IP packet is encapsulated in the ESP header and a new IP header is added to that." [Ref. 20] The current specifications for the ESP protocol are defined in [Ref. 26].

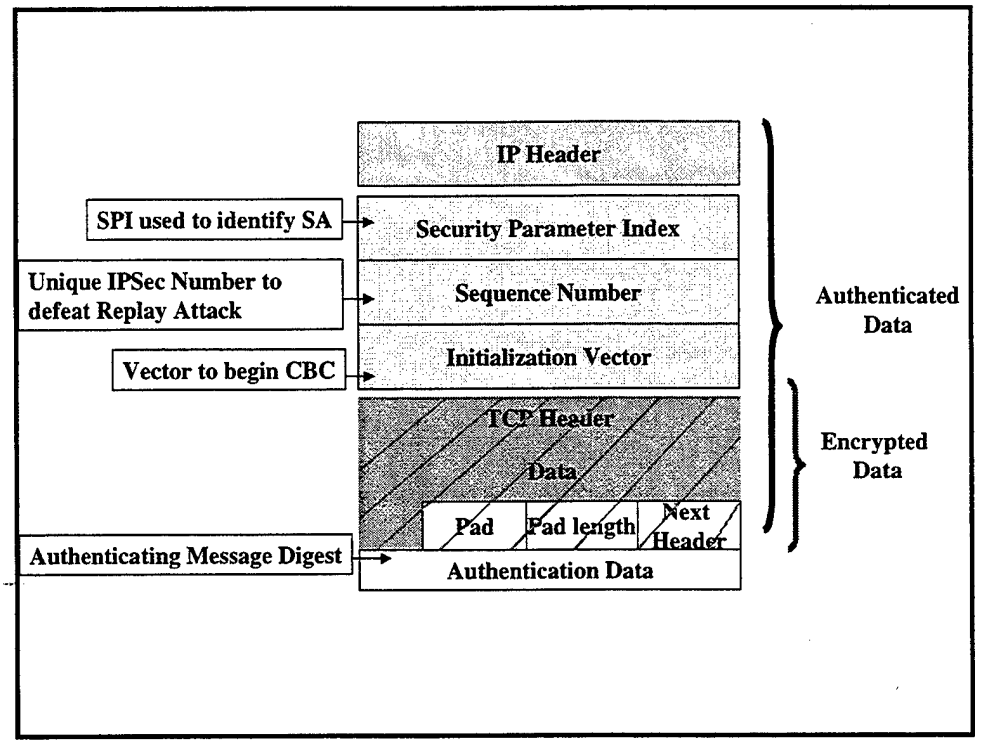

Figure 3.6 ESP Packet in Transport Mode [Ref. 19]

3. Key Negotiation and Management Structure

Before an IP packet can be secured with IPSec, a security association must be established between the entities with which the transmission is to take place. As mentioned previously, this SA establishment can be created either through manual negotiation (offline) or dynamically through online negotiation. The Internet Key Exchange is a hybrid protocol based on a framework defined by the Internet Security Association and Key Management Protocol (ISAKMP) to provide the dynamic negotiation of SAs. IKE is described in [Ref. 22] and incorporates parts of two separate key management protocols –Oakley and SKEME to provide for secure authenticated key exchange. As a hybrid protocol, "IKE uses the foundation of ISAKMP, the modes of Oakley, and the share and re-keying techniques of SKEME to define its own unique way of deriving authenticated keying material and negotiating shared policy" [Ref. 20].

The ISAKMP protocol [Ref. 21] is the basis for the IKE negotiation of key exchange. It uses two separate phases of negotiation to establish the SA. "Phase One" verifies the identity of the two entities and sets up an ISAKMP security association between them. This is necessary to set up an authenticated and secure channel that subsequently can be used to negotiate the specific security services desired in the link. "Phase Two" is the actual negotiation of these services – such as IPSec. Once a Phase One SA has been established between two entities multiple Phase Two negotiations can be conducted. ISAKMP does not define the method used to negotiate the SA policies; this is left to other key exchange documents such as IKE [Ref 20].

"IKE uses the language of ISAKMP to define a key exchange and a way to negotiate security services." [Ref. 20] IKE uses a predefined domain of interpretation (DOI) to outline the required and optional attributes that are negotiated during the Phase Two exchanges. During the IKE Phase One exchanges, the peers must agree on the "protection suite" to be used to encrypt and authenticate their messages. This suite defines the encryption algorithm, hash algorithm, authentication method, and public key exchange to be used.

Once the IKE SA has been established, IKE uses the ISAKMP Phase Two exchanges to generate IPSec SAs. These exchanges effectively concatenate multiple protection suite proposals into the ISAKMP payload to negotiate the specific AH and ESP selectors required for the SA. During these exchanges, the selectors are outlined for the unique SA and each entity records the SA information into their SADB under a unique SPI.

D. SECURITY OPTIONS APPLICABILITY

With proper implementation, both the Transport layer and the Network layer provide adequate end-to-end communications security for a specific connections. However, when the two are evaluated as to their specific applicability to the MLS LAN project, a number of characteristic differences are noted.

1. Application Client/Server Modification

One of the basic goals of the MLS LAN project is to provide a high assurance network that can offer interoperability with commercially procured popular office automation or application software. TLS requires each of the specific higher layer protocol (HTTP, FTP, IMAP, etc.) clients and servers to be modified for "TLS awareness". IPSec, on the other hand has no such requirement. Each IP packet, regardless of application or transport layer protocol will be secured in accordance with the policy defined in the specific negotiated security association. In the MLS LAN, session level information provided to a higher layer application protocol is advisory in nature. Application protocols are not allowed to enforce security policy.

2. Security Policy

The MLS LAN project requires that each of the connections to the TCB have encryption protection that supports sensitivity levels equivalent to or higher than that of the session sensitivity level at which the user is operating. These connections may, in fact, use different encryption transforms depending on the purpose of the connection. For example, a connection to the IMAP server will require Protected Communications Channel with encryption security equal to the user's session level, however, the same user's connection to the TCB Extension Server for session establishment or renegotiation must secured sufficiently to support the system high. If the MLS LAN were to

implement TLS to ensure the appropriate level of protection is provided, each application client and server would require knowledge of both the cryptographic session requirements to be used and the context of the communications between the client and server. In multilevel systems applications are insufficient to enforce security policy.

IPSec provides a mechanism through the Security Policy Database and Security Association Database to segregate the application of protection based upon a set of given attributes. This flexibility lends itself well to defining unique security tunnels to specific source hosts throughout the MLS LAN. The initial SPD of the TCBE can be placed in non-volatile memory, established by the Security Manager with a single entry: to apply security to connect to the TCB Extension Server and disallow all other connections. Once a session has been established, the TCB Extension Server can update the TCBE SPD with the security connection information commensurate with the sensitivity level negotiated on the MLS LAN. From this SPD, the TCBE will correctly negotiate all other connections to MLS LAN hosts utilizing the standard Security Association setup of ISAKMP. This remote management of the security policy of IPSec is not covered in the [Ref. 21], however, a trusted agent developed in the TCB could easily create and pass this information through the TCB-TCBE Protected Communications Channel used to negotiate the session.

3. Domain of Interpretation

Another benefit of IPSec is the use of a predefined domain of interpretation (DOI). Currently the ISAKMP DOI, [Ref. 26] allows the definition a specific "situation" that uses semantics such as "situational identity", "situational secrecy" and "situational integrity" to assemble the parameters for a given Security Association (SA). The DOI

then defines specific "Protocol Identifiers", "Transform or encryption algorithm Identifiers" and attributes such as SA life duration and encapsulation modes to create an SA. The ISAKMP DOI does not specifically address multilevel security, however, a future project could be the development of a MLS DOI for this purpose. The MLS DOI could easily incorporate MLS LAN specific characterization such as limiting the SA life duration default from eight hours to four effectively preventing a workstation from remaining in a session too long without a SAK being physically activated.

The applicability of Network layer security through the use of IPSec complements the goals of the MLS LAN project. Commercial applications and higher layer protocols can be incorporated without code modification. Individual connectivity between end systems can use a single Protected Communications Channel to secure a number of separate protocol services. And most importantly, the Trusted Path can be verifiably secured between the TCB and a TCBE.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. OVERVIEW OF THE MLS LAN CONNECTION FRAMEWORK

This chapter provides an overview and synopsis of the MLS LAN connection protocols presented in the framework. A detailed description of these protocols, can be found in Appendix C.

A. THE MLS LAN PROTECTED COMMUNICATIONS CHANNEL PROTOCOL

1. Overview

The Protected Communications Channel (PCC) protocol is used to create a security conduit between two MLS LAN TCB entities. All other MLS LAN protocols are then required to secure their traffic by using this conduit. The PCC is created through the use of IP layer security as defined in the IP Security Standard for the Internet [Ref. 22]. It provides the MLS LAN with a trusted channel that enforces a "two-way" mutual hardware authentication between the two connecting entities and provides security and integrity protection on all transmitted data. The use of this channel also provides some fault tolerance protection in the event of component loss. This ensures that if the communications between the two Protected Communications Channel connected entities ceases, the overall network will not be affected.

Since the MLS LAN utilizes the IP Security Standard (IPSec) to provide the framework for this channel, this document does not attempt to describe its architecture or mechanisms. Information of these topics can be found in the many RFCs that describe IPSec. Additionally, the specific design of the Protected Channel Initiator (PCI) and data structures necessary for IPSec implementation in the MLS LAN have yet to be finalized.

For this reason, the subsequent sections will provide an approach to be taken in the application of IPSec in the MLS LAN to create a PCC.

2. Logical Placement of MLS LAN IPSec

[Ref. 22] describes three common ways in which IPSec can be implemented in hosts, routers and security gateways.

- **Integration into the native IP layer implementation of the host.** This requires access to the IP source code for the entity that is to use IPSec.
- **"Bump-in-the-Stack" (BITS)** This implementation places the IPSec underneath an existing implementation of the IP protocol stack between the native IP and the local network drivers. This implementation does not require access to the IP source code utilized in the host.
- **"Bump-in-the-Wire" (BITW)** This implementation places an outboard crypto processor that provides the IPSec security services.

As described in Chapter II, the MLS LAN uses the Wang Government Services, Inc. XTS-300™ high assurance server as its source host and includes a prototype TCBE utilizing the Intel™ i960 processor. To maintain simplicity of the XTS-300 security kernel, it is recommended that the MLS LAN implement IPSec in a BITS configuration and create the Protected Communications Initiator as user defined trusted code to be controlled by the security kernel.

3. IPSec Security Policy for the MLS LAN

Each connection to the MLS LAN TCB must be protected in a manner commensurate with the sensitivity of the information transmitted. The Security Manager, e.g., the person responsible information assurance at a given site installation of a MLS

LAN, must ensure that the strength of the assigned encryption mechanisms are sufficient to protect the given sensitivity level. Once assigned, the TCB will maintain a virtual table, which maps the strength of mechanism for a given encryption transform with the sensitivity levels it can support. When encrypted, the information is considered to be safe for transmission across any medium until it reaches its intended recipient. The recipient's act of decryption once again transforms the information into a sensitive form. IPSec provides a mechanism through the Security Policy Database and Security Association Database to segregate the application of protection based upon a set of given attributes [Ref. 22]. The MLS LAN Security Manager will create a listing of the specific security parameters that a Protected Communications Channel must enforce for connection to each of the MLS LAN entities. This information will be maintained by the TCB and mapped to potential client session levels. This enables the TCB Extension Server to know the Security Policy Database (SPD) assignments for each session level.

The initial Security Policy Database of the TCBE will be placed in non-volatile memory, established by the Security Manager with a single entry: to apply security to connect to the TCB Extension Server and disallow all other connections. Once a session has been established, the TCB Extension Server will update the TCBE SPD with the security connection information commensurate with the sensitivity level negotiated for the session. From this Security Policy Database, the TCBE will correctly negotiate all other connections to MLS LAN hosts utilizing the standard Security Association setup of ISAKMP [Ref. 21]. Additional encryption algorithms or transforms can be developed to provide higher levels of encryption, e.g., NSA approved Type I encryption, for use on the MLS LAN. This remote management of the security policy of IPSec is available only

because the MLS LAN TCBE can create the initial Protected Communications Channel at system high through the non-volatile Security Policy Database placed on the TCBE.

A future requirement for the MLS LAN allows a TCBE-equipped workstation to operate as a Non-MLS LAN workstation, e.g., connect to untrusted protocol servers without first connecting to the MLS LAN TCB. In this situation, an additional Security Policy Database and Security Association Database may be required to establish "untrusted" (normal) IPSec security associations to commercial sites. The design and implementation of these mechanisms is left to future work.

4. IPSec Key Management for the MLS LAN

The MLS LAN will use the standard Internet Key Exchange (IKE) [Ref. 24] to define a key exchange and to negotiate security services to be provided for each PCC. IKE uses a predefined domain of interpretation (DOI) to outline the required and optional attributes that are negotiated during the phase two exchanges. Currently the DOI is written specifically for use with the ISAKMP [Ref. 27]. This DOI may be sufficient to provide the security attributes necessary for use in an MLS environment, however, future research may discover that a specific DOI is needed for the MLS LAN Project.

5. MLS LAN PCC Processing

The first Protected Communications Channel established must be a connection between the TCBE-equipped workstation and the source host running the TCB Extension Server process. This is initiated by the TCBE once the user requests attention from the TCB by activating a SAK. The Protected Communications Initiator process on the TCBE will use the initial Security Policy Database setting to establish the IKE Phase One exchanges and establish a secure and authenticated communications channel between the

TCBE and the TCB Extension Server host. Once the IKE security association (SA) has been established, the Phase Two negotiations can then be sent to generate the appropriate incoming and outgoing IPSec SAs. This exchange negotiates the specific AH and ESP selectors required for each SA. During these exchanges, the selectors are outlined for the unique SA and each entity records the SA information into its Security Association Database under a unique Security Parameter Index.

Once the Protected Communications Channel is established between the TCBE and the TCB Extension Server, the user will be allowed to login to the MLS LAN and negotiate a session. If the session establishment is successful, the TCB Extension Server will issue a "PCC Update" command and transfer the appropriate session level Security Policy data to the TCBE for inclusion in its Security Policy Database, as well as make available in the SPD the entries for communicating with other MLS LAN Components, e.g., Application Protocol Servers.

From this point, the user is logged in and operating on the MLS LAN at the negotiated session level. As application protocol services are requested, the TCBE Protected Communications Initiator will use the same method as above to create a separate Protected Communications Channel to the host that supports the requested application protocol server.

B. TCB-TCBE CONNECTION PROTOCOL

1. Overview

The TCB-TCBE Connection protocol is used to provide the Trusted Computing Base (TCB) with a method to conduct security related operations along a trusted path. This protocol is used by the TCBE as a method to gain secure attention from and to

respond to the commands of the TCB. The protocol also provides the TCB Extension Server with a method to control the actions of the TCBE through the use of specific TCBE state commands. The TCB-TCBE Connection protocol will only be initiated through a request for "secure attention" from the user. Protection against replay and spoofing is provided by the underlying Protected Communications Channel.

2. TCBE and TCB Extension Server States

a. TCBE States

The TCBE will use input such as the user activation of the Secure Attention Key or commands received from the TCB Extension Server to change its configuration. This configuration is commonly referred to as the current state of the TCBE. This section will describe the TCBE allowable states, however, the derivation of these states is contained in Appendix C.

There are a total of five allowable states for the TCBE.

- **State 1: Power Off** – The TCBE is not powered or active.
- **State 2: Idle** – The TCBE has been powered, and is prepared for user operations.
- **State 3: Unprotected Operations** – The TCBE has allowed the client workstation to load the operating system, however, it is not connected to the MLS LAN TCB. The design for this state is left for future work.

  Future work should also include a method of login at "system low" that allows the TCB Extension Server knowledge of the user login but not force a purge of the Operating System. For example, this would allow a user who is operating in the Unprotected Operations State, to access the MLS LAN at the lowest possible sensitivity level and utilize print services without a system purge at login.

- **State 4: Trusted Processing** – The TCBE is connected to the TCB to conduct "trusted path operations" such as User Identification and Authentication (I&A) and session negotiation.
- **State 5: Trusted Session** – The TCBE is connected to the TCB in association with a specific negotiated user session level. All previous memory has been purged and a new operating system has been loaded. In this state, the TCBE allows MLS LAN session operations at the negotiated sensitivity level.

b. TCB Extension Server States

The TCB Extension Server will use input such as the receipt of a Secure Attention Request, or TCB-TCBE Connection Protocol "response" payload type received from the TCBE to change its configuration. This configuration is commonly referred to as the current state of the TCB Extension Server. This section will describe the TCB Extension Server allowable states, however, the derivation of these states is contained in Appendix C, Section 3.3.

There are a total of six allowable states for the TCB Extension Server.

- **State 1: Power Off** – In this state the TCB Extension Server is not powered or active.
- **State 2: Idle** – The TCB Extension Server has been powered, and is listening for a Secure Attention Request (SAR) from TCBE to establish a connection. In this state the TCB Extension Server is not connected to the TCBE and the users is not logged in.
- **State 3: Connected** – The TCB Extension Server has made a connection with the TCBE. The TCB has been extended to the TCBE-equipped workstation and using the TCB-TCBE Connection Protocol, User I&A can be conducted.
- **State 4: Logged In** – The TCB Extension Server has validated the User I&A. The TCB Extension Server uses this state to conduct session negotiations through the TCBE to the user to establish a MLS LAN session.
- **State 5: Running** – The TCB Extension Server is connected to the TCBE, and has a user running trusted session operations in the MLS LAN.
- **State 6: Trusted Session Processing** – The TCB Extension Server is still connected to the TCBE and has a valid MLS LAN User logged in, however, a Secure Attention Request has been received. The TCB Extension Server uses this state to interact with the user through the TCBE to change the status of his session.

3. TCB-TCBE Connection Protocol Datagrams

The TCB-TCBE Connection Protocol has fixed Header formats followed by a payload field. There are two defined Header formats for the protocol. The first, the "Payload Datagram" is used to convey information and requests from the TCBE to the

TCB Extension Server. The second, the "Command Datagram" , is provided to enable the TCB Extension Server to control the TCBE State actions and convey information to the TCBE. The composition of these datagrams is provided in Appendix C.

a. Payload Datagram

The TCBE uses the Payload datagram to make requests of the TCB Extension Server and to pass information that the user has entered, such as "Username" or "Password" to the TCB. In Version One of the protocol, there are three Payload packets defined for use by the TCBE. They are as follows:

- **Secure Attention Request**. The TCBE will generate and transmit a Secure Attention Request packet (as described in Section 3.4.1) for each use of the Secure Attention Key by the user, regardless of its current state. This action will transition the TCBE into State [3] (TP Processing) and initialize a Protected Communications Channel or "Trusted Path" to the TCB if one does not already exist.
- **Response**. The TCBE will generate and transmit a Response Packet upon receipt of a Command Datagram packet from the TCB Extension Server that requires a response. The TCBE will remain in State [3] (TP Processing) and wait for input from the user. It will then generate and transmit a Response Datagram packet (as described in Appendix C, Section 3.4.1).
- **PCC Updated**. The TCBE will generate and transmit a PCC Updated packet (as described in Appendix C, Section 3.4.1) following the successful creation of the Protected Communications Channel Security Policy Database from the information provided by the TCB Extension Server.

b. Command Datagram

The TCB Extension Server uses the Command datagram to control the actions of the TCBE and to pass information to the user through the TCBE. In Version One of the protocol, the TCB Extension Server uses one of three Response types to pass the commands.

(1) Response Types

- **No Response**. The TCB Extension Server will generate and transmit a No Response packet (as described in Section 3.4.2) for datagrams when the TCB Extension Server does not require a response. The TCB Extension Server will use this response type for commands that are directive in nature, such as "RUN" or "LOGOUT" or informational in nature, such as "NOOP (No Operation Expected)".
- **Response with Echo**. The TCB Extension Server will generate and transmit a Response with Echo packet (as described in Section 3.4.2) for datagrams when the TCB Extension Server requires a response and there is no protection compromise if the user's response is echoed to the screen. The TCB Extension Server will use this response type for commands that require user input that is not of a private nature, such as "USERNAME" or "SESSION LEVEL CHANGE".
- **Response without Echo**. The TCB Extension Server will generate and transmit a Response without Echo packet (as described in Section 3.4.2) for datagrams when the TCB Extension Server requires a response and there is a possible protection compromise if the user's response is echoed to the screen. This response type will be entered when a response is expected from the TCBE and the TCB Extension Server does NOT allow the TCBE to display the user's response on the screen.

(2) Command Selections. Upon selecting the type of response required from the Command Datagram, the TCB Extension Server uses the Command field to control the actions of the TCBE and to pass information to the user through the TCBE. In Version One of the protocol, the TCB Extension Server may use one of seven command types.

- **No Operation (NOOP).** The NOOP command will cause the TCBE to display the received payload to the user. The nature of the payload is used to provide the user with an interactive login and session negotiation with the TCB. The TCB Extension Server will use this command field value to pass information directly to the user without TCBE intervention, or interpretation.
- **Logout.** The LOGOUT command directs the TCBE to purge the existing Operating System and files from the workstation's memory and return to an "Idle" state.
- **Run.** The RUN command directs the TCBE to transition into State [4] (Trusted Session) with a sanitized version of the Operating System. Any received payload will be displayed to the user. The TCB

Extension Server will use this command field value to activate a session with the TCBE equipped client workstation.

- **Resume.** The TCB Extension Server will use the RESUME command to re-activate a session with the TCBE-equipped client workstation. The RESUME command directs the TCBE to transition back into State [4] (Trusted Session) at the current session level. Any received payload will be displayed to the user. This command directs the TCBE to maintain the original version of the Operating System and return to the user's previous session configuration.

