(2)

CSC-EPL- 90/001

NATIONAL COMPUTER SECURITY CENTER

FINAL EVALUATION REPORT

Verdix Corporation

VSLAN 5.0

25 July 1990

Approved for Public Release:
Distribution Unlimited

FINAL EVALUATION REPORT

VERDIX CORPORATION

VSLAN 5.0

NATIONAL COMPUTER SECURITY CENTER

9800 Savage Road
Fort George G. Meade
Maryland 20755-6000

July 25, 1990

CSC-EPL-90/001
Library No. S235,898

VSLAN is a registered trademark of the VERDIX Corporation.
PC AT and IBM are registered trademarks of International Business Machines Corporation.
Ethernet is a registered trademark of the Xerox Corporation.
Multibus is a trademark of the Intel Corporation.
COMPAQ is a registered trademark of Compaq Computer Corporation.
Keyceptacle is a registered trademark of the Datakey Corporation.
LOCUS is a registered trademark of PC Interface.
MS-DOS and LINK are regist^r^red trademarks of Microsoft.
The Universal Development Interface is a registered trademark of the Real Time Computer Science Corporation.
UNIX is a registered trademark of UNIX System Laboratories Incorporated.
Macintosh and A/UX are trademarks of Apple Computer, Inc.

FOREWORD

This publication, the Final Evaluation Report, Verdix Corporation, Secure Local Area Network, is being issued by the National Computer Security Center under the authority of and in accordance with DoD Directive 5215.1, "Computer Security Evaluation Center". The purpose of this report is to document the results of the formal evaluation of Verdix's Secure Local Area Network component. The requirements stated in this report are taken from the Trusted Network Interpretation of the Trusted Computer System Evaluation Criteria, 31 July 1987, NCSC-TG-005, Version 1, National Computer Security Center.

Approved:

PATRICK R. GALLAGHER, JR.
Director
National Computer Security Center

July 25, 1990

This page intentionally left blank.

ACKNOWLEDGEMENTS

Team Members

Team members included the following individuals, who were provided by the following organizations:

Thomas A. Ambrosi
Ronald J. Bottomly
Paul A. Olson
Shawn M. Rovansek

NSA, Trusted Product and Network Security Evaluations Division
Fort Meade, Maryland

Frank Belvin
Manilal Daya
Dale M. Johnson

The MITRE Corporation
Bedford, Massachusetts

Jeremy E. Dawson

Department of Defence Commonwealth of Australia
Melbourne, Australia

Further Acknowledgements

Technical support was also provided by James Arnold, Alfred Arsenault, Stephen Carlton, David Chizmadia, and John Taylor.

This page intentionally left blank.

CONTENTS

Foreword iii
Acknowledgements v
Executive Summary xi

Section 1 Introduction 1
Evaluation Process Overview 1
Document Organization 2

Section 2 Product Overview 3
Security Policy 5
Communications 6
Operating Environment 8
System Integration 10

Section 3 Hardware Architecture 11
Introduction 11
The NSC Workstation 11
The NSD Board 12

Section 4 Software Architecture 19
Prntection features of the Intel 80286 20
NSD and NSD-Prime Software 21
Separation Kernel Service Groups 21
NSD and NSD-Prime Data Structures 24
NSD Application Tasks 26
NSD-Prime Task Organization 30
Network Security Center Software 31
Verdix Operating System 31
NSC Data Structures 32
NSC Application Processes 34

Section 5 VSLAN Subjects and Objects 37

Section 6 Software Protection Mechanisms 39
Secure Initialization and Shutdown 39
NSC Login and Logout Procedure 39
Principal Identification and 40
Authentication 40
Mandatory Access Control 41
Discretionary Access Control 42
Audit 43
Network Service Protection Mechanisms 44

Section 7 Assurances 47
Formal Security Policy Model 47
Covert Channel Analysis 50
System Integrity 53
Configuration Management 54

Section 8 Security Testing 57
Vendor Testing 57
Team Testing 58
Test Configuration 58
Team Test Results 59
Penetration Testing 60

Section 9 Evaluation as a B2 MDIA Network Component . 61
Discretionary Access Control 61
Object Reuse 63
Labels 64
Label Integrity 65
Exportation of Labeled Information 66
Exportation to Multilevel Devices 67
Exportation to Single-Level Devices 68
Labeling Human-Readable Output 69
Subject Sensitivity Labels 69
Device Labels 70
Mandatory Access Control 71
Identification and Authentication 72
Trusted Path 74
Audit 74
System Architecture 77
System Integrity 79
Covert Channel Analysis 80
Trusted Facility Management 81
Security Testing 81
Design Specification and Verification 83
Configuration Management 84
Security Features User's Guide 85
Trusted Facility Manual 86
Test Documentation 88
Design Documentation 88