- **New.** This command provides for a future capability in the MLS LAN. The "NEW" command is intended to allow the incorporation of an algorithm, which will determine if the client workstation's Operating System and memory need be purged. The algorithm will perform an evaluation of the user's current sensitivity level and the requested new sensitivity level. If the change in session level will cause a violation of the security policy through the use of the currently running operating system, the system will be purged through a RUN command. If the new session level does not violate the security policy, a NEW command could be used to change the session, but maintain the current operating system. This algorithm is left for future work.

- **Disconnect.** The receipt of a DISCONNECT command terminates the connection to the TCB Extension Server and returns the control of the client workstation to State [1] (Idle). Any received payload will be displayed to the user. This command directs the TCBE to terminate the client workstation's connection to the TCB.

- **Update PCC.** The UPDATE PCC command will direct the TCBE to modify the TCBE Security Policy Database with the data contained in the payload. Once completed, the TCB Extension Server will expect a "PCC Updated" Response packet.

4. TCB-TCBE Connection Protocol Processing

The user initiates the TCB-TCBE Connection protocol only through the activation of the Secure Attention Key. This action directs the TCBE to establish a Protected Communications Channel to the source host running the TCB Extension Server and transmit a Secure Attention Request (SAR) Packet. The receipt of a SAR packet, causes the TCB Extension Server to transmit a series of NOOP commands to request that the user provide a username and password for login. A username prompt will be delivered

using a Response with Echo packet, while a password prompt will be delivered using a Response without Echo packet.

If User I&A are successful, the TCB Extension Server will send to the TCBE a User Interface Menu as a payload in a Response with Echo packet using the NOOP command. The TCBE will display the packet payload to the user. This menu that is displayed provides a listing of selections, which can be used to perform "trusted processing" operations. The listing includes:

- **Session** – This selection provides the user with his current session information.
- **Change Session Level** – This selection provides the user with an interactive exchange with the TCB to negotiate a new Session Level.
- **Change Group** – This selection provides the user with an interactive exchange with the TCB to negotiate a new Group Setting.
- **Logout** – This selection expresses a desire for the User to end his session with the MLS LAN.
- **Run** – This selection tells the TCB that the User is satisfied with his negotiated session and would like to enter Trusted Session Operations.

In response to the "Session" selection, the TCB Extension server will relay the prompts to the user through the TCBE via Response with Echo packets using the NOOP command. The TCBE will simply display the information contained in the Command datagram payload to the user and wait for the user's response. In response to the "Change Session Level" and "Change Group" selections, the TCB Extension Server will enter an interactive exchange to determine the session level the user would like to use. The TCBE and TCB Extension Server will remain in their current State. This information will be presented to the user via Response with Echo packets using the NOOP command. During these exchanges, the TCBE will generate a Response packet with the user's selection or input in the payload and transmit it to the TCB Extension Server. In response to the

"Logout" selection, the TCB Extension Server will issue a Logout command to the TCBE via a No Response packet.

The receipt of a "Run" selection by the TCB Extension Server initiates a process to establish a session on the MLS LAN. The TCB Extension Server must first update the TCBE's Security Policy Database (SPD). This is conducted via a Response without Echo packet using the command "Update PCC". The payload of this datagram will be the database information necessary for the TCBE to negotiate future Protected Communications Channels at the currently negotiated session level. Once the TCBE has completed the update, it will generate and transmit a "PCC Updated" Response packet to the TCB Extension Server. Only following the receipt of this datagram will the TCB Extension Server issue a "Run" command to the TCBE. The receipt of a "Run" command by the TCBE, directs it to purge the client workstation's memory, load a fresh version of the Operating System and enter Trusted Operations. At any time during the process previously described the user activates the Secure Attention Key, the TCBE will suspend the current operation and generate and transmit a Secure Attention Request packet to the TCB Extension Server. The TCB Extension Server will in turn, stop its current process and return to the User I&A portion of the MLS LAN login.

If the User I&A is unsuccessful, the TCB Extension Server will send to the TCBE a No Response packet containing the Command DISCONNECT. This command will direct the TCBE to terminate the connection with the TCB.

Once the User is conducting Trusted Operations at his negotiated session level, no change can be made to this TCB configuration without the activation of a Secure Attention Key. The activation of the SAK, will cause the TCBE to transmit a SAR

packet, however, since the TCB Extension Server knows that the user is currently logged and running valid session, the User Interface Menu, that is sent in response includes an additional selection. The Trusted Session Processing menu includes a selection for "Resume", which allows the user to return to his previously negotiated session without change.

C. SESSION STATUS PROTOCOL

1. Overview

Following the successful session negotiation by a user into the MLS LAN, the TCB Extension Server must create a session database entry through the Session Database Server (SDS) that uniquely defines information such as who the user is, from which TCBE-equipped workstation the user logged in, and the sensitivity and integrity levels assigned to the current session. The integrity of the Session Status Database (SSD) is critical to the assurance of the overall LAN and therefore the ability to manipulate (read/write) its data must be constrained. The Session Status Protocol is provided as a method for the TCB Extension Server, acting as the only TCB entity with both read and write access to the SDS, to modify the contents of the SSD. This protocol is also used by other TCB entities to verify the session status of MLS LAN users. TCB entities, other than the TCB Extension Server are limited to "read only" access. Protection against replay and spoofing is provided by the underlying Protected Communications Channel.

2. TCB Extension Server and Session Database Server States

a. TCB Extension Server States

The TCB Extension Server is the only MLS LAN Entity with the capability to modify the contents of the database managed by the Session Database

Server (SDS). This action can only be taken from three states: State [2] (Connected), State [3] (Logged In), and State [5] (Trusted Session Processing) as described in Appendix C, Section 3.3. While other TCB entities can use this protocol to query the SDS, the state from which their request is issued is not germane to the protocol. The transmission of a Session Status Protocol datagram does not constitute a state transition for any TCB Entity.

b. Session Database Server States

The Session Database Server uses input commands received from the TCB Extension Server to modify the status of the session database, however, the configuration or States of the Session Database Server are not relevant to this protocol.

3. Session Status Protocol Datagrams

The Session Status Protocol has fixed Header formats followed by a payload field. There are two defined Header formats for the protocol. The first, the "Request Datagram" is used to convey information and requests from a TCB Entity to the Session Database Server. The second, the "Reply Datagram", is provided to enable the Session Database Server to respond to the TCB Entity's request. The composition of these datagrams is provided in Appendix C.

a. Request Datagram

All TCB Entities may use the Request datagram to make query (List) requests of the Session Database Server. The TCB Extension Server, however may additionally use the Request datagram to create, delete, or modify records in the Session Status Database. In Version One of the protocol, there are four commands defined for use. They are as follows:

- **List.** This command directs the Session Database Server to locate and return the attribute values contained in the entry found under the listing of the User Session Identification number. The response will determine whether the user is currently logged in.

- **Create.** This command directs the Session Database Server to create a new entry in the database. The TCB Extension Server will use the payload field value to pass the user and session information to the Session Database Server.

- **Modify.** This command directs the Session Database Server to modify a current record in the database. The TCB Extension Server will use the payload field value to pass the user and session information to the Session Database Server.

- **Delete.** This command directs the Session Database Server to delete a current record in the database.

b. Response Datagram

The Session Database Server uses the Response datagram to reply to a TCB Entity's Session Status Protocol Request Datagram. In Version One of the protocol, there are three Response types defined the Session Database Server to use. They are as follows:

- **ACK Response.** The Session Database Server will generate and transmit an ACK Response packet for Request datagrams when the TCB Entity requires only a response determining success. The SDS will use this response type for commands that are directive in nature, such as "CREATE", "MODIFY" and "DELETE". The payload for an ACK RESPONSE packet will contain success verification information for the TCB Extension Server.
- **NAK Response.** The Session Database Server will generate and transmit a NAK Response packet for Request datagrams when the TCB Entity requires determination of failure. The SDS will use this response type for commands such as "CREATE", "LIST", "MODIFY" and "DELETE". The payload for a NAK RESPONSE packet may contain information for the TCB Entity concerning the reason for the failure.
- **Payload Response.** The Session Database Server will generate and transmit a Payload Response packet for Request datagrams when the TCB Entity requires the information contained in the record. This response type will be entered when the SDS has been issued a command that requires the return of information contained in a database entry.

4. Session Status Protocol Processing

a. Use of the Session Status Protocol by TCB Entities other than the TCB Extension Server.

Upon Receipt of a request for Network Application Services, a TCB Entity will generate and transmit a LIST Request packet placing the requestor's TCBE ID in the User Session Identification field. This command directs the Session Database Server to locate and return the attribute values contained in the entry found under the listing of the User Session Identification number. The SDS will transmit this information using a "PAYLOAD" Response datagram. The response will determine whether the user is currently logged in. If the user is logged in, the TCB entity will continue with the connection process as described in Appendix C, Section 5.3. If, however, a NAK Response packet is received from the Session Database Server, the TCB entity will terminate the Application Protocol connection to the requesting TCBE-equipped workstation. No other Request datagram command selections are available for these TCB entities.

b. Use of the Session Status Protocol the TCB Extension Server.

The TCB Extension Server will generate and transmit a Request packet using the "LIST" command each time it receives a SAR packet. This enables the TCB Extension Server to query the Session Database Server to determine if a previous entry has been created for the identified TCBE. The response, as previously described, will determine whether the user is currently logged in. If the user is logged in, the TCB Extension Server will transition to State [3] (Logged in). If, however, a NAK Response

packet is received from the SDS, the TCB Extension Server will continue with the User I&A session as described in Appendix C, Section 3.5.2.c. and remain in its current State.

If the user is not logged in, the TCB Extension Server will complete the User I&A and it issue a Request packet using the "CREATE" command to instantiate a record for the new user. The TCB Extension Server will use this payload field value to pass the user and session information to the SDS. This command must be completed prior to the TCB Extension Server's transition to State [3] (Logged in). The SDS will generate an "ACK" Response packet upon completion. A "NAK" response will cause a retransmission. If a response is not returned, the TCB Extension Server will initialize a command mechanism to prevent all further connections to the MLS LAN or its services until communications to the SDS have been restored. This command mechanism is left to future work.

Once in the "Logged In" State, the TCB Extension Server will allow the user to negotiate a session in the MLS LAN through the TCB-TCBE Connection protocol as described in section B.4 of this Chapter. Upon the receipt of a TCB-TCBE Protocol "Payload" packet containing a "RUN" request, from the TCBE, the TCB Extension Server will issue a Request packet using the "MODIFY" command to request the SDS update the current session information to the values negotiated during the Trusted Path Processing. The SDS will use this command field to change the value of one or more of the attributes of a current database entry. The SDS will generate an "ACK" Response packet upon completion. A "NAK" response will cause a retransmission.

At the completion of the user's session, through either a TCB-TCBE Protocol "Payload" packet containing a "LOGOUT" request or the issuance of a

"DISCONNECT" from the TCB, the TCB Extension Server will issue a Request packet using a "DELETE" command. This command requests the SDS remove the User's current session record. The SDS will generate an "ACK" Response packet upon completion. A "NAK" response will cause a retransmission. The logout of a user does not depend on the success of this action.

D. TCBE-TO-SESSION SERVER CONNECTION PROTOCOL

1. Overview

The MLS LAN is intended to provide access to multiple Application Layer Protocols such as FTP, HTTP, or IMAP. For Version 1, these application services are only accessible to users who have successfully logged in to the MLS LAN and established a Session within the TCB. The TCBE-to-Session Server Connection Protocol is provided as a method for the TCBE to pass a unique identifier to the Secure Session Server (SSS) in order for it to check with the Session Database Server (SDS) for the user's session information. The MLS LAN uses the TCBE Identification Number as this identifier. The design of this protocol, however, will allow alternate future data, such as a unique session token, to be inserted adding flexibility to the MLS LAN. Once the user's information is returned from the SDS, the Secure Session Server will establish the proper session level connectivity to the appropriate MLS LAN Application Protocol Server (APS) as described in [Ref. 13]. If, however, the user is not found by the SDS, the connection to the Application Protocol Server will be terminated.

2. TCBE and Secure Session Server States

a. TCBE States

The TCBE uses this protocol only to pass its unique identifier to the Secure Session Server. In the current version of this protocol this action can only be taken from one state: State [4] (Trusted Session), however future versions may allow for this protocol in State [2] (Unprotected Operations) as described in Appendix C, Section 3.2. The use of this protocol does not constitute a state transition for the TCBE.

b. Secure Session Server States

A Secure Session Server is created for each higher layer application protocol supported by the MLS LAN. Its responsibility is to accept and validate requests for access to the particular protocol. The Secure Session Server uses the TCP/IP Application Protocol connection request packet from the TCBE equipped client workstation to change its configuration. The configuration of the Secure Session Server is not relevant to the use of this protocol.

3. TCBE-to-Secure Session Server Connection Protocol Datagrams

The TCBE-to-Secure Session Server Connection Protocol has a single fixed Header format followed by a payload field. The "Identification Datagram", is provided to enable the TCBE to pass its unique TCBE ID to the Secure Session Server. The composition of this datagram is provided in Appendix C.

4. TCBE-to-Secure Session Server Connection Protocol Processing

Upon the receipt of a "Application Protocol Service Connection Request" from a higher layer protocol client residing on the client workstation, the TCBE will

generate and transmit an Identification packet to the Secure Session Server which hosts that protocol.

The Secure Session Server does not respond directly to the TCBE using this protocol. The Secure Session Server uses the information contained in the TCBE-to-Session Server datagram to generate and transmit a Session Status Protocol Request packet using the "LIST" command to the Session Database Server as described in Appendix C, Section 4.5.1. This command will verify the user's current session information. Once this information has been verified, the Secure Session Server will continue with the Application Protocol Server operations as described in Appendix C, Section 4. If the user is not logged in, the Secure Session Server will simply terminate the connection to the requesting application. If the Identification datagram is not received, the "LIST" command cannot be transmitted and the Secure Session Server cannot connect the Application Protocol client request to the Application Protocol Server. This action will, in turn cause a time out in the Application Layer, thus requiring a retry.

V. CONCLUSIONS

A. MLS LAN PROJECT DEVELOPMENT

1. Previous Efforts

The MLS LAN Project is an ongoing effort. Most of these efforts have been documented in the thesis work of Naval Postgraduate School Graduates [Refs. 28, 29, 30]. It was the study of these documents, in addition to the exceptional instruction provided by the NPS CISR staff and study of numerous seminal papers on Computer/Network Security, which provided the requisite foundation to understand both the magnitude of the endeavor and the structure of the MLS LAN. Of particular note were the following documents:

a. NPS Thesis: Secure Local Area Network Services for a High Assurance Multilevel Network, by Susan BryerJoyner and Scott Heller, March 1999 [Ref. 28].

This thesis provided the initial design and proof-of-concept implementation for a secure LAN that supported the extension of the Trusted Computing Base to commercial grade Personal Computers. The culmination of this work furnished the NPS laboratory with an initial demonstration prototype of the basic MLS LAN.

b. NPS Thesis: Design of a High Assurance, Multilevel Secure Mail Server (HAMMS), by James Downey and Dion Robb, September, 1997 [Ref.29].

This thesis provided the requisite design characteristics for a high assurance mail server. While the current Application Protocol Server used in the MLS LAN project has changed, this work gave an overview of the issues involved in

multilevel operations and the incorporation of the Wang Federal XTS-300 high assurance server.

c. NPS Thesis: Analysis for a Trusted Computing Base Extension Prototype Board, by Bora Turan March, 2000 [Ref.30].

One of the fundamental enabling prerequisites for the MLS LAN project is its ability to extend the Trusted Computing Base from the high assurance server to a commercial PC. This thesis describes the hardware and software design for a custom plug-in board that can both successfully complete the trusted path connection and control the client PC. The completion of this work, with its functioning prototype, provided confidence in the premise that MLS LAN client PC's can be connected to the network through non-by-passable, tamper resistant network interface cards.

2. Engineering Team Effort

The system requirements and protocol design, that are part of this thesis were reviewed, discussed, and revised by an engineering team. The composition of the team included senior investigators from the NPS CISR staff, the MLS LAN design engineer, TCBE hardware/software engineers and the author, as the network/protocol engineer. This approach brought to the table decades of focused study in the areas of computer security, software and hardware engineering, and project development.

The engineering team approach was, without a doubt, an important factor in the successful development of the three documents that are the appendices in this thesis. With this, however, some comments and recommendations should be added to enhance future team efforts of this nature.

a. Mission and Format.

The initial meetings of the team were focused on the identification and development of protocol specifications for MLS LAN connectivity. Many hours were spent in the development of protocol proposals that required design decisions on the requirements of the MLS LAN system as a whole. Since there was no previous Systems Requirement Document outlining what the MLS LAN was to provide its users or the overall architecture of its components, the focus of the team's meetings shifted to its development. Concurrently, work continued on a high level analysis of what connection protocols would be required to implement the system. These efforts culminated in the MLS LAN Project Systems Requirement Document and Protocol High Level Analysis Document found in appendices A and B of this thesis.

A recommendation for future engineering team efforts would be to start with the identification of the mission requirements and use these to establish engineering goals.

b. Leadership and Team Composition

Key to the success of any team effort is the guidance provided by the team leader and the ability of the team to cooperatively work toward a collective goal. In this, we were blessed with both a strong leader who allowed free and open discussion, and a superb group of individuals, whose personalities and expertise complimented one another. The ability to professionally discuss, and sometimes argue a point without fear of personal ridicule or damaged feelings, creates a healthy work environment and leads to

success. This atmosphere is a product of the leadership brought to the team and should be emulated in future engineering team meetings.

c. Documentation

The documents produced by the engineering team went through many revisions. The need for copious notes and comments on modifications cannot be over-stated. The first documents provided to the team did not contain functional paragraph formatting or date/time attributions. This mistake was rectified, making the changes easier to track. Additionally, the product developer must provide the team adequate time to study proposals before team meetings. Many times, new proposals or changes were finalized the night before a meeting. This did not offer the team members sufficient time to review the work and slowed the progress of some meetings.

B. FUTURE WORK

1. Limitation of Session Sensitivity Levels.

Each TCBE could be assigned a security rating commensurate with its location or use. A security rating would indicate the highest sensitivity level the specific TCBE would be allowed to support. This would mean TCBE-equipped workstations located in physically secured spaces could be assigned ratings equal to the space, while TCBE-equipped workstations operating in non-secure surroundings could be assigned lower security ratings. The assignment of this security rating would allow the creation of an algorithm to enable the TCB to limit the allowable session sensitivity-level to the greatest lower bound between the user's clearance and the TCBE security rating. Once the

security ratings are assigned, physical control must be provided to TCBEs with high ratings.

2. Acceptance of a Non-TCBE-Equipped Workstation.

Current versions of the MLS LAN connection protocols require the use of a valid TCBE ID to establish a connection to the TCB Extension Server. The intent of the MLS LAN Project, however, is to provide flexibility by allowing connections by both TCBE and non-TCBE equipped workstations. This will require a modification of the existing protocols to use more generic User-Session identification values, such as a token. The value must maintain the protection of the unique identification of the TCBE gained by the use of the current TCBE ID, but must also support the identification of a non-TCBE-equipped workstation.

3. Non-TCBE-Equipped Workstations Access to Application Protocol Servers.

A future goal of the MLS LAN is the ability to support the connection of a workstation that is not using TCBE services to a MLS LAN Application Protocol Server (APS). This would allow normal commercially procured workstations, or TCBE-equipped workstations operating in an "Unprotected Operations" mode, to gain access to MLS LAN services operating as a system defined anonymous user.

Additionally, the future MLS LAN may allow the connection to an untrusted Application Protocol Server, such as Web or print Server, for use by non-TCB authenticated users. The Secure Session Server would require a method to accept Network Application Protocol Services requests from workstation/users that have not established a session and to pass these on to an untrusted APS. The user would need to

be accepted at a system defined low secrecy, low integrity, session sensitivity level. The use of untrusted APS would provide a method of login at "system low" that allows the TCB Extension Server knowledge of the user login, but not force a purge of the Operating System on the client workstation. For example, this would allow a user who is operating in the Unprotected Operations State, to access the MLS LAN at the lowest possible sensitivity level and utilize print services without a system purge at login.

Another possible service to be provided in an "Unprotected Operations" mode is the connection to a Non-MLS LAN Application Protocol Server, such as a commercial HTTP Web host (e.g., Yahoo.com). In this situation, an additional Security Policy Database and Security Association Database may be required to establish "untrusted" (normal) IPSec security associations to commercial sites.

4. Session Domination Algorithm.

A future modification to the TCB-to-TCBE Protocol must incorporate a "session domination algorithm" to determine if the operating state of the workstation requires modification (e.g., if the requested session sensitivity-level dominates the current session, the workstation operating system need not be cleared). This algorithm would be employed when a user requests a change in session level. The algorithm would perform a comparison of the user's current sensitivity level and the requested new sensitivity level. If the change in session level would cause a potential violation of the security policy through the use of the currently running client PC operating system, the client workstation must be purged by using a RUN command. If the new session level does not

violate the security policy, a NEW command could be used to change the session, but maintain the current operating system.

5. Protected Channel Initiator

The Protector Channel Initiator is a user-developed trusted module responsible for the creation of the Protected Communications Channel (PCC) between two MLS LAN components. This process is a software implementation of Network layer IPSec that will be placed into the XTS-300 and the TCBE. The initiator process will enforce a "two-way" mutual hardware authentication between the two connecting entities and provide security and integrity protection on all transmitted data.

6. Distributed Session Database

Currently the Session Database Server located on a single XTS-300 source host maintains the Session Status Database. If the Session Status Database were to be distributed throughout all XTS-300 source hosts on the MLS LAN more efficiency may be gained in the connections between the TCBE and Application Protocol Servers. This approach may also provide support for the use of token-based access. The Session Database Server could easily provide the database synchronization required to incorporate the distributed implementation of the database.