Section 10 Evaluation of Part II Requirements 91
Communications Integrity 92
Authentication 93
Communications Field Integrity 93
Non-Repudiation 94
Denial of Service 94
Continuity of Operations 95
Protocol-Based Protection Mechanisms 96

Network Management 96
Compromise Protection 97
Data Confidentiality 97
Traffic Flow Confidentiality 97
Selective Routines 98

Section 11 Evaluators' Comments 99

Appendix A Evaluated Hardware Components . . . . A

Appendix B Evaluated Software Components . . . . B

Appendix C Intel 80286 Hardware Overview C
Architectural Overview C-1
List of Terms C-2
Security Features C-2

Appendix D Acronym Glossary . . . . D

Appendix E References . . . . E

EXECUTIVE SUMMARY

The security protection provided by the Verdix Corporation Secure Local Area Network has been examined by the Trusted Product and Network Security Evaluations Division of NSA. The security features of the Verdix Secure Local Area Network (VSLAN) were examined against the requirements specified by the Trusted Network Interpretation (TNI) of the Trusted Computer System Evaluation Criteria [23].

The VSLAN satisfies all the specified requirements of the TNI for a B2 MDIA network component when using the specified hardware (see Appendix A, "Evaluated Hardware Components"), and software (see Appendix B, "Evaluated Software Components"), configured in the most secure manner as described in the Trusted Facility Manual [16]. A B2 MDIA network component fulfills all the requirements as stated in Appendix A of the TNI for a B2 Mandatory Access Control component, a C2 Discretionary Access Control componen:: a C2 Identification & Authentication component, and a C2 Audit component. The VSLAN is not intended to be a complete, network system in and of itself, but can be used to build and support a complete, B2 network system when included in the proper network system architecture.

A B division network component provides a Network Trusted Computing Base (NTCB) that preserves the integrity of sensitivity labels and uses them to enforce a set of mandatory access control rules. The system developer has provided a security policy model on which the NTCB is based, and has furnished a specification of the NTCB and evidence that the reference monitor concept has been implemented.

In addition to the network component rating, the evaluation team has determined that the VSLAN satisfies the requirements for some of the security services described in Part II of the TNI. The services which are offered by the VSLAN are Authentication, Communications Field Integrity, Continuity of Operations, Protocol-Based Protection Mechanisms, Network Management, and Data Confidentiality.

This page intentionally left blank.

This page intentionally left blank.

INTRODUCTION

In October 1988, the National Computer Security Center (NCSC) began a formal product evaluation of the Verdix Secure Local Area Network (VSLAN), a Verdix Corporation product. The objective of this evaluation was to rate the VSLAN against the Trusted Network Interpretation (TNI) of the Trusted Computer System Evaluation Criteria [23], and to place it on the Evaluated Products List (EPL). This report documents the results of that evaluation. This evaluation applies to VSLAN 5.0 available from the Verdix Corporation.

Material for this report was gathered by the NCSC VSLAN evaluation team from VSLAN documentation, interaction with system developers, examination of source code, and experience using the VSLAN system.

Evaluation Process Overview

The Department of Defense Computer Security Center was established in January 1981 to encourage the widespread availability of trusted computer systems for use by facilities processing classified or other sensitive information. In August 1985 the name of the organization was changed to the National Computer Security Center. In order to assist in assessing the degree of trust one could place in a given computer system, the Trusted Computer System Evaluation Criteria [22] was written. The Criteria establishes specific requirements that a computer system must meet in order to achieve a predefined level of trustworthiness. The Criteria levels are arranged hierarchically into four major divisions of protection, each with certain security-relevant characteristics. These divisions are in turn subdivided into classes. To determine the division and class at which all requirements are met by a system, the system must be evaluated against the Criteria by an NCSC evaluation team.

The NCSC performs evaluations of computer products in varying stages of development from initial design to those that are commercially available. The following is a description of the process by which this product was evaluated. For a description of the current evaluation process, see the Trusted Product Evaluations Vendor Guide [29].