7. Session Time Control Mechanism

One of the protection mechanisms sought for the MLS LAN is the ability of the TCB to maintain control over the user's LAN connection. The intent is to enable the TCB to confirm that the user is still physically there. This may require the development of a mechanism to control the time that a user may remain in a session without the

physical activation of the Secure Attention Key. A control protocol mechanism could be developed for transmission between the TCB Extension Server and other TCBE entities to pass directions, such as a request for a "user initiated SAK". If a Secure Attention Request is not returned, the LAN connection could be terminated. This control mechanism could also be expanded for use with other events germane to the TCB such as:

- Network administrative control.
- Network loss or restoration control.
- User Termination control. (to disconnect some or all MLS LAN Services from user).

8. TCB-TCBE Trusted Path Connectivity

The pros and cons of a persistent trusted path between the TCBE and the TCB Extension Server must be evaluated in depth with respect to the enforcement of the security policy by the TCB.

9. MLS LAN Domain of Interpretation

The ISAKMP Domain of Interpretation (DOI) does not specifically address multilevel security. This DOI may be sufficient to provide the security attributes necessary for use in an MLS environment; however, future research may reveal that a more specific DOI is needed for the MLS LAN Project.

C. CONCLUSIONS

The Multilevel Secure Local Area Network connection framework presented in this thesis is intended to provide protected communications between each of the components of the MLS LAN to ensure single level users can access multilevel data.

Initial user connection to the MLS LAN was described through a Trusted Path or Protected Communications Channel, which utilized the Internet Protocol Security Standard to sufficiently provide security for data transfer throughout the MLS LAN. A specific connection protocol was described to enable the TCB to extend protection and control to the TCBE-equipped workstation and enable the user to negotiate access to the LAN through the actions of the TCBE. A protocol was described that allows positive control of the Session Status Database by the TCB Extension Server, while concurrently enabling other TCB Entities query capability. Finally, a protocol was provided that enables users operating in trusted sessions to access Network Application Protocol Services.

This framework, coupled with the Systems Requirements Document and Protocol High Level Analysis Document included in the appendices, has proven that the MLS LAN initiative to extend the TCB to TCBE-equipped commercially procured personal computers can securely establish multilevel access across a LAN. I am confident that this thesis, in concert with the previous work on MLS hardware and software solutions, and ongoing research by the faculty and students at the Naval Postgraduate School will culminate in a realistic, workable, and cost effective solution to the Multilevel Secure LAN problem.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. MLS LAN SYSTEM REQUIREMENTS DOCUMENT

This requirements document provides extensive information concerning the design requirements for each of the components of the MLS LAN project. It outlines the mandated system goals perceived for successful completion of the project and the development of an operational multilevel secure local area network. It is understood that some of the specified requirements are designated as mandatory to fulfill near-term functionality and are to be addressed in the initial design. Other requirements, where annotated, are considered to be future goals and are recorded to support long-range design specifications. This requirements document should provide sufficient detail and content to assist the design team in specification definition.

Table of Contents

1. Introduction:
1.1. Purpose
1.2. Scope
2. The System OVERVIEW
2.1. MLS LAN System Overview
2.2. MLS LAN User Description
2.3. Component Descriptions
3. System Requirements
3.1. MLS LAN Requirements
3.2. Trusted Computing Base Requirements
3.3. MLS LAN Network Application Protocol Services Requirements
3.4. MLS LAN Workstation Requirements
4. System Restrictions
5. Appendix
    Appendix A – Abbreviations, Acronyms, and Definitions
    Appendix B – References

Naval Postgraduate School Center for INFOSEC Studies and Research

Multilevel Secure Local Area Network Project

SYSTEM REQUIREMENTS DOCUMENT VERSION 1.0

J.D. WILSON

APRIL 20, 2000

TABLE OF CONTENTS

1. INTRODUCTION: 84

1.1. Purpose 84

1.2. Scope 84

2. THE SYSTEM OVERVIEW: 84

2.1. MLS LAN System Overview 84

2.2. MLS LAN User Description 85

2.3. Component Descriptions 85

3. SYSTEM REQUIREMENTS: 87

3.1. MLS LAN Requirements: 87

3.2. Trusted Computing Base Requirements: 88

3.3. MLS LAN Network Application Protocol Services Requirements: 90

3.4. MLS LAN Workstation Requirements: 91

4. SYSTEM RESTRICTIONS: 91

5. APPENDIX: 93

Appendix A – Abbreviations, Acronyms, and Definitions 93

Appendix B – References 95

1. Introduction:

1.1 Purpose.

The purpose of this System Requirements Document is to define the design requirements for the Naval Postgraduate School Center for InfoSec Studies and Research (CISR) Multilevel Secure Local Area Network (MLS LAN) Project. This document is a product of a team effort led by Dr. Cynthia Irvine, Director of NPS CISR. The team members include: Mr. Timothy Levin, Mr. David Shifflett, Ms. Barbara Pereira, LtCol. J.D. Wilson, USMC, and the assistance of Mr. James P. Anderson, of J.P.A. Co.

1.2 Scope.

This requirements document provides extensive information concerning the design requirements for each of the components of the MLS LAN project. It outlines the mandated system goals perceived for successful completion of the project and the development of an operational multilevel secure local area network. It is understood that some of the specified requirements are designated as mandatory to fulfill near-term functionality and are to be addressed in the initial design. Other requirements, where annotated, are considered to be future goals and are recorded to support long-range design specifications. This requirements document should provide sufficient detail and content to assist the design team in specification definition.

2. The System Overview:

2.1. MLS LAN System Overview.

The MLS LAN Project is an effort to provide government and commercial organizations with a cost effective, multilevel, easy-to-use office environment leveraging existing high assurance technology. The goals of the project are to produce a networking environment that provides concurrent high assurance access for network users to data at multiple sensitivity levels through the incorporation of inexpensive commercial personal computers.

The proposed systems architecture for the MLS LAN is based on the use of the Wang Government Services Incorporated XTS-300™ B3 rated server. [Ref. 1] The XTS-300's multilevel features provide both mandatory and discretionary access controls, which "allow separation of users who are at different clearance levels, and prevents a lower level user from reading a higher level user's files or data". [Ref. 2] In accordance with the TCSEC Class B3 rating requirements, the XTS-300 establishes a "Trusted Computing Base" (TCB) that contains all of the Trusted Software Commands, the TCB System Services (TSS), and the Security Kernel. It is the last that implements the TCSEC defined Reference Monitor concept in the XTS-300 [Ref 3]. The MLS LAN incorporates

a **"logically isolated and unmistakably distinguishable"** trusted communications path between the server and its clients through development of a Trusted Computing Base Extension (TCBE). The TCBE will provide a trusted network interface entity for verifiable expansion of the TCB over the communications path to the client workstation. The current hardware solution for the TCBE is to be developed using the Intel I960jx processor. The TCBE will dominate all actions of the untrusted workstation and allow connectivity into the High Assurance LAN only following the establishment of a trusted path.

2.2 MLS LAN User Description.

The MLS LAN user is any operator, regardless of authentication, who accesses MLS LAN resources or network functionality. A TCB Authenticated user is one who has successfully established a TCB-to-User connection and been validated by the TCB for operations within the MLS LAN. A Non-TCB Authenticated User, which is a future requirement, is one who has not been validated by the TCB. Accountability of Non-TCB Authenticated users shall be provided using existing commercial authentication and identification mechanisms.

2.3 Component Descriptions.

The MLS LAN is comprised of three components (Fig 2.1). The principle component is the Trusted Computing Base (TCB), which provides an fixed security perimeter for MLS LAN operations. Network functionality for access to available application software, file transfer, electronic mail, or remote printing is provided by the Network Application Protocol Services. Finally, the MLS LAN requires a workstation that acts an agent for the User to access any required network functionality.

2.3.1. Trusted Computing Base. The Trusted Computing Base is an abstraction for the collection of elements of a computer system that pertain to the security policy. Its aegis encompasses all policy enforcement mechanisms, any auditing (retrieval and analysis), identification and authentication, and the interface for security administration.

2.3.1.1. Trusted Computing Base Services. The services provided by the MLS LAN to establish a Class B3 rated Trusted Computing Base were outlined in section 2.1 "MLS LAN System Overview". To extend this TCB securely to users additional services are required.

2.3.1.1.1. TCBE Extension Server. The use of the XTS-300 High Assurance Server enables the MLS LAN to place a trusted daemon process in the Operating System Services (OSS) Domain that can provide the protection and communications protocols necessary to establish a trusted path between the workstation and MLS LAN. This "Server" process is used to extend the TCB perimeter securely over

the network to the requesting TCBE equipped workstation. This "Server" process will provide the following functionality: user identification and authentication, session negotiation, session activation, and session termination. [Ref 5.]

2.3.1.1.2. Secure Session Server. The Secure Session Server is an additional trusted daemon "Server" process contained in the OSS. This process will only accept incoming Network Application Protocol Service requests from workstations/users that have established a session via the trusted path and the TCB Extension Server. Validated requests will be passed on to untrusted Application Protocol Servers, operating on behalf of the user, at the user's negotiated session sensitivity level [Ref 5.]

(***Future Requirement***) The Secure Session Server will accept Network Application Protocol Services requests from workstation/users that have not established a session. These requests will be passed on to untrusted Application Protocol Servers, operating as a system defined anonymous user, at a system defined low secrecy, low integrity, session sensitivity level.

2.3.1.1.3. MLS LAN Session Database Server. The MLS LAN requires a trusted database to maintain all pertinent information concerning each unique TCB session connection. The Session Database Server must provide protection for trusted "read" functionality from all TCB entities and "write" functionality from the TCB Extension Server.

2.3.1.2. Trusted Computing Base Extension. The Trusted Computing Base Extension (TCBE) is a hardware-based computer subsystem that is embedded into the MLS LAN workstation. The TCBE provides the MLS LAN with a verifiable high assurance entity that can be used to extend the TCB.

2.3.1.3. MLS LAN Connection Protocols. The MLS LAN connection protocols define the parameters for initiation, security and communications establishment between two or more components of the MLS LAN.

2.3.2 Network Application Protocol Services.

The MLS LAN uses the TCP/IP stack to support numerous Application Layer Protocol services such as Hyper Text Transfer Protocol (HTTP), Internet Message Access Protocol (IMAP), and File Transfer Protocol (FTP). These services are provided to the users through Application Protocol Servers (APS). While use of these application services are considered "untrusted" and external to the TCB, their access is controlled strictly through the Secure Session Server allowing access to data of multiple sensitivity levels.

2.3.3 MLS LAN Workstation.

The MLS LAN workstations are the network computers employed by the user to access MLS LAN resources and network functionality.

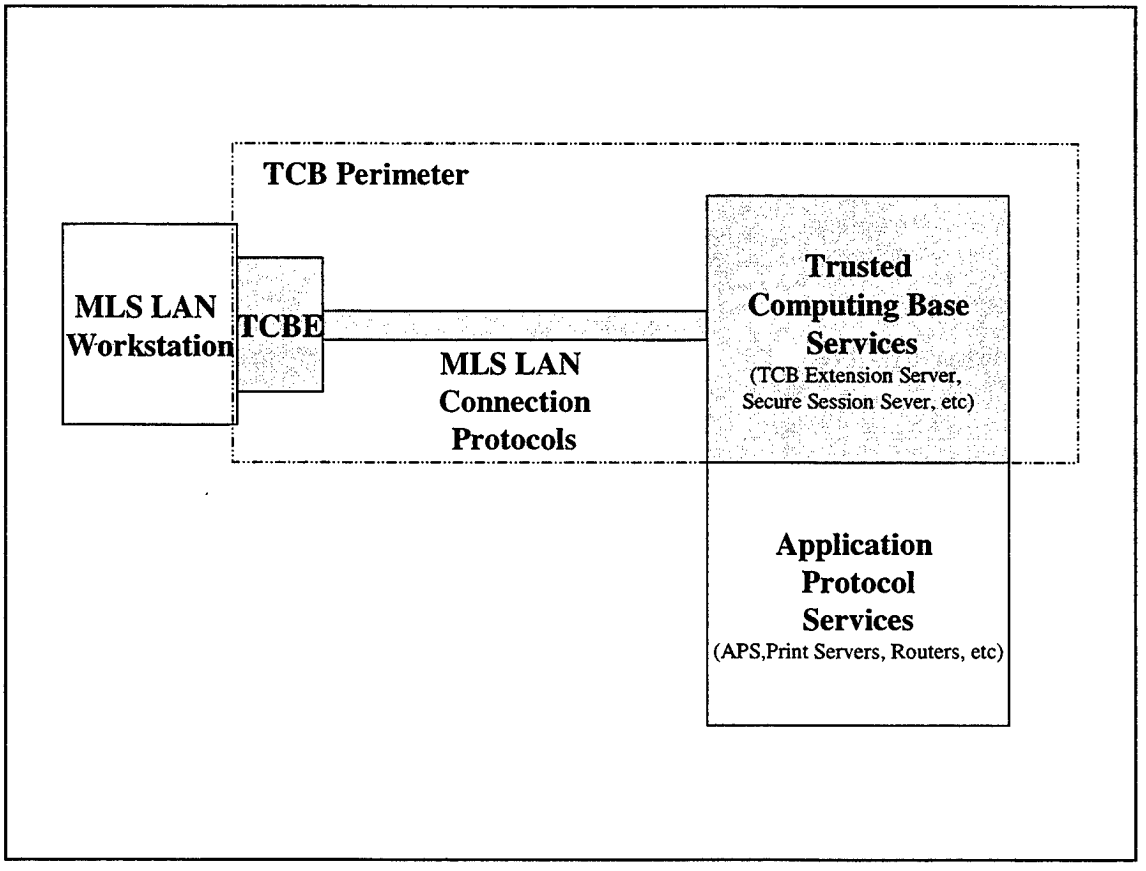

Figure 2.1 MLS LAN Component Overview

3. System Requirements:

3.1. MLS LAN Requirements.

3.1.1. The MLS LAN shall support multiple simultaneous workstation connections.

3.1.2. The MLS LAN shall support simultaneous high assurance access for unique workstations operating at different sensitivity levels.

3.1.3. The MLS LAN shall provide access to shared resources, application protocol services, and popular application products for both TCB Authenticated Users and, in the future, Non-TCB Authenticated Users.

3.1.4. The MLS LAN shall provide high assurance connectivity to application protocols that give access to multiple levels of data in accordance with security policies.

3.2. Trusted Computing Base Requirements.

This section elaborates on the requirements for the TCB in total. The overall requirements are germane to each of the sub-components while their specific requirements are contained in subsequent sections. A abstract depiction of the MLS LAN layering is provided in figure 3.1.

3.2.1. TCB Overall Requirements.

3.2.1.1 The TCB shall provide a Secure Attention Key (SAK) mechanism to invoke a trusted path from workstations to which the TCB has been extended.

3.2.1.2. The TCB shall establish a trusted path communications connection between network users and the Trusted Computing Base. This trusted path shall be established for initial session authentication purposes, such as "login" or for any specified user operations that require a trusted path, such as "logout", "set session level", downgrade, change user password, etc.

3.2.1.3. Once the session has been established, the TCB shall not allow the TCB-to-TCBE Protocol Channel to be broken without loss of network functionality with respect to shared resources, protocol services and applications provided by the MLS LAN.

3.2.1.4. The TCB shall allow the user to change the current session sensitivity-level up to the configured maximum for that user.

3.2.1.5. The TCB shall provide assurance that the security policy will be enforced in the presence of malicious software.

3.2.1.6. The TCB shall provide protection against disclosure and modification of information on all communications channels used by the network.

3.2.1.7. The TCB shall control access all devices and networks external to the MLS LAN.

3.2.1.8. (***Future Requirement***) The TCB shall limit the allowable session sensitivity-level to the greatest lower bound between the user's clearance and the TCBE security rating.

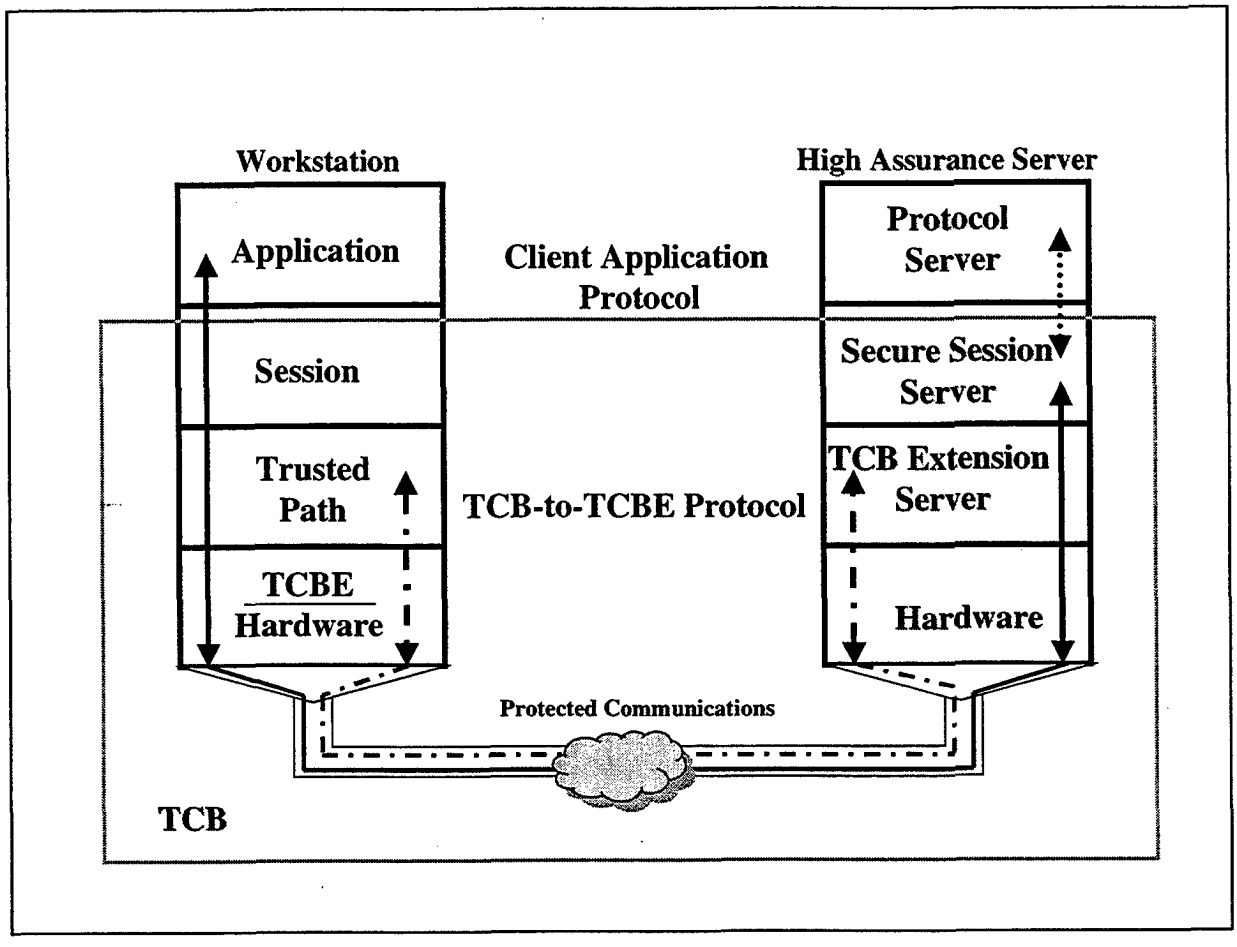

Figure 3.1 TCB Layering Abstractions

3.2.2. Trusted Computing Base Extension Requirements.

3.2.2.1. The TCBE shall support the use of Trusted Path communications with the TCB for security related operations.

3.2.2.2. The TCBE shall prevent data retention between session security levels and support proper object reuse.

3.2.2.3. The TCBE shall support a hardware mechanism that has the ability to purge all memory between session security levels.

3.2.2.4. The TCBE shall maintain the ability to reset the host computer system.

3.2.2.5. The TCBE shall support the use of a secure attention key.

3.2.2.6. The TCBE shall control the information flow into and out of the host computer system.

3.2.3. MLS LAN Connection Protocol Requirements.

3.2.3.1. The MLS LAN shall provide a protocol that supports both the establishment of a secure interaction communications channel and the mutual authentication between two TCB entities. This protocol will be known as the "Protected Communications Channel (PCC) Protocol". This protocol will establish the security conduit through which all other MLS LAN protocols operate.

3.2.3.2. The MLS LAN shall provide a protocol to support communications between a TCBE equipped workstation and the TCB Extension Server. This protocol will be known as the "TCB-to-TCBE Protocol".

3.2.3.3. The MLS LAN shall provide a protocol to support the secure transfer of information from the TCB Extension Server to the Session Database Server to initialize or modify the data maintained on each User Session. This protocol will additionally support the query by a TCB Entity to the Session Database Server for information concerning a User Session. This protocol will be known as the Session Status Protocol.

3.2.3.4. The MLS LAN shall provide a protocol to support a TCBE equipped workstation connection to a MLS LAN Secure Session Server. This protocol is the conduit for application protocols and will be known as the "TCBE-to-Session Server Protocol".

3.2.3.6. (***Future Requirement***) The MLS LAN shall provide a protocol to support the connection of a workstation that is not using TCBE services to an untrusted Application Protocol Server, e.g., INTERNET or WWW.

3.2.3.7. (***Future Requirement***) The MLS LAN shall provide a protocol to support a connection of a workstation that is not using TCBE services to a MLS LAN Application Protocol Server.

3.3. MLS LAN Network Application Protocol Services Requirements.

3.3.1. The MLS LAN shall support multiple simultaneous accesses to higher layer application protocols, e.g., HTTP, IMAP or FTP.

3.3.2. The MLS LAN Application Protocol Servers shall provide access to shared network resources, and popular application products for TCB authenticated users.

3.3.3. Access to data maintained on the MLS LAN Applications Protocol Servers (APS) shall be controlled the through the TCB in accordance with the security policy.

3.3.4. (***Future Requirement***) The MLS LAN Application Protocol Servers shall provide access to shared network resources, and popular application products for Non-TCB authenticated users

3.4. MLS LAN Workstation Requirements.

3.4.1. The MLS LAN shall support the use of two configurations of inexpensive commercial personal computers:

3.4.1.1. Trusted Computing Base Extension (TCBE) equipped.

3.4.1.2. (***Future Requirement***) Non-TCBE equipped.