This product evaluation consisted of a developmental phase and a formal phase. All evaluations begin with the developmental phase. The primary thrust of the developmental phase is an in-depth examination of a manufacturer's design either for a new trusted product or for security enhancements tυ an existing product. Since the developmental phase is based on design documentation and information supplied by the industry source, it involves no "hands on" use of the system. The developmental phase results in the production of an Initial Product Assessment Report (IPAR). The IPAR documents the evaluation team's understanding of the system based on the information presented by the vendor. Because the IPAR contains proprietary information, distribution is restricted to the vendor and the NCSC.

Products entering the formal phase must be completely implemented security systems. In addition, the release being evaluated must not undergo any further development. The formal phase is an analysis of the hardware and software components of a system, all system documentation, and a mapping of the security features and assurances to the TNI. The analysis performed during the formal phase requires "hands on" testing (i.e., functional testing and, if applicable, penetration testing). The formal phase results in the production of a final report and an Evaluated Products List entry. The final report is a summary of the evaluation and includes the EPL rating which indicates the final class at which the product successfully met all TNI requirements in terms of both features and assurances. The final report and EPL entry are made public.

Document Organization

This report consists of eleven major sections and five appendices. Section one is an introduction and provides a brief system background and history. Section two provides a general overview of the VSLAN. Sections three and four describe the VSLAN hardware and software architecture, respectively. Section five describes the VSLAN subjects and objects, section six describes the software protection mechanisms implemented by the VSLAN. Section seven discusses additional assurances incorporated into the development of the VSLAN. Section eight discusses testing. Sections nine and ten provide a mapping between the requirements specified in the TNI for Appendix A and Part II, respectively, and the VSLAN features that fulfill those requirements. Section eleven presents some further comments on the VSLAN by the evaluation team. The appendices identify specific hardware and software components to which the evaluation applies, provide an overview of the Intel 80286, and provide a glossary of acronyms and a list of references.

PRODUCT OVERVIEW

The Verdix Secure Local Area Network (VSLAN) is a network component that is capable of interconnecting host systems operating at different security levels. The VSLAN mediates access between hosts and datagrams. The VSLAN controls access only to itself and does not mediate access attempts of host processes to information on host systems. It is intended to be used as a trusted building block upon which complete trusted network systems can be built. The VSLAN was developed to provide the following services to its hosts:

- a system bus interface
- a datagram-oriented communications service
- mediation of all data transfers between attached hosts in accordance with the VSLAN mandatory access control and discretionary access control policies
- identification and authentication of the individual responsible for operating a node of the network
- centralized management functions for security officers to exercise control over the operation of the VSLAN
- a capability to protect host datagrams and VSLAN control information against modification by random (e.g., transmission) errors

Figure 1 shows the security perimeter that isolates the VSLAN from its attached hosts. This perimeter represents a logical interface between the trusted hardware and software components of the VSLAN and the external host processing domain. The VSLAN consists of a single Network Security Center (NSC) and up to 128 Network Security Devices (NSDs) interconnected by the VSLAN transmission medium (i.e., coaxial cable). The NSC is a dedicated computer system that provides a c.ntralized management capability. Each NSD operates as an individual node of the VSLAN, providing a trusted network interface for its host by mediating incoming and outgoing datagrams according to the VSLAN security policy.

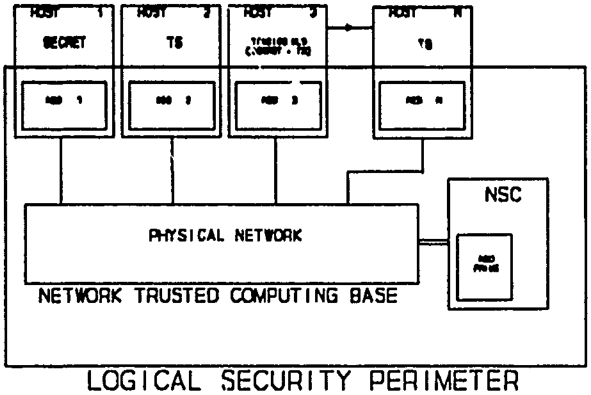

Figure 1 VSLAN Architecture

The NSC workstation provides the capability for a security officer to control and audit the operation of the VSLAN. The NSC continually monitors the status of all on-line NSDs and provides a time-stamped audit trail of all security relevant events and security officer commands. In addition to auditing security relevant events, the security officer has the option, through statistical (performance) auditing, to audit all host data transfers across the VSLAN on an NSD-to-NSD basis.