3.4.2. The MLS LAN Workstations shall support up-to-date commercial operating systems.

3.4.3. The MLS LAN TCBE Equipped Workstation shall be "diskless thin-client" computers operating under the control of the TCBE.4. MLS LAN SYSTEM RESTRICTIONS:

4. MLS LAN System Restrictions.

4.1 MLS LAN Restrictions

4.1.1 The MLS LAN shall support no more than one logged in user per workstation at a time.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX

Appendix A. Abbreviations, Acronyms, and Definitions.

A.1. Abbreviations, Acronyms

APS – Application Protocol Server

CISR – Center for InfoSec Studies and Research

FTP – File Transfer Protocol

HTTP – Hyper Text Transfer Protocol

IMAP – Internet Message Access Protocol

LAN – Local Area Network

MLS – Multilevel Secure

NPS – Naval Postgraduate School

OSS – Operating System Services

SAK – Secure Attention Key

TCB – Trusted Computing Base

TCBE – Trusted Computing Base Extension

TIC – Trusted Interaction Channel

TSS – TCB System Services

TCSEC – Trusted Computer Security Evaluation Criteria

A.2. Definitions

A.2.1. **Trusted Computing Base**: The Trusted Computing Base is defined as "***The totality of protection mechanisms within a computer system – including hardware, firmware, and software – the combination of which is responsible for enforcing a security policy. A TCB consists of one or more components that together enforce a unified security policy over a product or system. The ability of a trusted***

computing base to correctly enforce a security policy depends solely on the mechanisms within the TCB and on the correct input by system administrative personnel of parameters (e.g., a user's clearance) related to the security policy" [Ref 3.]

A.2.3. **Trusted Path**: The Trusted Path is defined as "***A mechanism by which a person at a terminal can communicate directly with the Trusted Computing Base. This mechanism can only be activated by the person or the Trusted Computing Base and cannot be imitated by untrusted software.***" [Ref 3.]

A.2.3. **Session**: A Session is defined as the period of interaction between a user and entities within the MLS LAN following session activation and until session termination. Sessions are established or denied based upon based on "***attributes such as the location or port or access, the user's security attribute (e.g., identity, clearance level, integrity level, membership in a role), ranges of time (e.g., time-of-day, day-of-week, calendar dates) or combinations of parameters***." Limitations may be placed upon user active sessions such as limitations of the number of multiple concurrent sessions or session locking based upon inactivity. [Ref 4.]

A.2.4. **TCB Authenticated User**: A TCB Authenticated user is one who has successfully established a TCB-to-User connection and been validated by the TCB for operations within the MLS LAN.

A.2.5. **Non-TCB Authenticated User**: A Non-TCB Authenticated user is one who has not been validated by the TCB.

Appendix B. References.

1. Downey, J., Robb, D. *Design of a High Assurance Multilevel Secure Mail Server (HAMMS)*, Naval Postgraduate School, Monterey CA, September 1997.

2. *XTS-300, STOP 4.4.2, Trusted Facility Manual*, Document ID: FS96-371-07, Wang Government Services Inc.

3. *Department of Defense Trusted Computer System Evaluation Criteria*, DoD 5200.28-STD, National Computer Security Center, December 1985.

4. *Common Criteria for Information Technology Security Evaluation Version 2.1*, Common Criteria Project Sponsoring Organisations, August, 1999

5. Bryer-Joyner, S., Heller, S. *Secure Local Area Network Services for a High Assurance Multilevel Network*, Naval Postgraduate School, Monterey, CA. March 1999.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. MLS LAN PROTOCOL HIGH LEVEL ANALYSIS DOCUMENT

This protocol High Level Analysis document provides extensive information concerning the design requirements for each of the six principle connection protocols outlined in the System Requirements Document. The MLS LAN shall provide connection protocols to support the extension of the Trusted Computing Base (TCB) of the MLS LAN to the user through the Trusted Computing Base Extension (TCBE). It is understood that the first four protocols defined in this document are necessary to fulfill near-term functionality and will be addressed in the initial design. Others, when so designated, are recorded to support the long-range goals of the completed project.

Table of Contents

1. Introduction
1.1. Purpose
1.2. Scope
2. The System Protocol OVERVIEW
2.1. MLS LAN Connectivity
3. Connection protocol requirements
3.1. Protected Communications Channel Protocol Requirements
3.2. TCB-to-TCBE Connection Protocol Requirements
3.3. Session Status Protocol
3.4. TCBE-to-Session Server Connection protocol Requirements
3.5. Future Requirement Protocols
4. Appendix
    Appendix A – Abbreviations, Acronyms, and Definitions
    Appendix B – References

Naval Postgraduate School Center for INFOSEC Studies and Research

Multilevel Secure Local Area Network Project

PROTOCOL HIGH LEVEL ANALYSIS DOCUMENT VERSION 1.0

J.D. WILSON

APRIL 20, 2000

Table of Contents

1. INTRODUCTION: 100

1.1. Purpose 100

1.2. Scope 100

2. THE SYSTEM PROTOCOL OVERVIEW: 100

2.1. MLS LAN Connectivity 100

3. CONNECTION PROTOCOL REQUIREMENTS: 106

3.1. Protected Communications Channel Protocol Requirements: 106

3.2. TCB-to-TCBE Connection Protocol Requirements: 106

3.3. Session Status Protocol: 107

3.4. TCBE-to-Session Server Connection protocol Requirements: 107

3.5. Future Requirement Protocols: 107

4. APPENDIX: 109

Appendix A – Abbreviations, Acronyms, and Definitions 109

Appendix B – References 111

1. Introduction:

1.1. PURPOSE.

The purpose of this Protocol High Level Analysis Document is to define the protocol design requirements for the Naval Postgraduate School Center for InfoSec Studies and Research (CISR) Multilevel Secure Local Area Network (MLS LAN) Project. This document is a product of a team effort led by Dr. Cynthia Irvine, Director of NPS CISR. The team members include: Mr. Timothy Levin, Mr. David Shifflett, Ms. Barbara Pereira, LtCol. J.D. Wilson, USMC, and the assistance of Mr. James P. Anderson, of J.P.A. Co.

1.2. SCOPE.

This Protocol High Level Analysis document provides extensive information concerning the design requirements for each of the six principle connection protocols outlined in the System Requirements Document. The MLS LAN shall provide connection protocols to support the extension of the Trusted Computing Base (TCB) of the MLS LAN to the user through the Trusted Computing Base Extension (TCBE). It is understood that the first four protocols defined in this document are necessary to fulfill near-term functionality and will be addressed in the initial design. Others, when so designated, are recorded to support the long-range goals of the completed project.

2. The System Protocol Overview:

2.1. MLS LAN CONNECTIVITY.

The MLS LAN Project is an effort to provide government and commercial organizations with a cost effective, multilevel, easy-to-use office environment leveraging existing high assurance technology. The goals of the project are to produce a networking environment that provides concurrent high assurance access for network users to multiple sensitivity level data through the incorporation of inexpensive commercial personal computers. To ensure positive control over the communications between MLS LAN entities, the definition of certain connection protocols is required. An overview of how these protocols facilitate the MLS LAN connectivity are illustrated in figure 2.1.

2.1.1. Transmission Protection. To provide the high assurance required throughout the network, the Trusted Computing Base (TCB) must provide protection against disclosure and modification of information on all transmissions between components of the MLS LAN. This is accomplished through the establishment of a non-by-passable protected communications channel that provides mutual authentication for the two TCB entities and data encryption on all transmissions between them. This

protected communications channel (PCC) thus presents the protected conduit through which all other MLS LAN protocols may negotiate connectivity.

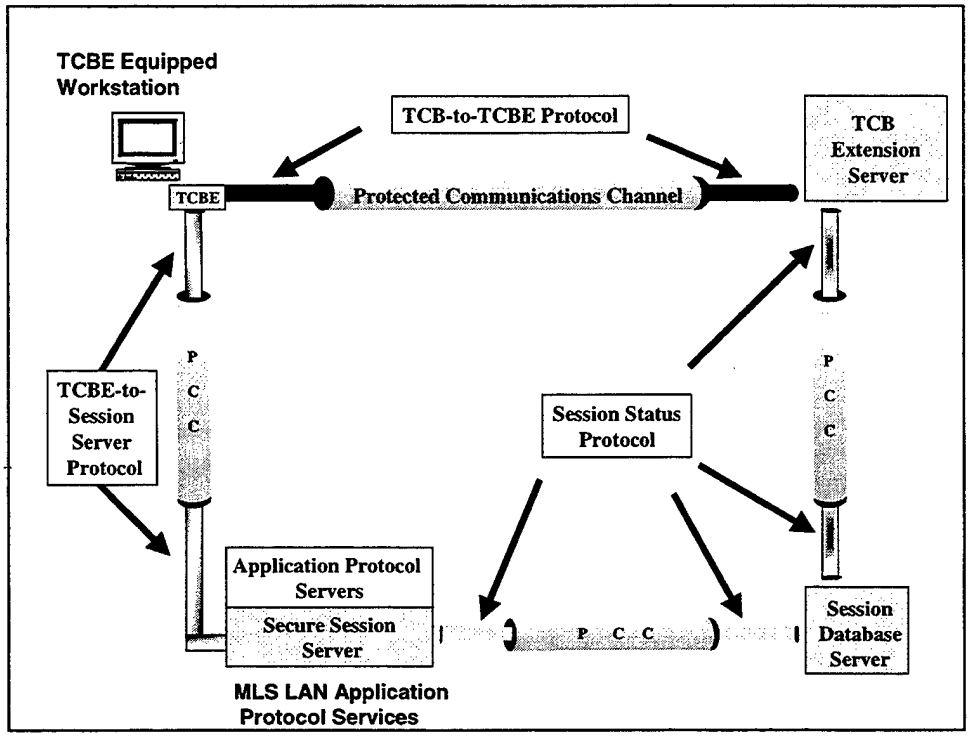

Figure 2.1 MLS LAN Connectivity Overview

2.1.2 Trusted Path Communication. Access to the MLS LAN is controlled through the establishment of a session which requires the user to authenticate him (or her) self to the Trusted Computing Base. This operation, as well as any other security related operations between the user and the TCB must be conducted through a Trusted Path. This requirement is predicated in the Trusted Computer Security Evaluation Criteria (TCSEC) section 3.3.2.1.1 (Trusted Path) which states:

> ***"The TCB shall support a trusted communications path between itself and users for use when a positive TCB-to-user connection is required (e.g., login, change subject security level). Communications via this trusted path shall be activated exclusively by a user of the TCB and shall be logically isolated and unmistakably distinguishable from other paths"*** **[Ref 1.]**

It is also required of the Common Criteria for Information Technology Security Evaluation Version 2.1, under the Trusted Path class (FTP). In general, the Common Criteria states:

> ***"Absence of a trusted path may allow breaches of accountability or access control in environments where untrusted applications are used. These applications can intercept user-private information such as passwords, and use***

***it to impersonate other users. As a consequence, responsibility for any system actions cannot be reliably assigned to an accountable entity. Also, these applications could output erroneous information on an unsuspecting user's display, resulting in subsequent user actions that may be erroneous and may lead to a security breach."* [Ref 2.]**

Specifically, the Common Criteria designates a Trusted Path family (FTP_TRP) for communications between the user and the TCB for use during all security related operations dealing with the establishment, modification and termination of a session.

***"This family defines the requirements to establish and maintain trusted communications to and from users and the TSF [Target of Evaluation Security Functions]. A trusted path may be required for any security-relevant interaction. Trusted path exchanges may be initiated by a user during any interaction with the TS, or the TSF may establish communications with the user via a trusted path."* [Ref 2.]**

The MLS LAN must provide a protocol to support these "Trusted Path" security related operations conducted between a Trusted Computing Base Extension (TCBE) equipped workstation and the TCB. These communications are supported by the TCB-to-TCBE connection protocol.

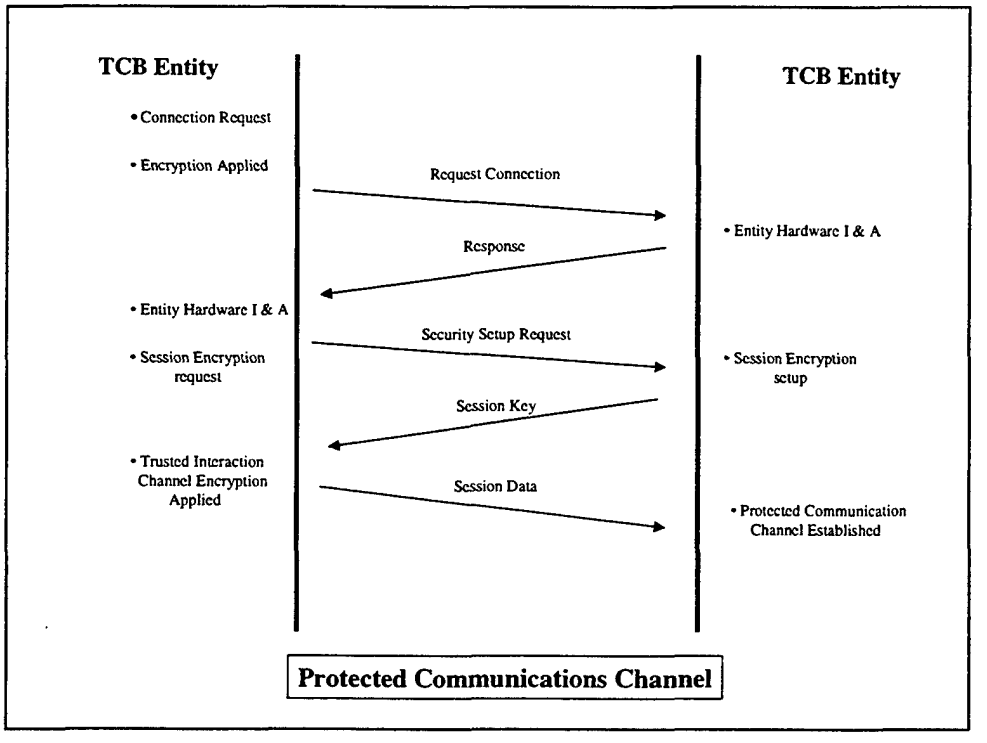

Figure 2.2 Establishment of the Protected Communications Channel

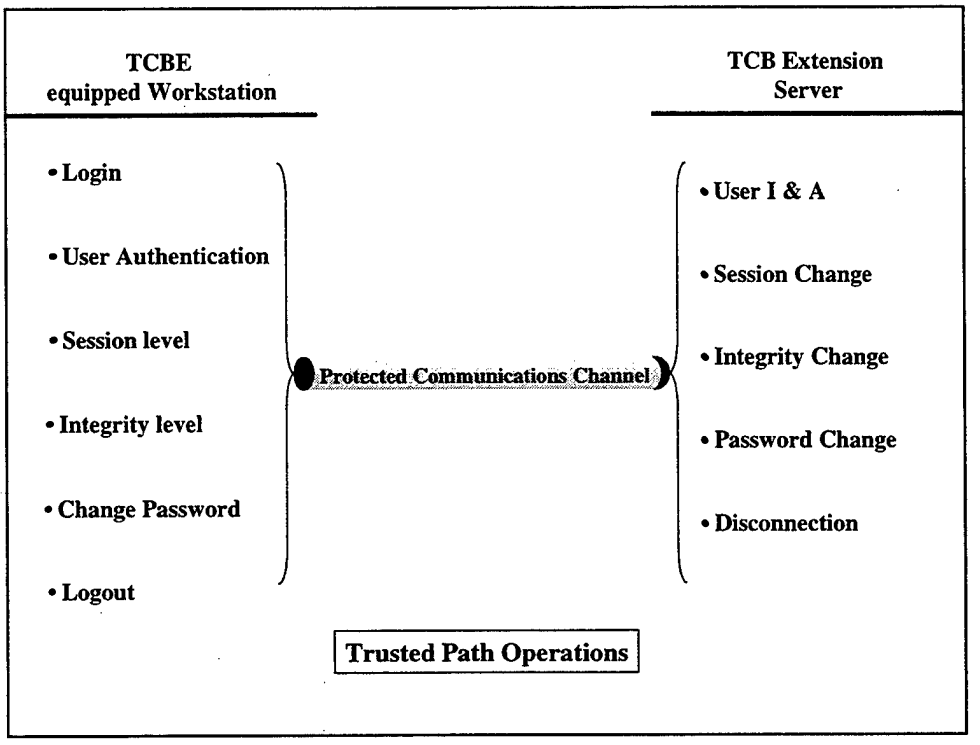

Figure 2.3 Trusted Path Operations conducted through the TCB-To-TCBE Protocol Connection

2.1.3 TCB Session Status Maintenance. The TCB will contain a trusted database server that is responsible for the maintenance of unique information pertinent to all MLS LAN sessions established on the network. The TCB Extension Server utilizing the Session Status Protocol will make all changes and modifications through this Session Database Server.

2.1.4 Network Application Services. Following session establishment, the MLS LAN user will be authorized to conduct normal operations within the MLS LAN environment. This will include connectivity to the Network Application Protocol Services (e.g., HTTP, IMAP, FTP, etc.) supported by the LAN. To ensure the security of the network services connections, application service requests are transmitted from the client to the Secure Session Server handling communications from clients. The Session Server will validate the user's session sensitivity level and access. If the user is authorized, the Session Server will create a socket interface to the Application Protocol Server and allow application operations to commence. [Ref 3.] The Secure Session Server requires a connection protocol that

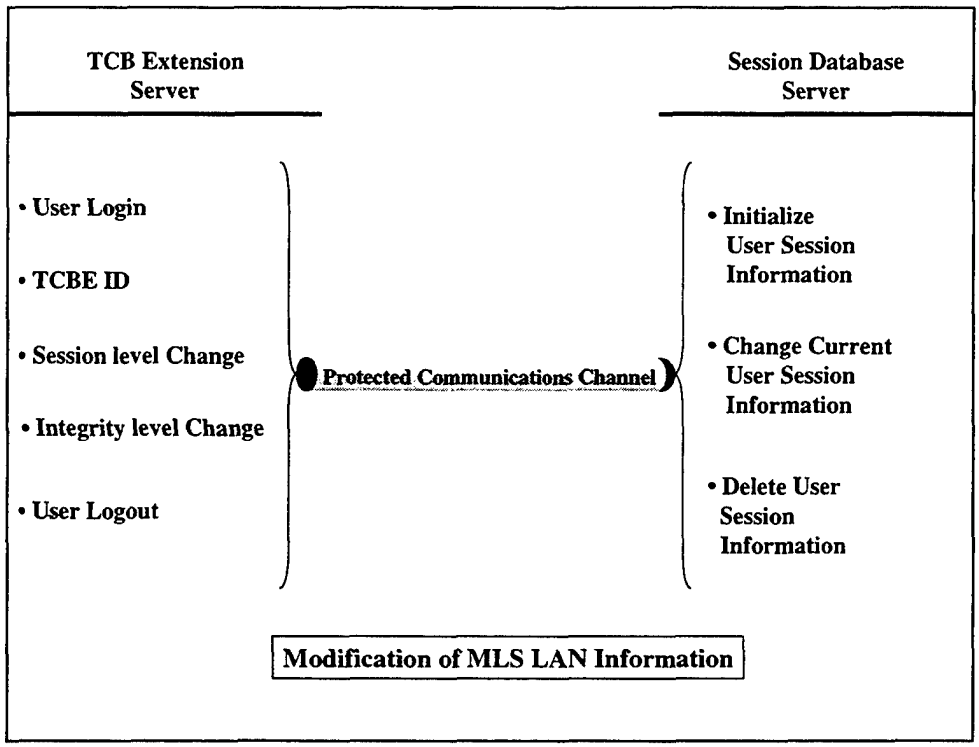

Figure 2.4 TCB Entity Information Modification through the Session Status Protocol Connection

ensures the user is presented services commensurate with the current session established by the TCB. This interaction will be provided by the TCBE-to-Session Server connection protocol.

2.1.5. Application Services Validation. When a service request for access to a MLS LAN Application Protocol Server (APS) is received, a way must be provided to validate the client's current session sensitivity level and service authorization. This validation process must be accomplished prior to the Secure Session Server allowing application operations. The Secure Session Server requires a connection protocol between itself and the TCB Session Status Database in order to compare the information contained in the user's service request and the user's security information maintained by the TCB. The Client Application Services Validation Protocol supports these communications.

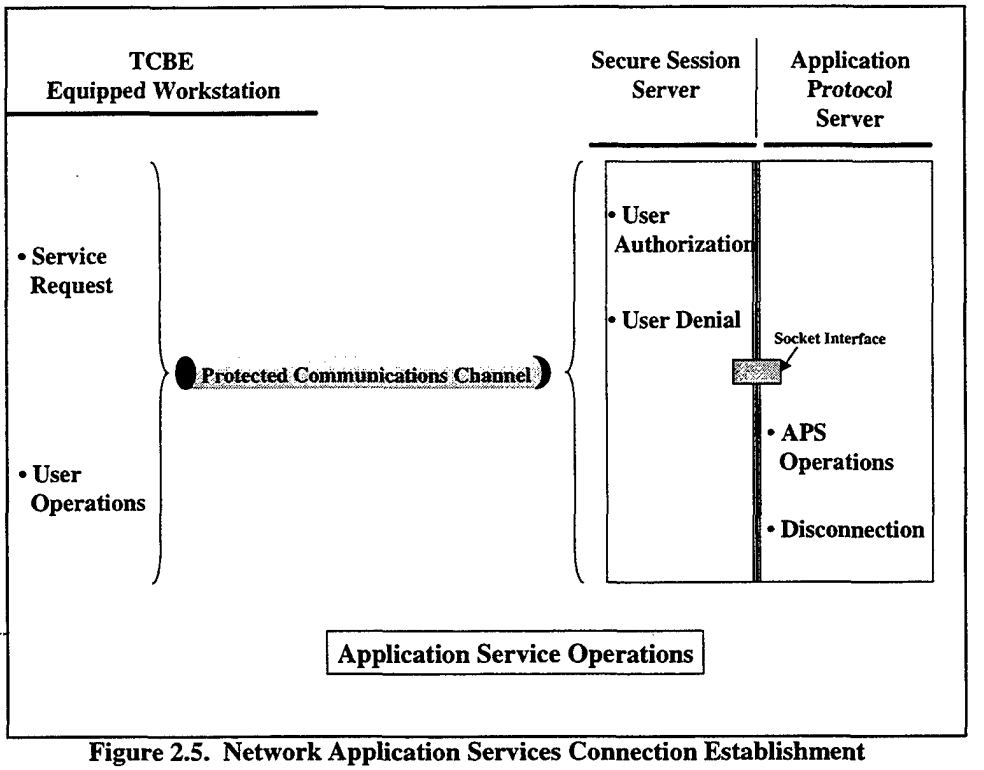

Figure 2.5. Network Application Services Connection Establishment through the TCBE-to-Session Server Protocol

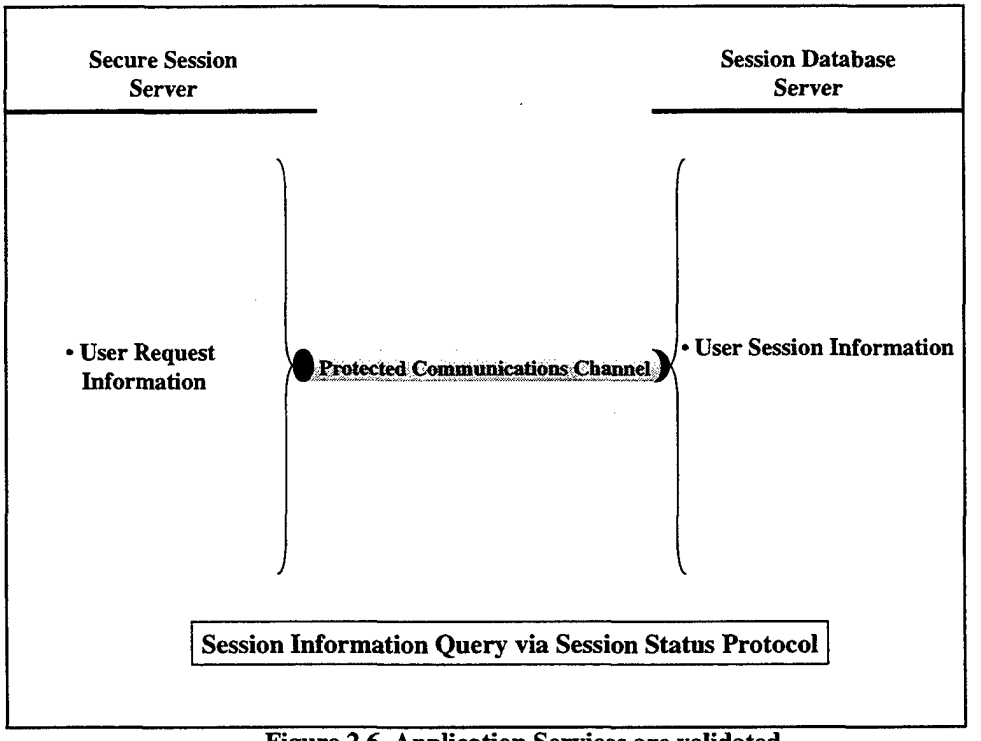

Figure 2.6 Application Services are validated through the Session Status Protocol

2.1.6. Future Requirements. **Future requirements for the MLS LAN involve the connection of workstations that are not using the services of a TCBE. Protocols will be required for the connection of these computers to untrusted Application Protocol Servers (e.g., external HTTP Servers) and to MLS LAN Application Protocol Servers.**

3. Connection Protocol Requirements:

3.1. Protected Communications Channel (PCC) Protocol Requirements.

3.1.1 The PCC Protocol shall enforce mutual "two-way" hardware identification and authentication between two TCB entities prior to the establishment of trusted path communications the trusted communications.

3.1.2 The PCC Protocol shall incorporate security and integrity protection through encryption and verification on all data transmitted between MLS LAN entities.

3.1.3. All connection protocols, e.g., TCBE-to-Session Server, TCB-to-TCBE, shall only be initiated following the establishment of a PCC between the two MLS LAN entities.

3.2. TCB-to-TCBE Connection Protocol Requirements.

3.2.1 The TCB-to-TCBE Protocol shall only be initiated through a request for "secure attention" from the user.

3.2.2 The TCB-to-TCBE Protocol shall support the trusted path security related operations necessary to establish the initial session such as "login" and "user identification and authentication" or for any specified user operations that require a trusted path, such as "logout", "set session level", downgrade, change user password, etc.

3.2.3. The TCB-to-TCBE Protocol shall allow establishment of a session only following activation by the user.

3.2.4 The TCB-to-TCBE Protocol shall control the actions of the TCBE through the specific TCBE state commands.

3.2.5 ***(Future Requirement)*** The TCB-to-TCBE Protocol shall incorporate a "session domination algorithm" to determine if the operating state of the workstation requires modification (e.g., if the requested session sensitivity-level dominates the current session, the workstation operating system need not be cleared).

3.3. Session Status Protocol Requirements.

3.3.1. The Session Status Protocol shall be initiated for every instantiation or modification of any information concerning the status of a user's current session.

3.3.2. The Session Status Protocol shall support trusted communications between the TCB Extension Server and the Session Database Server, which is responsible for the maintenance of user-session security information.

3.3.3 The Session Status Protocol shall support the encapsulation of session information, such as TCBE Identification Number, User Identification, Current Session Level, etc.

3.4. TCBE-to-Session Server Connection Protocol Requirements.

3.4.1. The TCBE-to-Session Server Protocol shall only be initiated following the establishment of an Authorized Session between the client workstation and the TCB.

3.4.2. The TCBE-to-Session Server Protocol shall support the encapsulation of information from the client workstation necessary for the identification and validation of the user's session sensitivity level and application service request.

3.4.3. The TCBE-to Session Server Protocol shall allow communications between a client and an MLS LAN Application Protocol Server only following positive validation of the user's session sensitivity level and the authorization for the specific application service.

3.5. Future Connection Protocol Requirements.

3.5.1. To be Defined.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX

Appendix A. Abbreviations, Acronyms, and Definitions.

A.1. Abbreviations, Acronyms

APS – Application Protocol Server

CISR – Center for InfoSec Studies and Research

FTP_TRP – Common Criteria Trusted Path Family

FTP – File Transfer Protocol

HTTP – Hyper Text Transfer Protocol

IMAP – Internet Message Access Protocol

LAN – Local Area Network

MLS – Multilevel Secure

NPS – Naval Postgraduate School

SAK – Secure Attention Key

TCB – Trusted Computing Base

TCBE – Trusted Computing Base Extension

TCSEC – Trusted Computer Security Evaluation Criteria

PCC – Trusted Interactive Channel

TSF – Target of Evaluation Security

A.2 Definitions

A.2.1. **Trusted Computing Base**: The Trusted Computing Base is defined as "***The totality of protection mechanisms within a computer system – including hardware, firmware, and software – the combination of which is responsible for enforcing a security policy. A TCB consists of one or more components that together enforce a unified security policy over a product or system. The ability of a trusted computing base to correctly enforce a security policy***

depends solely on the mechanisms within the TCB and on the correct input by system administrative personnel of parameters (e.g., a user's clearance) related to the security policy" [Ref 1.]

A.2.3. **Trusted Path**: The Trusted Path is defined as "***A mechanism by which a person at a terminal can communicate directly with the Trusted Computing Base. This mechanism can only be activated by the person or the Trusted Computing Base and cannot be imitated by untrusted software.***" [Ref 1.]

A.2.3. **Session**: A Session is defined as the period of interaction between a user and entities within the MLS LAN following session activation and until session termination. Sessions are established or denied based upon based on "***attributes such as the location or port or access, the user's security attribute (e.g., identity, clearance level, integrity level, membership in a role), ranges of time (e.g., time-of-day, day-of-week, calendar dates) or combinations of parameters***." Limitations may be placed upon user active sessions such as limitations of the number of multiple concurrent sessions or session locking based upon inactivity. [Ref 2.]

A.2.4. **TCB Authenticated User**: A TCB Authenticated user is one who has successfully established a TCB-to-User connection and been validated by the TCB for operations within the MLS LAN.

A.2.5. **Non-TCB Authenticated User**: A Non-TCB Authenticated user is one who has not been validated by the TCB.

Appendix B. References.

1. *Department of Defense Trusted Computer System Evaluation Criteria*, DoD 5200.28-STD, National Computer Security Center, December 1985.

2. *Common Criteria for Information Technology Security Evaluation Version 2.1*, Common Criteria Project Sponsoring Organisations, August, 1999

3. Bryer-Joyner, S., Heller, S. *Secure Local Area Network Services for a High Assurance Multilevel Network*, Naval Postgraduate School, Monterey, CA. March 1999.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. MLS LAN CONNECTION FRAMEWORK DOCUMENT

This document provides an overview of the connection protocols required to establish a session and operate within the MLS LAN. Its intent is to provide a coherent description of the datagram formats and state transitions used to communicate within the MLS LAN. This document is in support of the Naval Postgraduate School Center for INFOSEC Studies and Research Multilevel Secure Local Area Network (MLS LAN) Project.

Table of Contents

1. Introduction
2. The MLS LAN Protected Communications Channel
3. TCB-TCBE Connection Protocol
4. Session Status Protocol
5. TCBE-to-Session Server Connection Protocol
6. References

Naval Postgraduate School Center for INFOSEC Studies and Research

Multilevel Secure Local Area Network Project

MLS LAN CONNECTION FRAMEWORK VERSION 1.0

J.D. WILSON

JUNE 12, 2000

NPS CISR
MLS LAN PROJECT

J.D. Wilson
Naval Postgraduate School

The MLS LAN Project Connection Framework

Status of Memo

This document provides an overview of the connection protocols required to establish a session and operate within the MLS LAN. Its intent is to provide a coherent description of the datagram formats and state transitions used to communicate within the MLS LAN. This document is in support of the Naval Postgraduate School Center for INFOSEC Studies and Research Multilevel Secure Local Area Network (MLS LAN) Project. Distribution of this memo is at the discretion of Dr. Cynthia Irvine.

Acknowledgements

The information to develop this protocol framework has been the result of an engineering team effort researching the definition of the MLS LAN's architecture, its components, and connectivity. The team members include: Dr. Cynthia Irvine, Director of NPS CISR, Mr. Timothy Levin, Mr. David Shifflett, Ms. Barbara Pereira, LtCol. J.D. Wilson, USMC, and the assistance of Mr. James P. Anderson, of J.P.A. Co.

Table of Contents

1. Introduction 117
   1.1 Summary of Contents of the Document 117
   1.2 Terminology 117
   1.3 Related Documents 118
2. The MLS LAN Protected Communications Channel 119
   2.1 Overview 119
   2.2 Logical Placement of IPSec for the MLS LAN 119
   2.3 IPSec Security Policy for the MLS LAN 120
   2.4 IPSec Key Management for the MLS LAN 121
   2.5 MLS LAN Protected Communications Channel Processing 121
3. TCB-TCBE Connection Protocol 123
   3.1 Introduction 123
   3.2 TCBE States 123
      3.2.1 TCBE State Variables 123
      3.2.2 TCBE Disallowed States 124
      3.2.3 TCBE Allowable States 124
   3.3 TCB Extension Server States 124
      3.3.1 TCB Extension Server State Variables 125
      3.3.2 TCB Extension Server Disallowed States 126
      3.3.3 TCB Extension Server Allowable States 127

3.4 TCBE-to-TCB Extension Server Protocol Datagram Format..... 127
3.4.1 TCBE to TCB Extension Server Datagram Field Descriptions.. 127
3.4.2 TCB Extension Server to TCBE Datagram Field Descriptions.. 129
3.4.3 TCBE-to-TCB Extension Server Protocol Datagram Packaging 130
3.5 TCBE-to-TCB Extension Server Interaction..... 131
3.5.1 TCBE State Options and Transitions..... 131
3.5.2 TCB Extension Server State Options and Transitions..... 134
4. Session Status Protocol..... 139
4.1 Introduction..... 139
4.2 TCB Extension Server States..... 139
4.3 Session Database Server States..... 139
4.4 Session Status Protocol Datagram Format..... 139
4.4.1 TCB Extension Server to SDS Datagram Field Descriptions.... 140
4.4.2 SDS to TCB Extension Server Datagram Field Descriptions..... 141
4.4.3 Session Status Protocol Datagram Packaging..... 142
4.5 SDS to TCB Extension Server Interaction..... 142
4.5.1 TCB Entity State Options 142
4.5.2 Session Database Server Options 144
4.5.3 Session Database Server Response..... 144
5. TCBE-to-Session Server Connection Protocol..... 147
5.1 Introduction..... 147
5.2 TCBE States..... 147
5.3 Secure Session Server States..... 147
5.4 TCBE-to-Session Server Protocol Datagram Format..... 147
5.4.1 TCBE-to-Session Server Datagram Field Descriptions..... 148
5.4.2 Application Protocol Service Request Packet..... 148
5.4.3 TCBE-to-Session Server Protocol Datagram Packaging..... 148
5.5 TCBE-to-Secure Session Server Interaction..... 149
5.5.1 TCBE State Options 149
5.5.2 Secure Session Server State Options 149
6. References..... 151

1. Introduction

1.1 Summary of Contents of the Document

The MLS LAN Project incorporates specific connection protocols to provide communications between the MLS LAN components. This framework provides an overview of these protocols with respect to the information contained in their defined datagrams and how the information that is passed is used by the components to effect state transitions. This document does not address all aspects of the MLS LAN architecture. Subsequent documents and established Requests For Comments (see Section 1.3) will address the architectural details of a more advanced nature.

The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL, when they appear in this document, are to be interpreted as described in RFC 2119 [BRA97].

1.2 Terminology

APS – Application Protocol Server: An untrusted, industry standard application protocol server that provides higher layer application services to MLS LAN users.

MLS – Multilevel Secure: Computer system[s] containing information with different sensitivities that simultaneously permits access by users with different security clearances and need-to-know, but prevents users from obtaining access to information for which they lack authorization.

NPS – Naval Postgraduate School

PCC – Protected Communications Channel: An IPSec secured conduit through which all other MLS LAN connection protocols operate.

PCI – Protected Channel Initiator: A trusted process within the network layer of MLS LAN high assurance servers and TCBEs that provides security services to create a Protected Communications Channel.

SAK – Secure Attention Key: A specified key[s] that when activated will cause a TCBE-equipped MLS LAN workstation to disconnect with all untrusted applications and connect to the TCB.

SDS – Session Database Server: A trusted process within the MLS LAN TCB that manages the session status data for all users logged into the MLS LAN.

SSS – Secure Session Server: A trusted process within the MLS LAN TCB that provides connectivity for users to Application Protocol Servers.

TCB – Trusted Computing Base: A Trusted Computing Base is the collection of security-related elements of a computer system that is responsible for enforcing a security policy.

TCBE – Trusted Computing Base Extension: An high assurance enhanced network interface card (NIC) that is installed into the MLS LAN workstation to support the extension of the TCB.

TCB Extension Server – Trusted Computing Base Extension Server: A trusted process within the MLS LAN TCB that conducts the user identification and authentication (I&A) and session negotiation necessary to access the MLS LAN.

Workstation – MLS LAN Client Workstation: A commercially personal computer.

1.3 Related Documents

As mentioned above, other documents provide detailed information as to the specifics of the MLS LAN architecture, specifications and requirements. Additionally, as the Protected Communications Channel is constructed using the IPSec standard, it refers to the following documents and RFCs.

a. MLS LAN system requirements – "MLS LAN System Requirements Document" [WIL00a].
b. MLS LAN protocol high level analysis – "MLS LAN Protocol High Level Analysis Document" [Wil00b].
c. MLS LAN Draft Design Document [Shif00]
d. IPSec architecture – "IP Security Document Roadmap" [TDG97], "Security Architecture for the Internet Protocol" [KA98a].
e. Security protocols – RFCs describing the Authentication Header (AH) [KA98b] and Encapsulating Security Payload (ESP) [KA98c].
f. Algorithms for authentication and encryption – a separate RFC for each algorithm.
g. Automatic key management – RFCs on "The Internet Key Exchange" (IKE) [HC98], "Internet Security Association and Key Management Protocol" (ISAKMP) [MSST97], and "The Internet IP Security Domain of Interpretation for ISAKMP" [Pip98].

2. The MLS LAN Protected Communications Channel Protocol

2.1 Overview

The Protected Communications Channel protocol is used to establish the security conduit through which all other MLS LAN protocols must operate. The Protected Communications Channel is created through the use of IP layer security as defined in the IP Security Standard for the Internet (See Section 1.3 for pointers to the references). The channel provides the MLS LAN with a trusted channel that enforces a "two-way" mutual hardware authentication between the two connecting entities and provides security and integrity protection on all transmitted data. The use of this channel also provides some fault tolerance protection in the event of component loss, as the communications between the two Protected Communications Channel connected entities will cease, but the overall network will not be affected.

Since the MLS LAN utilizes the IPSec Standard to provide the framework for this channel, this document does not attempt to describe its architecture or mechanisms. Information of these topics can be found in the many RFCs that describe IPSec. Additionally, the specific design of the Protected Communications Initiator and data structures necessary for IPSec implementation in the MLS LAN have yet to be finalized. For this reason, the subsequent sections will, provide an approach to be taken in the application of IPSec in the MLS LAN to create a Protected Communications Channel.

2.2 Logical Placement of MLS LAN IPSec

[KA98a] describes three common ways in which IPSec can be implemented in hosts, routers and security gateways.

a. Integration into the native IP layer implementation of the host. This requires access to the IP source code for the entity that is to use IPSec.
b. "Bump-in-the-Stack" (BITS) implementation places the IPSec underneath an existing implementation of the IP protocol stack between the native IP and the local network drivers. This implementation does not require access to the IP source code utilized in the host.
c. "Bump-in-the-Wire" (BITW) implementation places an outboard crypto processor that provides the IPSec security services.

The MLS LAN presently utilizes the Wang Government Services, Inc. XTS-300™ high assurance server as its source host and has created a prototype TCBE utilizing the Intel™ i960 processor. To maintain simplicity of the XTS-300 security kernel, it is recommended that the MLS LAN implement IPSec in a BITS configuration and create the Protected Communications Initiator as user defined trusted code to be controlled by the security kernel.

2.3 IPSec Security Policy for the MLS LAN

Each connection to the MLS LAN TCB must be encrypted with an algorithm that is suitable to protect the transmitted information. The Security Manager is responsible for ensuring that the strength of the assigned encryption mechanisms are sufficient to protect the given sensitivity level. Once assigned, the TCB will maintain a virtual table that maps the available encryption transforms with the sensitivity levels they can support. When encrypted, the information is considered to be safe for transmission across any medium until it reaches its intended recipient. The recipient's act of decryption once again transforms the information into a sensitive form. IPSec provides a mechanism through the Security Policy Database and Security Association Database to segregate the application of protection based upon a set of given attributes [KA98a]. The MLS LAN Security Manager will create a listing of the specific security parameters that a Protected Communications Channel must enforce for connection to each of the MLS LAN entities. These security parameters will be mapped to the listing of available MLS LAN session levels enabling the TCB Extension Server to know the Security Policy Database (SPD) assignments for each session level.

The initial Security Policy Database of the TCBE will be placed in non-volatile memory, established by the Security Manager with a single entry: to apply security to connect to the TCB Extension Server and disallow all other connections. Once a session has been established, the TCB Extension Server will update the TCBE SPD with the security connection information commensurate with the sensitivity level negotiated for the session. From this Security Policy Database, the TCBE will correctly negotiate all other connections to MLS LAN hosts utilizing the standard Security Association setup of ISAKMP [MSST97]. Additional encryption algorithms or transforms can be developed to provide higher levels of encryption, e.g., NSA approved Type I encryption, for use on the MLS LAN. This remote management of the security policy of IPSec is available only because the MLS LAN TCBE can create the initial Protected Communications Channel at system high through the non-volatile Security Policy Database placed on the TCBE.

A future requirement for the MLS LAN allows a TCBE-equipped workstation to operate as a Non-MLS LAN workstation, e.g., connect to untrusted protocol servers without first connecting to the MLS LAN TCB. In this situation, an additional Security Policy Database and Security Association Database may be required to establish "untrusted" (normal) IPSec security associations to commercial sites. The design and implementation of these mechanisms is left to future work.

2.4 IPSec Key Management for the MLS LAN

The MLS LAN will use the standard Internet Key Exchange (IKE) [HC98] to define a key exchange and to negotiate security services to be provided for each Protected Communications Channel. IKE uses a predefined domain of interpretation (DOI) to outline the required and optional attributes that are negotiated during the phase two exchanges. Currently the DOI is written specifically for use with the ISAKMP [Pip98]. This DOI may be sufficient to provide the security attributes necessary for use in an MLS environment, however, future research may discover that a specific DOI is needed for the MLS LAN Project.

2.5 MLS LAN Protected Communications Channel Processing

The first Protected Communications Channel established must be a connection between the TCBE-equipped workstation and the source host running the TCB Extension Server process. This is initiated by the TCBE once the user requests attention from the TCB by activating a SAK. The PCI process on the TCBE will use the initial Security Policy Database setting to establish the IKE phase one exchanges and establish a secure and authenticated communications channel between the TCBE and the TCB Extension Server host. Once the IKE security association (SA) has been established, the phase two negotiations can then be sent to generate the appropriate incoming and outgoing IPSec SAs. This exchange effectively negotiates the specific AH and ESP selectors required for each SA. During these exchanges, the selectors are outlined for the unique SA and each entity records the SA information into its Security Association Database under a unique Security Parameter Index.