The NSC workstation provides a menu-driven user interface for the security officer to manage the VSLAN. It supports separate administrator and operator roles which are determined during login to the NSC. Each role (i.e., administrator or operator) has associated with it a specific set of control functions. The security officer is responsible for the operation of the VSLAN in accordance with the VSLAN security policy and must configure each NSD with the parameters necessary for the proper operation of the VSLAN. These parameters are then downloaded to each NSD where the VSLAN security policy is enforced.

The VSLAN uses the Data Encryption Standard (DES)[1] to encrypt all data transfers across the network. The VSLAN does not use the DES encryption algorithm as a mechanism to enforce the VSLAN security policy and therefore, the VSLAN B2 MDIA rating is not dependent on DES. The NSC acts as the key distribution center for the VSLAN and is the central repository for all encryption and decryption keys for authorized NSD-to-NSD and NSD-to-NSC data transfers.

The NSD is a trusted local area network (LAN) interface unit that provides the LAN communications and enforces the VSLAN security policy for all host-to-host data transfers. Each NSD must be configured by the security officer for operation at a single security level or at a range of security levels.

In order to use an NSD, a valid Datakey is required. A valid Datakey is one which has been programmed by a security officer for the individual responsible for the operation of a specific NSD[2]. This individual is called a principal. A Datakey is a physical device that contains the identification and authentication information for that principal as well as the encryption and decryption keys that will be used for the communication between that NSD and the NSC for that session (i.e., from initialization until shutdown of an NSD).

Relative to the VSLAN, the principal represents a collection of users that are associated with a host attached to a specific NSD. The principal initializes the NSD for which the Datakey was programmed by inserting the Datakey in the Keyceptacle on the NSD and turning it clockwise 90 degrees. The Datakey is valid only for that principal/NSD pair. Multiple principals (and therefore multiple Datakeys) can be associated per NSD; however, only one may be active at a time.

[1] The DES algorithm has been approved by the National Institute of Standards and Technology (NIST). The NCSC evaluation of the VSLAN did not include any examination of the DES implementation used by Verdix.

[2] The NCSC provides an interface for the security officer to program Datakeys.

Security Policy

The VSLAN defines subjects and objects according to the connection-oriented abstraction as described in the Introduction of the TNI. Specifically, subjects are hosts[1] and objects are datagrams (i.e., containers of information that are transmitted and received over the VSLAN communications medium).

The VSLAN security policy supports both discretionary and mandatory access controls. A datagram must pass both discretionary and mandatory access control checks before being allowed to be transmitted onto the LAN or received by the host.

The discretionary access control (DAC) policy allows two VSLAN subjects to exchange datagrams only if the security officer has authorized communication between them. All transmission attempts are mediated against a transmit association list (transmit access control list) that defines a list of destination hosts (NSDs) to which the sending host is allowed to send data. All data received by an NSD are mediated against a separate receive association list (receive access control list) that defines a list of hosts (NSDs) from which the host is allowed to receive data.

The mandatory access control (MAC) policy mediates access between VSLAN subjects and datagrams. Each datagram has a unique sensitivity label associated with it indicating its security level[2]. Each VSLAN subject has separate transmit and receive security windows associated with it. A security window is defined by a maximum security level and a minimum security level, where the maximum security level dominates the minimum security level, and consists of all security levels between these two[3].

The VSLAN MAC policy is enforced for all datagrams transmitted or received by VSLAN subjects. A VSLAN subject is allowed to transmit a datagram across the VSLAN only if the security level of the datagram dominates the minimum security level of the subject's transmit security window and is dominated by the maximum security level of the subject's transmit security window. A VSLAN subject is allowed to receive a datagram from the VSLAN only if the security level of the datagram is dominated by the maximum security level of the subject's receive security window and dominates the minimum security level of the subject's receive security window.

[1] More precisely, a VSLAN subject is defined as the combination of a host and a valid Datakey. For a more detailed discussion of VSLAN subjects see page 37, "VSLAN Subjects and Objects".

[2] A security level is defined as the combination of a hierarchical classification and a set of non-hierarchical categories that represents the sensitivity of information. The VSLAN supports up to 16 classifications and 64 categories.

[3] Security windows should be set to reflect the range of security levels of information allowed to be processed by the host system.

Communications

For the purposes of this report, VSLAN communications have been divided into external and internal communications. External communications are communications between an NSD and its attached host, whereas internal communications are those between two NSDs or between an NSD and the NSC.