Once the Protected Communications Channel is established between the TCBE and the TCB Extension Server, the user will be allowed to login to the MLS LAN and negotiate a session. If the session establishment is successful, the TCB Extension Server will issue a "PCC Update" command and transfer the appropriate session level Security Policy data to the TCBE for inclusion in its Security Policy Database, as well as make available in the SPD the entries for communicating with other MLS LAN Components, e.g., Application Protocol Servers.

From this point, the user is logged in and operating on the MLS LAN at the negotiated session level. As application protocol services are requested, the TCBE Protected Communications Initiator will use the same method as above to create a separate Protected Communications Channel to the source host that supports the requested application protocol server.

THIS PAGE INTENTIONALLY LEFT BLANK

3. TCB-TCBE Connection Protocol

3.1 Introduction

The TCB-TCBE Connection protocol is used to provide the Trusted Computing Base (TCB) with a method to conduct security related operations along a trusted path. This protocol is used by the TCBE as a method to gain secure attention from and to respond to the commands of the TCB. The protocol also provides the TCB Extension Server with a method to control the actions of the TCBE through the use of specific TCBE state commands. The TCB-TCBE Connection protocol will only be initiated through a request for "secure attention" from the user. Protection against replay and spoofing is provided by the underlying Protected Communications Channel.

3.2 TCBE States

The TCBE will use input such as the activation of the Secure Attention Key or Commands received from the TCB Extension Server to change its configuration. This configuration is commonly referred to as the current state of the TCBE. This section will describe the TCBE allowable states and the values of the corresponding state variables.

3.2.1 TCBE State Variables

The TCBE has 3 different state variables, or indicators as shown in Table 1. The variable "Power" indicates that the TCBE is powered and active. The variable "Trusted Path Operations" represents connectivity with the TCB and the negotiation of a secure session. The variable "Client OS Loaded" indicates that the client workstation's memory has been purged and a fresh copy of the operating system has been loaded. This provides a total of 2^3 or 8 possible states.

Description	Values	Abbreviation
Power	On/Off	Power
Trusted Path Operations	Yes/No	TPO
Client OS Loaded	Yes/No	OS

Table 1. TCBE State Variables

3.2.2 TCBE Disallowed States

The following states are disallowed, meaning there is no way to transition into the state. These occur when the system power is 'Off', and any other state is 'Yes'. There are a total of 3 disallowed states as shown in Table 2.

Power	TPO	OS	Reason for disallowed state
Off	Yes	No	No TPO w/o Power
Off	No	Yes	No O/S w/o Power
Off	Yes	Yes	No TPO or O/S w/o Power

Table 2. TCBE Disallowed States

3.2.3 TCBE Allowable States

There are a total of five allowable states as shown in Table 3. In the case of the activation of a SAK from State [2], where the O/S might have been previously loaded, the TCBE will purge the previous copy and reload the O/S following a successful login and transition to State [4]. If the login is unsuccessful the TCBE should return to its previous state (State [2]). The TCBE States are depicted in Figure 1.

Future work should include a method of login at "system low" that allows the TCB Extension Server knowledge of the user login but not force a purge of the O/S. For example, this would allow a user who as been using a workstation in an unprotected mode, to access the MLS LAN at the lowest possible sensitivity level and utilize print services without a system purge at login.

State Number	Power	TPO	OS	Name
0	Off	No	No	Power Off
1	On	No	No	Idle
2	On	No	Yes	Untrusted Operations
3	On	Yes	No	Trusted Processing
4	On	Yes	Yes	Trusted Session

Table 3. TCBE Allowable States

3.3 TCB Extension Server States

The TCB Extension Server will use input such as the receipt of a Secure Attention Request, or response payload type received from the TCBE to change its

configuration. This configuration is commonly referred to as the current state of the TCB Extension Server. This section will describe the TCB Extension Server allowable states and the values of the state variables for each.

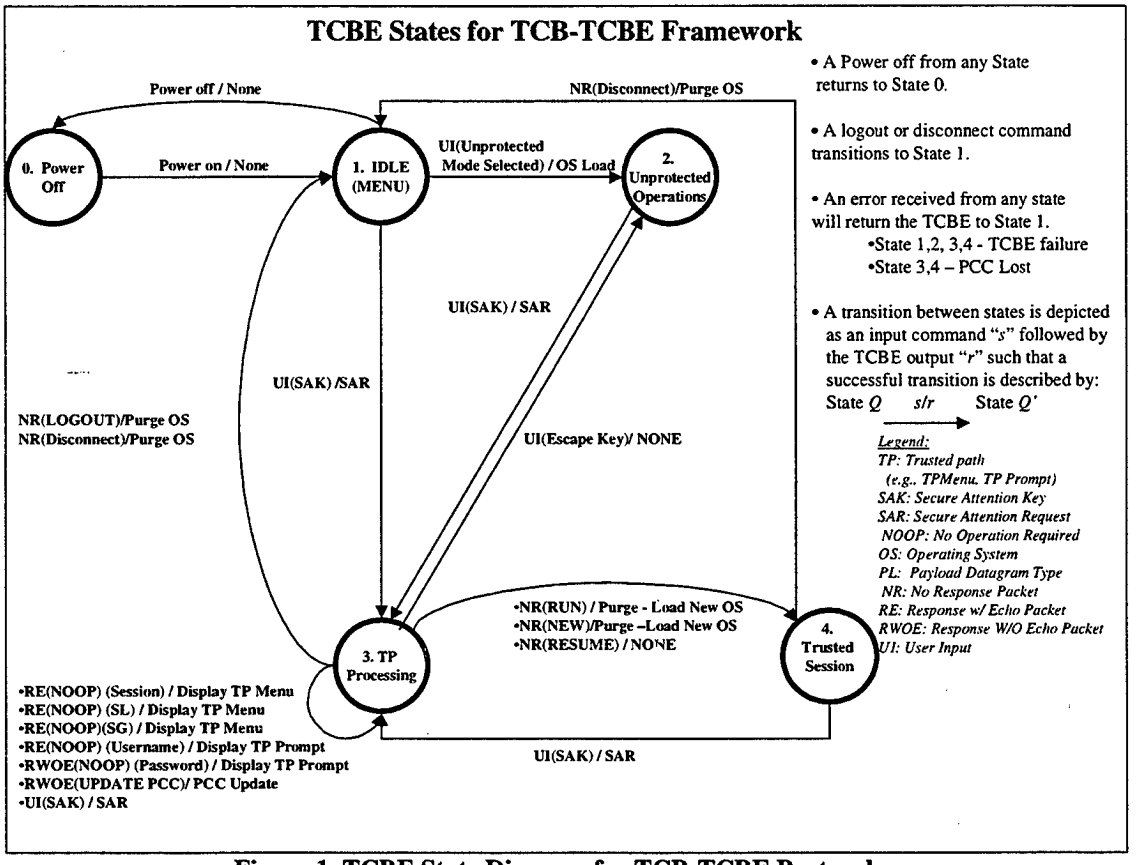

Figure 1. TCBE State Diagram for TCB-TCBE Protocol

3.3.1 TCB Extension Server State Variables

The TCB Extension Server has 5 different state variables, or indicators as shown in Table 4. The variable "Power" indicates that the TCB Extension Server is powered and active. The variable "Connected to TCBE " represents logical connectivity with the TCBE. The variable "User Logged in" indicates that the User has successfully completed I&A within the TCB. The variable "Session Operations" indicates that the User has successfully negotiated a session security level within the TCB. The variable "Level Changed" indicates that the User has changed his session level within the TCB. This provides a total of 2^5 or 32 possible states.

Description	Values	Abbreviation
Power	On/Off	Power
Connected to TCBE	Yes/No	Connect
User Logged in	Yes/No	Log
Session Operations	Yes/No	Session
Level Change	Yes/No	Level

Table 4. TCB Extension Server State Variables

3.3.2 TCB Extension Server Disallowed States

The following states are disallowed, meaning there is no way to transition into the state. These occur when the system power is 'Off', and any other state is 'Yes'. This accounts for 15 of the possible states. The reason for the other disallowed states is presented in Table 5. There are a total of 26 disallowed states.

Power	Connect	Log	Session	Level	Reason for disallowed state
Off	Yes/No	Yes/No	Yes/No	Yes/No	No Power: Total States disallowed is 15
On	No	No	No	Yes	No Level w/o Connection
On	No	No	Yes	No	No Session w/o Connection
On	No	No	Yes	Yes	No Session or Level w/o Connection
On	No	Yes	No	No	No Log w/o Connection
On	No	Yes	No	Yes	No Log or Level w/o Connection
On	No	Yes	Yes	No	No Log or Session w/o Connection
On	No	Yes	Yes	Yes	No Log, Session, or Level w/o Connection
On	Yes	No	No	Yes	No Level w/o Login
On	Yes	No	Yes	No	No Session w/o Login
On	Yes	No	Yes	Yes	No Session or Level w/o Login
On	Yes	Yes	No	Yes	No Level w/o Session

Table 5. TCB Extension Server Disallowed States

3.3.3 TCB Extension Server Allowable States

Of the original 32 possible states, 26 cannot be reached. This leaves a total of 6 possible states for the TCB Extension Server as shown in Table 6. The TCB Extension Server state diagram is depicted in Figure 2.

State Number	Power	Connect	Log	Session	Level	Name
0	Off	No	No	No	No	Power Off
1	On	No	No	No	No	Idle
2	On	Yes	No	No	No	Connected
3	On	Yes	Yes	No	No	Logged in
4	On	Yes	Yes	Yes	No	Running
5	On	Yes	Yes	Yes	Yes	Trusted Session Processing

Table 6. TCB Extension Server Allowable States

3.4 TCBE-to-TCB Extension Server Protocol Datagram Format

There are two defined datagram formats for the protocol. The first, shown in Figure 3, is the "Payload" datagram used to convey information and requests from the TCBE to the TCB Extension Server. The second, shown in Figure 4, is the "Command" datagram provided to enable the TCB Extension Server to control the TCBE State actions and convey information to the TCBE.

3.4.1 TCBE to TCB Extension Server Datagram Field Descriptions

The following subsections define the fields that comprise the TCBE-to-TCB Extension Server Datagram or "Payload Datagram" as depicted in Figure 3. All fields described are mandatory, i.e., they are always present in the TCB-TCBE Connection Protocol.

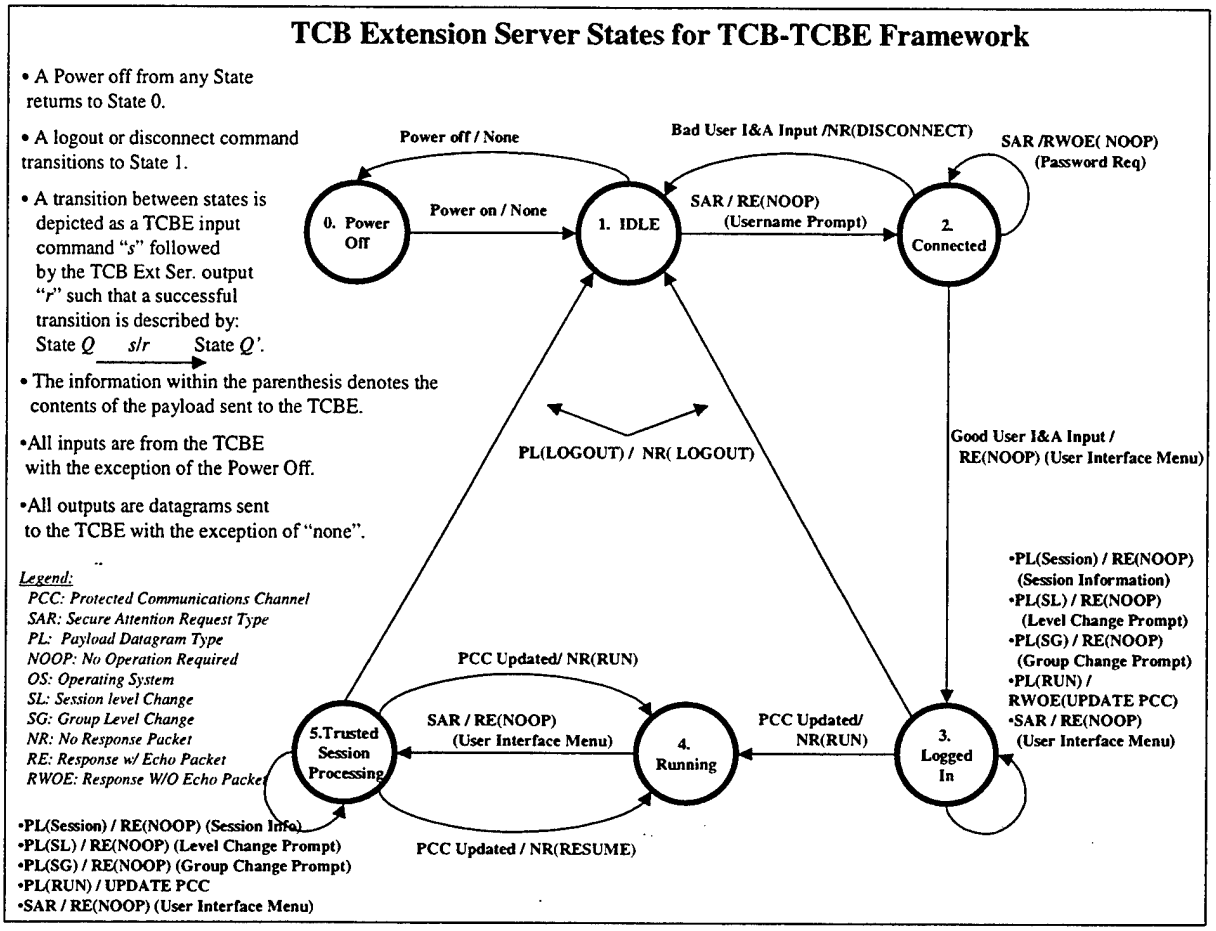

Figure 2. TCB Extension Server States for TCB-TCBE Framework

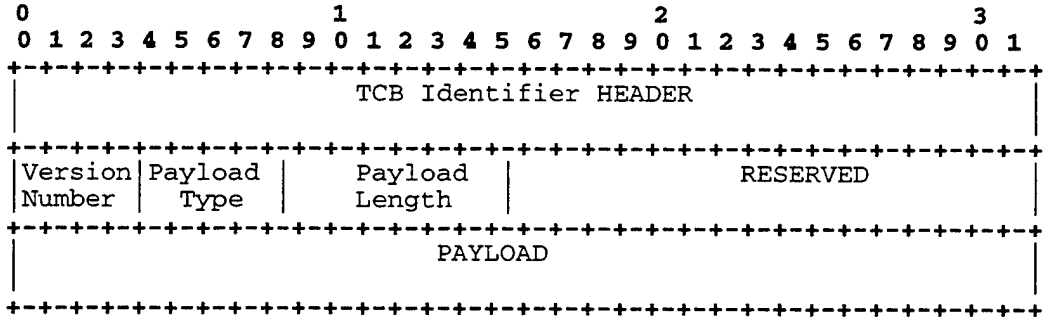

Figure 3. TCBE-to-TCB Extension Server Payload Datagram Format

a. TCB Identifier Header.
This is a 32-bit value that identifies the TCBE that created the packet. This will be used by the TCB to facilitate Hardware Identification and Authentication.

b. Version Number Field.
The Version Number field is a 4-bit value that identifies to the version of the MLS LAN Protocol employed. The value in this field for the Version 1 of this protocol will be set to 1.

c. Payload Type.
This is a field is a 4-bit value that identifies the type of payload contained in the datagram. The value of this field is chosen from the listing below. 16 payload types are possible, however, in the current version only 3 are defined.

- *Value 0 -- Secure Attention Request*
- *Value 1 -- Response*
- *Value 2 – PCC Updated*

d. Payload Length:
This field is an 8-bit field specifying the length of the payload in 32 bit words.

e. Reserved:
This 16-bit field is reserved for future use. It should be ignored by the receiving TCB Entity, but is best set to "zero".

f. Payload:
This is a variable length field that contains the data to be sent to the TCB Extension Server, typically, this will be the input from the user. The payload may be padded with "zeros" to fill up the last 32-bit word.

3.4.2 TCB Extension Server to TCBE Datagram Field Descriptions

The following subsections define the fields that comprise the TCB Extension Server-to-TCBE datagram or "Command datagram" as depicted in Figure 4. The TCB Identifier, Version Number, Payload Length and Reserved fields are the same as described in Section 3.4.1 and will not be repeated here. All fields in the datagram, however are mandatory, i.e., they are always present in the TCB-TCBE Connection Protocol.

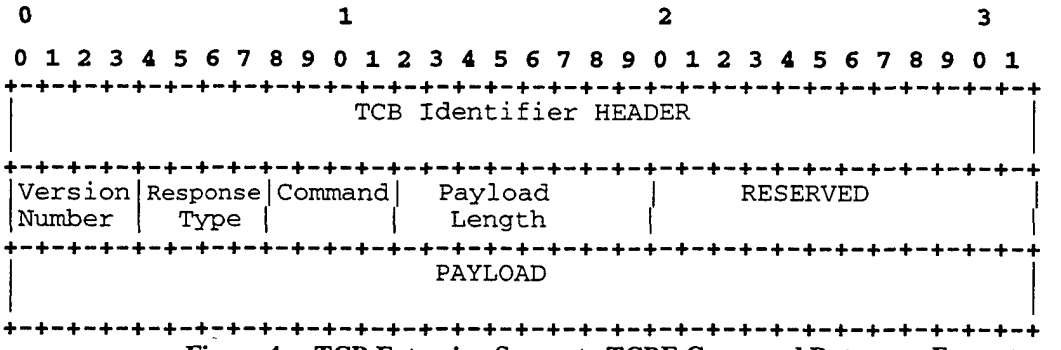

Figure 4. TCB Extension Server to TCBE Command Datagram Format

a. Response Type:
This is a field is a 4-bit value that identifies the type of response that the TCB Extension Server will expect in response to the datagram. The value of this field is chosen from the listing below. 16 response types are possible, however, in the current version only 3 are defined.

- *Value 0 -- No Response*
- *Value 1 -- Response with Echo*
- *Value 2 -- Response without Echo*

b. Command:
This field is a 4-bit value that identifies the command that the TCB Extension Server is issuing to the TCBE. The value of this field is chosen from listing below. 16 command types are possible, however, in the current version only 7 are defined.

- *Value 0 -- NOOP*
- *Value 1 -- RUN*
- *Value 2 -- NEW*
- *Value 3 – PCC UPDATE*
- *Value 4 -- RESUME*
- *Value 5 -- LOGOUT*
- *Value 6 -- DISCONNECT*

c. Payload:
This is a variable length field that contains the data to be presented to the User or information to update the TCBE itself. For example, the RUN or LOGOUT commands may use the payload to pass confirmation information to the user, while the PCC UPDATE command uses the payload to pass Protected Communications Channel Database updates to the TCBE. The payload may be padded with "zeros" to fill up the last 32-bit word.

3.4.3 TCBE-to-TCB Extension Server Protocol Datagram Packaging.

The TCBE and TCB Extension Server will generate one of the types of TCB-TCBE datagram packets described in Sections 3.4.1 or 3.4.2 for all secure operation communications within the TCB. The datagram will be created by either the TCBE or TCB Extension Server and passed to the lower layers

protocols for transmission to the other entity. Since the protocol is created using fixed fields, the value in these datagram fields need no manipulation and can be parsed for use. The packaging used for transmission of each of the MLS LAN Protocols is depicted in Figure 5.

3.5 TCBE to TCB Extension Server Interaction.

This section describes the uses of the TCB-TCBE Connection Protocol. Prior to use of the protocol both the TCBE and TCB Extension Server must be powered and in at least State [1] Idle. As previously mentioned, only the user through the use of the Secure Attention Key, can initiate the TCB-TCBE Connection Protocol. With the exception of the "UPDATE PCC" command, the TCBE will display the payload to the user for all datagrams received. If a datagram is received with a "Response" type of "RESPONSE WITHOUT ECHO", the TCBE will wait for input from the user without echoing the input to the screen. User input can be interrupted at any time by the following actions:

- Power Off
- Activation of a Secure Attention Key
- Activation of the "Escape" key.

3.5.1 TCBE State Options and Transitions

The TCBE has the capability of generating only three types of TCB-TCBE Protocol Datagrams. They are as follows:

- **Secure Attention Request**. The TCBE will generate and transmit a Secure Attention Request packet (as described in Section 3.4.1) for each use of the Secure Attention Key by the user, regardless of its current state. This action will transition the TCBE into State [3] (TP Processing) and initialize a Protected Communications Channel or "Trusted Path" to the TCB if one does not already exist.
- **Response**. The TCBE will generate and transmit a Response Packet when the TCB Extension Server requires a response. The TCBE will remain in State [3] (TP Processing) and wait for input from the user. It will then generate and transmit a Response Datagram packet (as described in Section 3.4.1).
- **PCC Updated**. The TCBE will generate and transmit a PCC Updated packet (as described in Section 3.4.1) following the successful creation or update of the Protected Communications Channel Security Policy Database from the information provided by the TCB Extension Server.