External Communications

Each NSD provides a system bus interface to its attached host and acts as a memory device on its host bus. The host is allowed to access a 64 KB bank of the NSD's RAM. It is through this shared bank of RAM that the host and the NSD communicate. The NSD also uses interrupts and an I/O port to communicate with the host. Verdix supplies a different version of the NSD for each of the supported system bus interface. These interfaces are: IEEE P796 (Multibus-I), DEC Q22, IBM PC XT and PC AT, IEEE P1014/D1.2 (VME), IEEE 1196 (NuBus), and the AT&T 3B2.

Internal Communications

Communication over the VSLAN is controlled by associations. The ability to transmit is granted by a permission, called a transmit association. The ability to receive is granted by a permission, called a receive association. A pair of associations (transmit and receive) between two entities allows for communication in one direction. In order for a transmission to exist from NSD-A to NSD-B, NSD-A must have a transmit association for NSD-B, and NSD-B must have a receive association for NSD-A.

If an association grants access between two NSDs, it is called a data association; if it grants access between an NSD and a NSC, it is called a control association. Data associations are used to allow the exchange of user data between host systems. Each data association allows a one-way communication path from the source NSD to the destination NSD. Although an association may exist between two NSDs, communication may not be able to occur if the security windows of the two NSDs do not overlap. The data associations for a given NSD are collectively referred to as an association list[1].

Control associations allow the NSC to send network control information to each of the NSDs and to receive status and audit information from each NSD. A pair of associations (one in each direction) must always exist between the NSC and each NSD. Control and data associations between VSLAN components are shown in Figure 2, located on the following page.

[1] An association list for an NSD can be thought of as an access control list for authorized communication between two entities (e.g. NSDs). VSLAN associations must be set up by a VSLAN security officer.

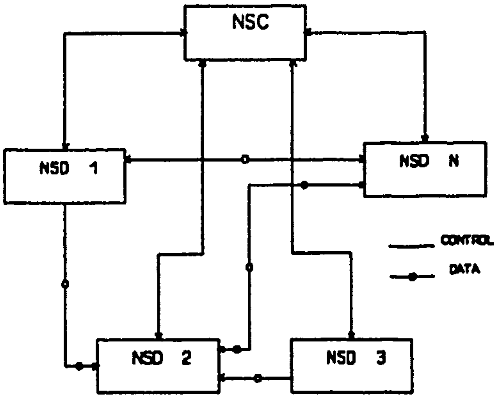

Fig 2 NTCB Control and Data Associations

Communications Protocols

Of the seven protocol layers defined in the Open System Interconnection (OSI) - Basic Reference Model [25], the VSLAN operates at the Physical and the Data Link layers, OSI layers 1 and 2, respectively. The VSLAN uses IEEE Standard 802.3 protocols to handle the physical layer and a portion of the data link layer communications[2]. IEEE 802.3, "CSMA/CD Access Method and Physical Layer Specifications", defines a protocol to establish an unreliable communications path between two nodes on a broadcast network. That is, there are no acknowledgements transmitted to indicate that datagrams have been received. Reliability (provided, for example, in the form of acknowledgements of correctly received packets, correct ordering of packets, etc.) must be provided by higher layer protocols.

The VSLAN Physical layer is concerned with the transmission of the bit stream over the VSLAN transmission medium. The Physical layer protocol conforms to the IEEE 802.3 Media Attachment Unit specifications.

[2] The Verdix Implementation differs from the IEEE standard in one respect. IEEE 802.3 defines a two-byte length, which indicates the length of a datagram. Verdix uses this field to identify the source NSD ID or principal ID (depending on whether a data or control association has been established). The VSLAN does not include an explicit length indicator with the diagram. Instead, the receiving NSD determines the end of a datagram by the quiescence of the line. It strips off the last 32 bits of the received message for CRC comparison, and thus determines the end of the data field.

The protocols residing at the VSLAN Data Link layer include the IEEE 802.3 Media Access Control Protocol, an encryption protocol, and a logical link control protocol. The IEEE 802.3 Media Access Control Protocol controls access to the VSLAN transmission medium by governing who can send information on the cable at what time, in order to minimize collisions. The VSLAN Media Access Control protocol implementation, except for its use of the length field, conforms to the IEEE 802.3 Media Access Control specifications.