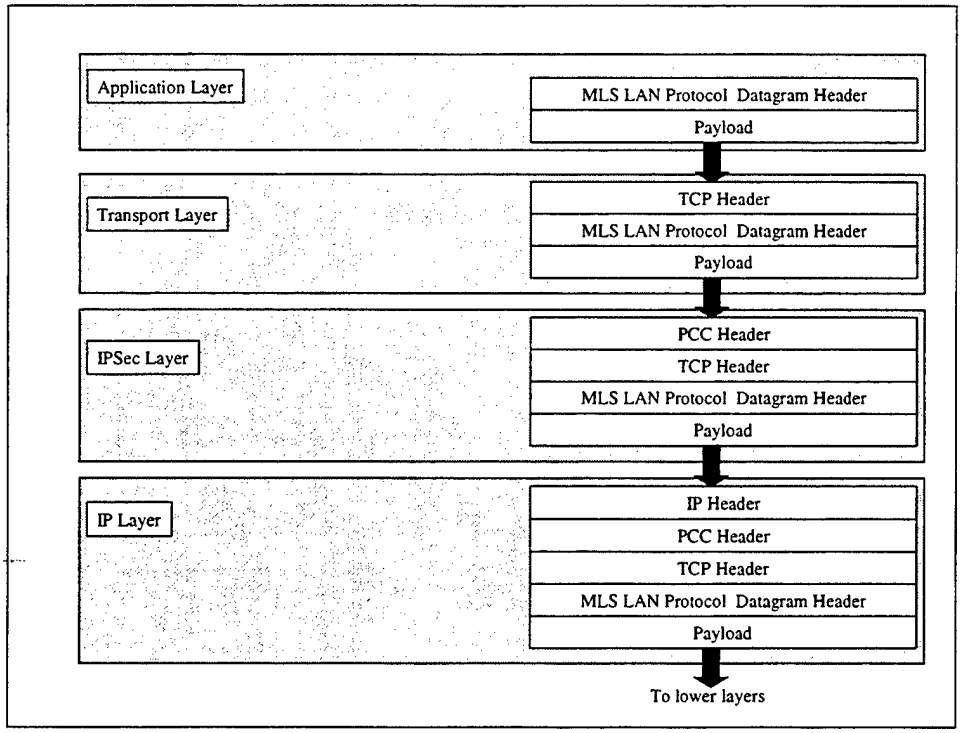

Figure 5. MLS LAN Protocol Datagram Packaging

a. TCBE Options and transitions in State [0] (Power Off):
The only option for a client to transition out of State [0] is to apply power to the system.

b. TCBE Options and transitions in State [1] (Idle):
The following listing describes the allowable inputs and the appropriate actions to be taken by the TCBE in this state.

- **Unprotected Mode:** The selection of Unprotected Mode will transition the TCBE to State [2] (Unprotected Operations). (This will be developed as part of Future work)
- **SAK:** The activation of the SAK will transition the TCBE to State [3] (Trusted Processing). A Secure Attention Request packet will be transmitted.
- **Power Off:** The removal of power to the workstation will transition the TCBE to State [0] (Power Off).

c. TCBE Options and transitions in State [2] (Unprotected Operations):

The following listing describes the allowable inputs and the appropriate actions to be taken by the TCBE in this state.

- Unprotected operations in the client workstation environment. (This will be developed as part of Future work)
- **SAK:** The activation of the SAK will transition the TCBE to State [3] (Trusted Processing). A Secure Attention Request packet will be transmitted.

- **Power Off:** The removal of power to the workstation will transition the TCBE to State [0] (Power Off).

d. TCBE Options and transitions in State [3] (Trusted Processing):

The following listing describes the allowable inputs and the appropriate actions to be taken by the TCBE in this state.

- **NOOP**: The receipt of a NOOP command will cause the TCBE to display the received payload to the user. The nature of the payload is used to provide the user with an interactive login and session negotiation with the TCB. The TCB Extension Server will use this command field value to pass information directly to the user without TCBE intervention, or interpretation. The TCBE will remain in State [3] (Trusted Processing).
- **Logout:** The receipt of a LOGOUT command will transition the TCBE to State [1] (Idle). Any received payload will be displayed to the user. This command directs the TCBE to purge the existing Operating System and files from the workstation's memory and return to an "Idle" state.
- **Run**: The receipt of a RUN command will cause the TCBE to purge the workstation's memory and transition to State [4] (Trusted Session) with a sanitized version of the Operating System. Any received payload will be displayed to the user. The TCB Extension Server will use this command field value to activate a session with the TCBE equipped client workstation.
- **Resume**: The receipt of a RESUME command will transition the TCBE back to State [4] (Trusted Session) at the current session level. Any received payload will be displayed to the user. This command directs the TCBE to maintain the original version of the Operating System and return to the user's previous session configuration. The TCB Extension Server will use this command field value to re-activate a session with the TCBE equipped client workstation.
- **New:** This command provides for a future capability in the MLS LAN. The "NEW" command is intended to allow the incorporation of an algorithm, which will determine if the client workstation's Operating System and memory need be purged. The algorithm will enable the TCB Extension Server to perform an evaluation of the user's current sensitivity level and the requested new sensitivity level. If the change in session level will cause a violation of the security policy through the use of the currently running operating system, the system will be purged through a RUN command. If the new session level does not violate the security policy, a NEW command could be used to change the session, but maintain the current operating system. This algorithm is left for future work.

- **Disconnect:** The receipt of a DISCONNECT command terminates the connection to the TCB Extension Server and returns the control of the client workstation to State [1] (Idle). Any received payload will be displayed to the user. This command directs the TCBE to terminate the client workstation's connection to the TCB.

- **SAK:** The activation of the SAK will transition the TCBE to State [3] (Trusted Processing). A Secure Attention Request packet will be transmitted.

- **Update PCC:** The receipt of a UPDATE PCC command will direct the TCBE to modify the TCBE Security Policy Database with the data contained in the payload. The TCBE will send a "PCC Updated" Response packet.

- **Escape**: Future work. The use of an escape key will transition the TCBE to the state from which it entered State [3] (Trusted Processing).

- **Power Off**: The removal of power to the workstation will transition the TCBE to State [0] (Power Off).

e. Options and transitions in State [4] (Trusted Session):
The following listing describes the allowable inputs and the appropriate actions to be taken by the TCBE in this state.

- Trusted operations in the client workstation environment.
- **SAK**: The activation of the SAK will transition the TCBE to State [3] (Trusted Processing). A Secure Attention Request packet will be transmitted.
- **Power Off**: The removal of power to the workstation will transition the TCBE to State [0] (Power Off).
- **Disconnect**: The receipt of a DISCONNECT command terminates the connection to the TCB Extension Server and returns the control of the client workstation to State [1] (Idle). Any received payload will be displayed to the user. This command directs the TCBE to terminate the client workstation's connection to the TCB.

3.5.2 TCB Extension Server State Options and Transitions

The TCB Extension Server has the capability of generating only three types of TCB-TCBE Protocol Datagrams. They are as follows:

- **No Response**. The TCB Extension Server will generate and transmit a No Response packet (as described in Section 3.4.2) for datagrams when the TCB Extension Server does not require a response. The TCB Extension Server will use this response type for commands that are directive in nature, such as "RUN" or "LOGOUT" or informational in nature, such as "NOOP (No Operation Expected)".

- **Response with Echo.** The TCB Extension Server will generate and transmit a Response with Echo packet (as described in Section 3.4.2) for datagrams when the TCB Extension Server requires a response and there is no protection compromise if the user's response is echoed to the screen. The TCB Extension Server will use this response type for commands that require user input that is not of a private nature, such as "USERNAME" or "SESSION LEVEL CHANGE".
- **Response without Echo.** The TCB Extension Server will generate and transmit a Response without Echo packet (as described in Section 3.4.2) for datagrams when the TCB Extension Server requires a response and there is a possible protection compromise if the user's response is echoed to the screen. This response type will be entered when a response is expected from the TCBE and the TCB Extension Server does NOT allow the TCBE to display the user's response on the screen.

a. TCB Extension Server Command Options and transitions in State [0] (Power Off):

The only option for the TCB Extension Server to transition out of State (0) is to apply power to the system.

b. TCB Extension Server Command Options and transitions in State [1] (Idle):

The following listing describes the allowable inputs and the appropriate actions to be taken by the TCB Extension Server in this state.

- **SAR**: The receipt of a Secure Attention Request packet will transition the TCB Extension Server to State [2] (Connected).
- **Power Off**: The removal of power to the TCB Extension Server will transition the TCB Extension Server to State [0] (Power Off).

c. TCB Extension Server Command Options and transitions in State [2] (Connected):

The following listing describes the allowable inputs and the appropriate actions to be taken by the TCB Extension Server in this state.

- **SAR**: The receipt of a Secure Attention Request packet will initiate the User I&A portion of this State. The TCB Extension Server will remain in State [2].
  - **User Identification and Authentication Processing**: The TCB Extension Server will transmit a series of NOOP commands to request the user provide their username and password for login. The "Username" prompt will be delivered using a Response with Echo packet, while the "Password" prompt will be delivered using a Response without Echo packet.
  - **Incorrect User Identification and Authentication:** An incorrect User I&A will transition the TCB Extension Server to State [1] (Idle). A No Response packet will be transmitted to the TCBE containing the Command DISCONNECT.

- **Correct User Identification and Authentication:** A correct User I&A will transition the TCB Extension Server to State [3] (Logged In).

- **Power Off**: The removal of power to the workstation will transition the TCB Extension Server to State [0] (Power Off).

d. TCB Extension Server Command Options and transitions in State [3] (Logged In):

The following listing describes the allowable inputs and the appropriate actions to be taken by the TCB Extension Server in this state. Once the menu has been displayed to the user by the TCBE, there is no specific order that must apply to the user requests. The transition to State [3] (Logged In) will cause the TCB Extension Server to send to the TCBE a User Interface Menu as a payload in a Response with Echo packet using the NOOP command. The TCBE will display the packet payload to the user. This menu that is displayed provides a listing of selections, which can be used to perform trusted path operations. The TCB Extension Server will remain in State [3] (Logged In). Selections offered to the user will be: Session, Session Level Change, Group Change, Logout, and Run.

- **Session**: Upon the receipt of a response packet containing a "SESSION" request, the TCB Extension Server will return the current session level information. The TCB Extension Server will remain in the current State. This information will be presented to the user via No Response packets using the NOOP command.
- **Set level** (SL): Upon the receipt of a response packet containing a "SESSION LEVEL CHANGE" request, the TCB Extension Server will enter an interactive exchange to determine the session level the user would like to use. The TCB Extension Server will remain in the current State. This information will be presented to the user via Response with Echo packets using the NOOP command.
- **Set Group** (SG): Upon the receipt of a response packet containing a "SET GROUP" request, the TCB Extension Server will enter an interactive exchange to determine the group that the user would like to use. The TCB Extension Server will remain in the current State. This information will be presented to the user via Response with Echo packets using the NOOP command.
- **Logout**: Upon the receipt of a response packet containing a "LOGOUT" request, the TCB Extension Server will send a LOGOUT command to the TCBE. The TCB Extension Server will transition to State [1] (Idle). This command will be transmitted to the TCBE using a No Response packet. The payload of the packet may be empty or it may contain a "Logout complete" message.
- **Update PCC**: Upon the receipt of a response packet containing a "RUN" request, the TCB Extension Server will update the TCBE Security Policy Database using the UPDATE PCC command. This command MUST be transmitted by the TCB Extension Server prior to issuing a "RUN" command. The information for the Security Policy

Database will be placed in the payload of a Response packet. The format of the information in the SPD payload will be developed as part of a future effort. Following successful update of the TCBE SPD, the TCBE will transmit a "PCC UPDATED" packet. If this packet is not received from the TCBE, the TCB Extension Server connection will "time out", disconnecting the TCBE-equipped workstation from the MLS LAN and transition to State [1] (Idle). Otherwise, the TCB Extension Server will remain in the current State.

The incorporation of a "sensitivity algorithm (as described in Section 3.5.1.d) may require the inclusion of an associated "Ready" State for the TCB Extension Server. This "Ready" state would allow the TCB Extension Server to decide whether the TCBE-equipped workstation's memory needs to be purged. The development of this new state is left to future work.

- **Run**: Upon the receipt of a "PCC UPDATED" packet, the TCB Extension Server will send a RUN command to the TCBE. The TCB Extension Server will transition to State [4] (Running). This command will be transmitted to the TCBE using a No Response packet.
- **SAR**: The receipt of a Secure Attention Request packet will cause the TCB Extension Server to send the User Interface Menu Command Options and the TCB Extension Server will remain in the current State. These options will be presented to the user via Response with Echo packets using the NOOP command.
- **Power Off**: The removal of power to the workstation will transition the TCBE to State [0] (Power Off).

e. TCB Extension Server Command Options and transitions in State [4] (Running):
   - The TCB Extension Server will remain active, waiting for a Secure Attention Request packet from the client workstation environment.
   - **SAR**: The receipt of a Secure Attention Request packet will transition the TCB Extension Server to State [5] (Trusted Session Processing).
   - **Power Off**: The removal of power to the workstation will transition the TCBE to State [0] (Power Off).

f. TCB Extension Server Command Options and transitions in State [5] (Trusted Session Processing):
   The transition to State [5] (Trusted Session Processing) will cause the TCB Extension Server to send to the TCBE an updated User Interface Menu as a payload in a Response with Echo packet using the NOOP command. The TCBE will display the packet payload to the user. This menu that is displayed is essentially the same as the provided in State [3] (Logged In), except the user is now provided the additional selection of "Resume". The TCB Extension Server will remain in State [5] (Trusted Session Processing).
   - **Session**: The receipt of a response packet containing a "SESSION" request will be handled as described in Section 3.5.2.d.

- **Set level** (SL): The receipt of a response packet containing a "SESSION LEVEL CHANGE" request will be handled as described in Section 3.5.2.d.
- **Set Group** (SG): The receipt of a response packet containing a "SET GROUP" request will be handled as described in Section 3.5.2.d.
- **Logout**: The receipt of a response packet containing a "LOGOUT" request will be handled as described in Section 3.5.2.d.
- **Update PCC**: This command will function as described in Section 3.5.2.d.
- **Run**: This command will function as described in Section 3.5.2.d.
- **Resume**: Upon the receipt of a response packet containing a "RESUME" request, the TCB Extension Server will send a RESUME command to the TCBE. The TCB Extension Server will transition to State [4] Running. This command will be transmitted to the TCBE using a No Response packet. The payload of the packet may be empty or it may contain a "Resume completed" message. This command should NOT be available following the user's change of session level or group.
- **SAR**: The receipt of a Secure Attention Request packet will be handled as described in Section 3.5.2.d.
- **Power Off**: The removal of power to the workstation will transition the TCBE to State [0] Power Off.

4. Session Status Protocol

4.1 Introduction

Following the successful session negotiation by a user into the MLS LAN, the TCB Extension Server must create a session database entry through the Session Database Server (SDS) that uniquely defines information such as who the user is, from which TCBE-equipped workstation he logged in, and the sensitivity and integrity levels assigned to the current session. The integrity of the Session Status Database (SSD) is critical to the assurance of the overall LAN and therefore the ability to manipulate (read/write) its data must be constrained. The Session Status Protocol is provided as a method for the TCB Extension Server, acting as the only TCB entity with both read and write access to the SDS, to modify the contents of the SSD. This protocol is also used by other TCB entities to verify the session status of MLS LAN users. TCB entities, other than the TCB Extension Server are limited to "read only" access. Protection against replay and spoofing is provided by the underlying Protected Communications Channel. The protocol will be described in terms of the TCB Extension Server state transitions, SDS state transitions, and "other TCB Entities" requests.

4.2 TCB Extension Server States.

The states from which the TCB Extension Server will use the Session Status Protocol are described in Section 3.3. While other entities can use this protocol to query the SDS, the state transitions are more germane to the creation, modification and deletion of database entries. This protocol does not constitute a transition for the TCB Extension Server.

4.3 Session Database Server States.

The Session Database Server uses input commands received from the TCB Extension Server to modify the status of the Session Status Database, however, the configuration of the Session Database Server is not relevant to this protocol.

4.4 Session Status Protocol Datagram Format

There are two defined datagram formats for the Session Status protocol. The first, shown in Figure 6, is the "Request" packet used to convey information from the TCB Extension Server or other TCB entities to the Session Database Server (SDS). The second, shown in Figure 7, is the "Response" packet provided to enable the Session Database Server to respond to the TCB entities' request.

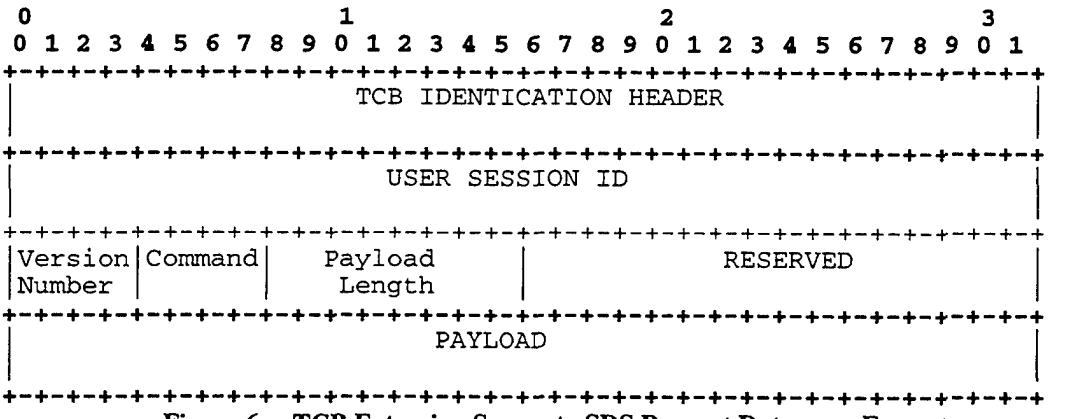

Figure 6. TCB Extension Server to SDS Request Datagram Format

4.4.1 TCB Entity to SDS Datagram Field Descriptions

The following subsections define the fields that comprise the TCB entity-to-SDS "Request" datagram as depicted in Figure 7. The TCB Identifier, Version Number, Payload Length and Reserved fields are the same as described in Section 3.4.1 and will not be repeated here. All fields in the datagram, however are mandatory, i.e., they are always present in the Session Status Protocol.

a. User Session Identification.
The User Session Identification is a 32-bit field that uniquely identifies the TCBE equipped workstation that has established a session on the MLS LAN. This field is used to identify the specific record in the SSD. Version 1 uses the TCBE ID as the User Session ID, however, future versions may incorporate a different ID value.

b. Command.
This field is a 4-bit value that identifies the type of command that is being passed to the SDS. The value of this field is chosen from the listing below. 16 command types possible, however, in the current version only 4 are defined.
- *Value 0 -- Create*
- *Value 1 – Modify*
- *Value 2 – List*
- *Value 3 – Delete*

c. Payload:
This is a variable length field that contains the user and session information to be added by the SDS. The payload will be organized into (attribute name / input data) pairs such as: "Current Session Level: Secret". Available attributes are as follows:
- *USER ID*: The user's unique "Username".
- *CURRENT SESSION LEVEL:* The current negotiated session sensitivity level.
- *CURRENT INTEGRITY LEVEL:* The current negotiated session integrity level.

- *CURRENT GROUP SETTING:* The current negotiated group or role.
- *RUNNING:* A flag that denotes whether or not the current session is started.

4.4.2 Session Database Server to TCB Entity Datagram Field Descriptions

The following subsections define the fields that comprise the SDS-to-TCB Entity "Response" datagram as depicted in Figure 8. The TCB Identifier, Version Number, Payload Length and Reserved fields are the same as described in Section 3.4.1 and will not be repeated here. All fields in the datagram, however are mandatory, i.e., they are always present in the Session Status Protocol.

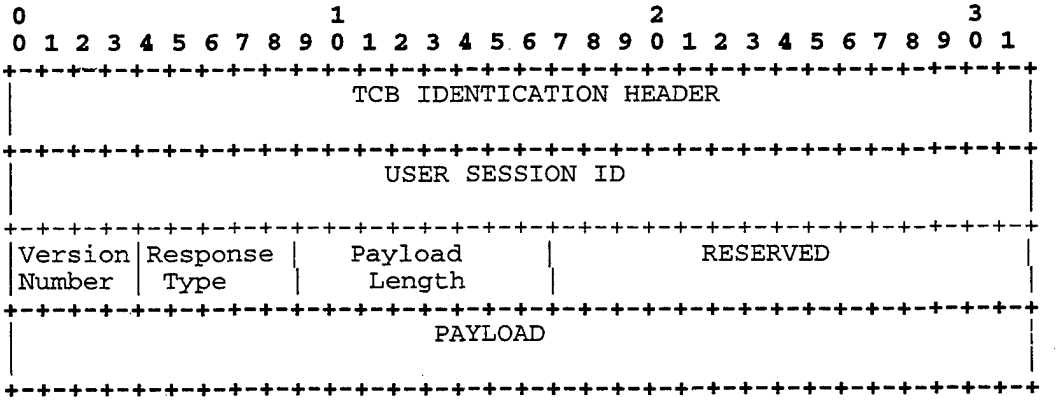

Figure 7. SDS to TCB Extension Server Response Datagram Format

a. User Session Identification.
This field has the same attributes as described in Section 4.4.1.b.

b. Response.
This is a field is a 4-bit value that identifies the type of response that is being passed to the TCB Entity. The value of this field is chosen from a listing of response types defined in the latest version of the Session Status Protocol. 16 response types are possible, however, in the current version only 3 are defined. For Version 1, the response type values are as follows:
- *Value 0 -- ACK Response*
- *Value 1 – NAK Response*
- *Value 2 – Payload Response*

c. Payload:
- This is a variable length field that contains the user and session information to be passed to the TCB Entity. The payload will be organized the same as in Section 4.4.1.g with the addition of the following payload type.
- *ERROR:* This will be an informative message describing the reason for failure.

4.4.3 Datagram Packaging.

The TCB Entity and SDS will generate one of the request types of Session Status datagram packets described in Sections 4.4.1 or 4.4.2 for their secure operation communications. The datagram will be created by either the TCB Entity or the SDS and passed to the lower layers protocols for transmission to the other entity. Since the protocol is created using fixed fields, the value in these datagram fields need no manipulation and can be parsed for use. The packaging used for transmission of the Session Status Protocol is the same as depicted in Figure 5.

4.5 SDS to TCB Extension Server Interaction.

This section describes the uses of the Session Status protocol between the SDS and the TCB Extension Server. The use of the "List" Request, however, could be used similarly from any TCB Entity. Prior to use of the protocol both the TCB Entity and Session Status Database must be powered and in at least State [1] Idle.

The loss of communications between the TCB Extension Server and the SDS could allow unwarranted access to the MLS LAN. To prevent an insecurity, the MLS LAN requires some control mechanism that could prevent new connections to the MLS LAN and its services in this event. The development of this mechanism is left to future work.

4.5.1 TCB Entity State Options

All TCB entities, such as the Secure Session Server, have the capability to generate only one type of Session Status Protocol datagram. It is a Request datagram as described in Section 4.4.1, however, the only available type of request is the "List" command. The transmission of this datagram does not constitute a State transition for any TCB entity.

The function of the LIST command datagram of the Session Status Protocol for TCB entities other than the TCB Extension Server is as follows:

- **List:** Upon Receipt of a request for Network Application Services, a TCB Entity will generate and transmit a LIST Request packet placing the requestor's TCBE ID in the User Session Identification field. This command directs the SDS to locate and return the attribute values contained in the entry found under the listing of the User Session Identification number. The response will determine whether the user is currently logged in. If the user is logged in, the TCB entity will continue with the connection process as described in Section 5.3. If, however, a NAK response is received from the SDS the TCB entity will terminate the Application Protocol connection to the requesting TCBE-equipped workstation.

The TCB Extension Server has the capability of generating only one type of Session Status Protocol datagram. It is also a Request datagram as described in Section 4.4.1.The TCB Extension Server will generate and transmit a Request

packet to request the creation or modification of records. This action can only be taken from three states: State [2] (Connected), State [3] (Logged In), and State [5] (Trusted Session Processing). The transmission of this datagram does not constitute a transition for the TCB Extension Server.

a. TCB Extension Server Options in State [2] (Connected).

- **List:** Upon the receipt of a SAR packet containing the TCBE ID the TCB Extension Server will issue a "LIST" command to see if the SDS has created a previous entry for the current user. This command directs the SDS to locate and return the attribute values contained in the entry found under the listing of the User Session Identification number. The response will determine whether the user is currently logged in. If the user is logged in, the TCB Extension Server will transition to State [3] (Logged in). If a NAK response is received from the SDS, the TCB Extension Server will continue with the "User I&A session as described in Section 3.5.2.c. and remain in the current State.

- **Create:** Once the TCB Extension Server has verified the User I&A (as described in Section 3.5.2.c, it will issue a "CREATE" command to instantiate a record for the new user. This command must be completed prior to the TCB Extension Server completing the successful User I&A which enables the transition to State [3] (Logged in). This command tells the SDS to create a new entry in the database. The TCB Extension Server will use this payload field value to pass the user and session information to the SDS.

b. TCB Extension Server Command Options in State [3] (Logged In):

- **List:** Upon the receipt of a response packet containing a "SESSION" request, the TCB Extension Server will issue a "LIST" command to retrieve the attribute values contained in the session database entry found under the listing of the User Session Identification number. The TCB Extension Server will pass this information to the TCBE as described in Section 3.5.2.d (Session). The TCB Extension Server will remain in State [3] (Logged in).

- **Create:** This command cannot be issued from this State.

- **Modify:** Upon the receipt of a response packet containing a "RUN" request, from the TCBE, the TCB Extension Server will issue a "MODIFY" command requesting the SDS to update the current session information to the values negotiated during the Trusted Path Processing. This use of this protocol does not constitute a transition. The SDS will use this command field to change the value of one or more of the attributes of a current database entry.

- **Delete:** Upon the receipt of a response packet containing a "LOGOUT" request, or the issuance of a "DISCONNECT", the TCB Extension Server will issue a "DELETE" command to request the SDS remove the User's current session record. The use of this protocol does not

constitute a transition. This command directs the SDS to delete a current entry in the database.

c. TCB Extension Server Command Options in State [5] (Trusted Session Processing):

- **List**: The receipt of a response packet containing a "SESSION" request will be handled as described in Section 4.5.1.b.
- **Create**: This command cannot be issued from this State.
- **Modify**: The receipt of a response packet containing a "RUN" request will be handled as described in Section 4.5.1.b.
- **Delete**: The receipt of a response packet containing a "LOGOUT" request will be handled as described in Section 4.5.1.b.

4.5.2 Session Database Server Options

The Session Database Server has the capability of generating only two types of Session Status Protocol datagrams. They are as follows:

- **ACK Response**. The Session Database Server will generate and transmit an ACK Response packet for Request datagrams when the TCB Entity requires only a response determining success. The SDS will use this response type for commands that are directive in nature, such as "CREATE", "MODIFY" and "DELETE". The payload for an ACK RESPONSE packet will contain success verification information for the TCB Extension Server.
- **NAK Response**. The Session Database Server will generate and transmit a NAK Response packet for Request datagrams when the TCB Entity requires determination of failure. The SDS will use this response type for commands such as "CREATE", "LIST", "MODIFY" and "DELETE". The payload for a NAK RESPONSE packet may contain information for the TCB Entity concerning the reason for the failure.
- **Payload Response**. The Session Database Server will generate and transmit a Payload Response packet for Request datagrams when the TCB Entity requires the information contained in the record. This response type will be entered when the SDS has been issued a command that requires the return of information contained in a database entry.

4.5.3 Session Database Server Response

The Session Database Server will respond to a TCB Entities' commands in the following manner:

- **Create:** The receipt of a "CREATE" Request packet will direct the SDS to create a record under the index coinciding with the User Session Identification Field received in the request packet. The SDS will generate

and transmit the outcome of the operation to the TCB Extension Server using an ACK or NAK Response packet.

- **Modify**: The receipt of a "MODIFY" Request will direct the SDS to update the attributes associated with the User Session Identification to the values contained in the payload. The SDS will generate and transmit the outcome of the operation to the TCB Extension Server using an ACK or NAK Response packet.
- **Delete:** The receipt of a "DELETE" Request will direct the SDS to remove the record associated with the User Session Identification contained in the User Session ID datagram field. The SDS will generate and transmit the outcome of the operation to the TCB Extension Server using an ACK or NAK Response packet.
- **List:** The receipt of a "LIST" Command will direct the SDS to transmit the values of the attributes associated with the User Session Identification contained in the User Session ID datagram field. The SDS will generate and transmit the information to the requesting TCB entity using a Payload Response packet. If the result of the search is a failure, the SDS will generate and transmit a NAK Response packet.

THIS PAGE INTENTIONALLY LEFT BLANK

5. TCBE-to-Session Server Connection Protocol

5.1 Introduction

The MLS LAN is intended to provide access to multiple Application Layer Protocols such as FTP, HTTP, or IMAP. For Version 1, these application services are only accessible to users who have successfully logged in to the MLS LAN and established a Session within the TCB. The TCBE-to-Session Server Connection Protocol is provided as a method for the TCBE to pass a unique identifier to the Secure Session Server (SSS) in order to check with the Session Database Server (SDS) for the user's session information. The MLS LAN uses the TCBE identification number as this identifier. The design of this protocol, however, will allow alternate future data, such as a unique session token, to be inserted adding flexibility to the MLS LAN. Once the user's information is returned from the SDS, the Session Server will establish the proper session level connectivity to the appropriate MLS LAN Application Protocol Server (APS) as described in [Shif00]. If, however, the user is not found by the SDS, the connection to the Application Protocol Server will be terminated.

In the future, the MLS LAN should allow users to access these services through an anonymous or "untrusted" connection, but this will not affect the applicability of this protocol.

5.2 TCBE States.

The states from which the TCBE will use the Session Status Protocol are described in Section 3.2.

5.3 Secure Session Server States.

A Secure Session Server is created for each higher layer application protocol supported by the MLS LAN. Its responsibility is to accept and validate requests for access to the particular protocol. Following the acceptance of a request for service from a TCBE, the SSS will use the TCBE ID to verify that a session has been established. Connections from TCBEs with valid sessions will be passed to a "child" session server, while connections from TCBEs without sessions will be terminated. The SSS uses the TCP/IP Application Protocol connection request packet from the TCBE equipped client workstation to change its configuration. The configuration of the Secure Session Server is not relevant to the use of this protocol.

5.4 TCBE-to-Session Server Protocol Datagram Format

The datagram format for the protocol is shown in Figure 8.

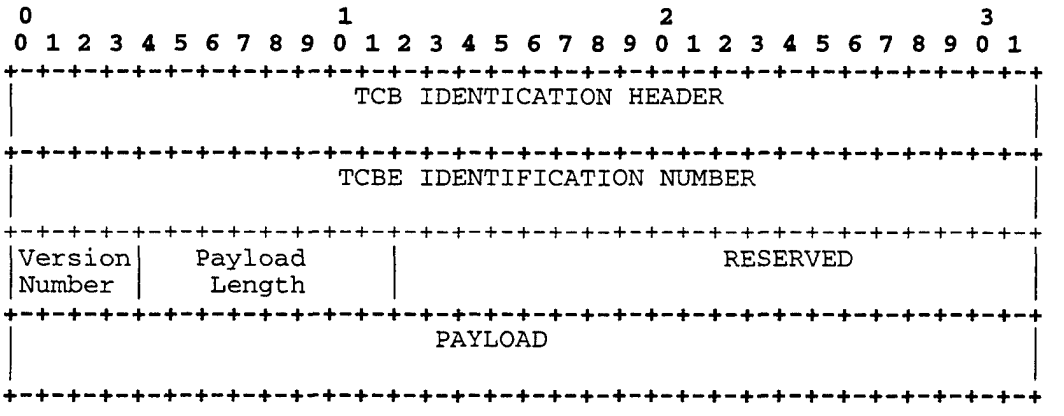

Figure 8. TCBE-to-Session Server Datagram Format

5.4.1 TCB Extension to SSS Datagram Field Descriptions

The following subsections define the fields that comprise the TCBE to SSS "Identification" datagram. The TCB Identifier, Version Number, Payload Length and Reserved fields are the same as described in Section 3.4.1 and will not be repeated here. All fields in the datagram, however are mandatory, i.e., they are always present in the TCBE-to-Session Server Connection Protocol. .

a. TCBE Identification Number Field.
   This is a 32-bit value that identifies the TCB Entity that created the packet. This will be used by the Secure Session Server to facilitate Hardware Identification and Authentication.

b. Payload Field.
   This is a variable length field that contains the information to be sent to the SSS from the TCBE. This field is empty in Version 1 of the protocol.

5.4.2 Application Protocol Service Request Packet

Each application protocol client residing on the client workstation can generate an Application Protocol Service Connection Request packet. This is a generic TCP/IP Client-Server packet. The TCBE forwards this request to the SSS, which hosts the particular protocol without modification.

5.4.3 TCBE-to-Session Server Datagram Packaging.

The TCBE will generate a TCBE-to-Session Server Identification datagram for all APS requests. The Identification datagram will be created by TCBE Extension and passed to the lower layers protocols for transmission to the other entity. Since the protocol is created using fixed fields, the value in these datagram fields needs no manipulation and can be parsed for use. The packaging used for transmission of the TCBE-to-Session Server Protocol is the same as depicted in Figure 5.

5.5 TCBE to Secure Session Server Interaction.

This section describes the uses of the protocol. Prior to use of the protocol both the TCBE and Session Server must be powered and in at least State [1] (Idle).

5.5.1 TCBE State Options

The TCBE has the capability of generating only one type of TCBE-to-Session Server Connection datagram. It is designed as follows:

- **Identification**. The TCBE will generate and transmit an Identification packet following each application protocol request received from the client workstation. In the current version of this protocol, this action can only be taken from one state: State [4] (Trusted Session), however future versions may allow for this protocol in State [2] (Unprotected Operations). The use of this protocol does not constitute state transitions for the TCBE.

a. TCBE Options in State [2] (Unprotected Operations):

- This protocol is not available in this State, however, it may be used in future upgrades of the MLS LAN Unprotected Operations.

b. TCBE Command Options in State (4) (Trusted Operations):

The following listing describes the allowable inputs and the appropriate actions to be taken by the TCBE in this state.

- **Identification Packet** Upon the receipt of a "Application Protocol Service Connection Request" from a higher layer protocol client residing on the client workstation, the TCBE will generate and transmit an Identification packet to the Secure Session Server which hosts that protocol.

5.5.2 Secure Session Server Options

The Secure Session Server does not respond directly to the TCBE using this protocol. The Secure Session Server uses the information contained in the TCBE-to-Session Server datagram to generate and transmit a "LIST" command to the Session Database Server as described in Section 4.5.1. This command will verify the user's current session information. Once this information has been verified, the Secure Session Server will continue with the Application Protocol Server operations as described in [Shif00]. If the user is not logged in, the Secure Session Server will simply terminate the connection to the requesting application. If the Identification datagram is not received, the "LIST" command cannot be transmitted and the Secure Session Server cannot connect the Application Protocol client request to the Application Protocol Server. This action will, in turn cause a time out in the Application Layer requiring a retry.

THIS PAGE INTENTIONALLY LEFT BLANK

REFERENCES

[WIL00a], Wilson, J., "MLS LAN System Requirements Document", Naval Postgraduate School, Monterey CA. April 2000.

[WIL00b], Wilson, J., "MLS LAN Protocol High Level Analysis Document", Naval Postgraduate School, Monterey CA. April 2000.

[Shif00], Shifflett, D., "MLS LAN Draft Design Document", Naval Postgraduate School, Monterey CA. April 2000.

[BRA97], Bradner, S., "Key words for use in RFCs to Indicate Requirement Level", BCP 14, RFC 2119, March 1997.

[TDG97], Thayer, R., and Doraswamy, N., IP Security Document Roadmap", RFC 2411, November, 1998

[KA98a], Kent, S., and Atkinson, R., "Security Architecture for the Internet Protocol", RFC 2401, November 1998.

[KA98b], Kent, S., and Atkinson, R., "IP Authentication Header (AH)", RFC 2402, November 1998.

[KA98c], Kent, S., and Atkinson, R., "IP Encapsulating Security Payload (ESP)", RFC 2406, November 1998.

[HC98], Harkins, D., and Carrel, D., "The Internet Key Exchange (IKE)", RFC 2409, November, 1998.

[MSST97], Maughan, D., Schertler, M., Schneider, M., and Turner, J., "Internet Security Association and Key Management Protocol" (ISAKMP), RFC 2408, November, 1997.

[Pip98], Piper, D.,"The Internet IP Security Domain of Interpretation for ISAKMP", RFC 2407, November, 1998.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

1. *Department of Defense Information Security Program*, DoD Directive 5200.1, Assistant Secretary of Defense Command, Control Communications, and Intelligence (C3I), January 1997

2. *Glossary of Computer Security Terms,* National Computer Security Center, NCSC-TG-004, Version-1, 21 October 1988.

3. *Department of Defense Trusted Computer System Evaluation Criteria*, DoD 5200.28-STD, National Computer Security Center, December 1985.

4. *Common Criteria for Information Technology Security Evaluation Version 2.1*, Common Criteria Project Sponsoring Organisations, August, 1999

5. Sterne, Daniel F., *On the Buzzword "Security Policy"*, Proceedings of the 1991 IEEE Symposium on Research in Security and Privacy, pg 219-230, May 1991, Oakland Ca.

6. White, Gregory B., Fisch, Eric A., Pooch, Udo W., *Computer System and Network Security*, CRC Press, Boston MA, 1996.

7. *Trusted Network Interpretation of The Trusted Computer System Evaluation Criteria*, National Computer Security Center, NCSC-TG-005, Version 1, 31 July 1987.

8. Bell, D.E. and LaPadula, L.J. - *Secure Computer Systems: Unified Exposition and Multics Interoperation*, MTR-2997 Rev. 1, MITRE Corp., Bedford Mass., March 1976.

9. Pfleeger, Charles, *Security in Computing*, Prentice Hall PTR, Upper Saddle River, NJ, 1997.

10. Biba, K, *Integrity Considerations for Secure Computer Systems*; ESD-TR-76-372, ESD/AFSC, Hanscom AFB, Bedford, MA, Apr. 1977 [NTIS AD A039324]

11. Wang Government Services, www.wang.com/gov_services/security/sso/gov_services_ssso_xts_page2_c.asp

12. *XTS-300, STOP 4.4.2, Trusted Facility Manual*, Document ID: FS96-371-07, Wang Government Services Inc.

13. Shifflett, David, *Multi-level Secure Local Area Network Project Design Document Draft.* NPS, May 2000.

14. Fellows, Jon, Hemenway, Judy, Kelem, Nancy, Romero, Sandra, *The Architecture of a Distributed Trusted Computing Base*, Proceedings for the 10th Annual Computer Security Conference, 1987

15. Stallings, William, *Cryptography and Network Security*, Prentice Hall, Upper Saddle River, New Jersey,1999.

16. Dierks, T, Allen, C, *RFC 2246, The TLS Protocol Version 1.0*, Network Working Group, IETF, Jan 1999.

17. Diffie, W, Hellman, M, *New Directions in Cryptography*, IEEE Transactions on Information Theory, 1976.

18. Turner, Sean, Arsenault, Alfred, *Internet X.509 Public Key Infrastructure PKIX Roadmap*", IETF PKIX Working Group Mar, 2000 [draft-ietf-pkix-roadmap-05.txt]

19. Rivest, R, Shamir, A, Adelman, L, *A Method for Obtaining Digital Signatures and Public Key Cryptosystems*, Communications of the ACM, Feb 1978.

20. Doraswmy, Naganand, Harkins, Dan, *IPSec, The New Security Standard for the Internet, Intranets, and Virtual Private Networks*, Prentice Hall, Upper Saddle River, New Jersey, 1999.

21. Maughan, D, Schertler, M, Schneider, M, Turner, J., *RFC 2408, The Internet Security Association and Key Management Protocol (ISAKMP)*, Network Working Group, IETF, Nov. 1998.

22. Kent, S, Atkinson, R, *RFC 2401, Security Architecture for the Internet Protocol*, Network Working Group, IETF, Nov. 1998.

23. Michael Kabay, *Enterprise Security, Protecting Information Assets*, McGraw Hill, New York, NY, 1996.

24. Harkins, D, Carrel, D, *RFC 2409, The Internet Key Exchange (IKE)*, Network Working Group, IETF, Nov. 1998

25. Kent, S, Atkinson, R, *RFC 2402, IP Authentication Header*, Network Working Group, IETF, Nov. 1998.

26. Kent, S, Atkinson, R, *RFC 2406, IP Encapsulating Security Payload (ESP)*, Network Working Group, IETF, Nov. 1998.

27. Piper, D, *RFC 2407, The Internet IP Security Domain of Interpretation for ISAKMP*, Network Working Group, IETF, Nov. 1998.

28. BryerJoyner, S, Heller S., *Secure Local Area Network Services for a High Assurance Multilevel Network*, Naval Postgraduate School, March 1999.

29. Downey, J., Robb, D., *Design of a High Assurance, Multilevel Secure Mail Server (HAMMS)*, Naval Postgraduate School, September, 1997.

30. Turan, Bora, *Analysis for a Trusted Computing Base Extension Prototype Board*, Naval Postgraduate School, March 2000.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

No. Copies

1. Defense Technical Information Center 2
   8725 John J. Kingman Rd., Ste 0944
   Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library 2
   Naval Postgraduate School
   411 Dyer Rd.
   Monterey, CA 93943-5101

3. Chairman, Code CS 1
   Computer Science Department
   Naval Postgraduate School
   Monterey, CA 93943-5000

4. Dr. Cynthia E. Irvine 3
   Computer Science Department Code CS/Ic
   Naval Postgraduate School
   Monterey, CA 93943-5000

5. Mr. James P. Anderson 1
   James P. Anderson Company
   Box 42
   Fort Washington, PA 19034

6. Director, Training and Education 1
   MCCDC, Code C46
   1019 Elliot Rd.
   Quantico, Virginia, 22134-5027

7. Director, Marine Corps Research Center 2
   MCCDC, Code C40RC
   2040 Broadway St.
   Quantico, Virginia, 22134-5027

8. Director, Studies and Analysis Division 1
   MCCDC, Code C45
   300 Russell Road
   Quantico, Virginia, 22134-5030

9. Marine Corps Tactical Systems Support Activity 1
   Technical Advisory Branch
   Attn: Librarian
   Box 555171
   Camp Pendleton, CA 92055-5080

10. Commandant of the Marine Corps 1
Headquarters Marine Corps Code CP
2 Navy Annex
Washington, D.C. 20380-1775

11. Commandant of the Marine Corps 1
Headquarters Marine Corps Code CS
2 Navy Annex
Washington, D.C. 20380-1775

12. Marine Corps Representative 1
Naval Postgraduate School
Code 037, Bldg 330, Ingersoll Hall, Room 116
555 Dyer Road
Monterey, CA. 93943

13. LtCol. Jeffery D. Wilson 2
533 Galveston Rd
Fredericksburg, VA. 22405

14. Mr. Paul Pitelli 1
National Security Agency
Research and Development Building
R2, Technical Director
9800 Savage Road
Fort Meade, MD 20755-6000

15. Dr. Lee Taylor 1
National Security Agency
Research and Development Building
R22, Chief
9800 Savage Road
Fort Meade, MD 20755-6000

16. Mr. Howard Holm 1
National Security Agency
Research and Development Building
R23, Chief
9800 Savage Road
Fort Meade, MD 20755-6000

17. Carl Siel 1
Space and Naval Warfare Systems Command
PMW 161
Building OT-1, Room 1024
4301 Pacific Highway
San Diego, CA 92110-3127

18. Commander, Naval Security Group Command 1
Naval Security Group Headquarters
9800 Savage Road
Suite 6585
Fort Meade, MD 20755-6585

19. Ms. Deborah M. Cooper 1
Deborah M. Cooper Company
P. O. Box 17753
Arlington, VA 22216

20. Ms. Louise Davidson 1
N643
Presidential Tower 1
2511 South Jefferson Davis Highway
Arlington VA 22202

21. Mr. William Dawson 1
Community CIO Office
Washington DC 20505

22. Capt James Newman 1
N64
Presidential Tower 1
2511 South Jefferson Davis Highway
Arlington VA 22202

23. Mr. James Knoke 1
Wang Government Services Inc.
7900 Westport Dr.
Mclean, VA 22102-4299

24. Mr. Michael Focke 1
Wang Government Services Inc.
7900 Westport Dr.
Mclean, VA 22102-4299