The 802.3 protocols treat datagrams prepared by the encryption protocol as data. The encryption protocol is a Verdix-developed protocol that uses a NIST approved DES encryption algorithm to protect datagrams from being modified without detection while they are in transit across the network. The encryption protocol is used to protect both control and host datagrams.

Because of the need for reliable communications between the NSC and each NSD, the VSLAN protocol suite includes a logical link control protocol. The logical link control protocol provides a reliable data transfer service for network control datagrams only. This is accomplished by specifying separate data and acknowledgement datagrams. The receiver accepts only datagrams that are received in sequence, and generates an acknowledgement that identifies the received datagram by sequence number. The receiver transmits an acknowledgement for every datagram received. Packets prepared by the logical link control protocol are treated as data by the encryption protocol.

The VSLAN operates transparently to higher layer protocols (e.g., X.25, TCP/IP) implemented on hosts. Because of the VSLAN's independence of these upper layer protocols, it can be used to integrate a variety of host systems ranging from DoD inter-net gateways to vendor-specific systems. It is important to note that even though the VSLAN can support communications between different host systems, host systems must implement the same or compatible upper layer protocol suites (network layer and above) to be able to communicate with each other.

Operating Environment

The VSLAN is a network component designed to be used in a complete, network system NTCB. Figure 3, located on following page, shows the general architecture envisioned. For purposes of this discussion, the parts of the diagram inside of the inner boundary will be referred to as "region A," the parts of the diagram between the inner and outer boundaries will be called "region B," and the parts outside of the outer boundary will be referred to as "region C."

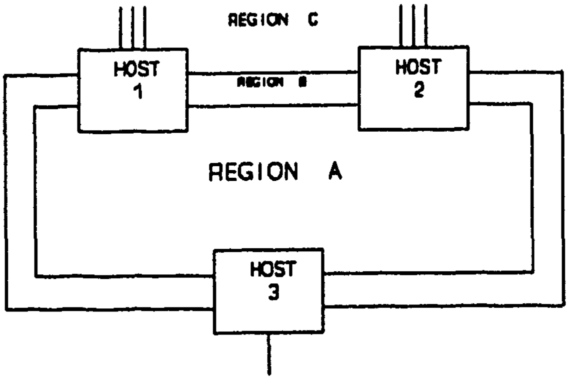

Figure 3 General Architecture

The VSLAN itself comprises region A. The boundary between regions A and B identifies the VSLAN NTCB partition interface. The TCBs of the hosts on the network system comprise region B and typically make use of the services available at the VSLAN NTCB partition interface. The boundary between regions B and C marks the network system NTCB interface. The functions visible at this interface would be expected to conform to the requirements of one of the classes defined in the TCSEC [22]. Region C is where all users, user subjects, and untrusted server subjects reside. Anything residing in region C is assumed to be outside the network system NTCB.

It is expected that the VSLAN will be used to interconnect hosts operating at different security modes and accreditation ranges allowing a multi-level secure (MLS) LAN operation. It is important to note that when using the VSLAN for classified MLS applications an appropriate protected wire distribution service must be provided and the hardware and software components that make up the VSLAN must be protected to the highest classification of data processed by the VSLAN.

Because the VSLAN is trusted to segregate data from hosts not authorized to handle data whose classification is outside its accreditation range, a network system incorporating the VSLAN could possibly allow hosts operating in MLS mode to be connected to the same LAN as hosts operating in dedicated or system high mode[1]. In addition to application environments involving the protection of classified information, the VSLAN could be used to protect unclassified and unclassified but sensitive information, which would include financial, proprietary, private, and mission-sensitive data.

[1] A B2 M-component can be used to separate data at two hierarchically-adjacent security levels (e.g. "secret" and "TOP SECRET"). For a discussion of computer security requirements for open and closed environments, see [24].

These application environments may be isolated VSLANs, or may involve connections to packet switched networks allowed to access other hosts or networks, including other VSLANs. When VSLANs are inter-networked to other end systems, the communication system used to connect the two systems must provide the required security services, even if the other end system is also a VSLAN.

A further discussion of potential operating environments in which the VSLAN can be used is available in [19].

System Integration

System Integrators must be aware that a trusted network system based on the VSLAN must combine its security services with those security services of the host operating system. Integrators should take care in defining the network system-wide DAC and MAC policies and its relationship to the VSLAN DAC and MAC policies.

When defining a network system-wide DAC policy, the approach for authenticating network connections at or above the transport layer should be considered. Appropriate authentication measures are necessary in support of a network system-wide DAC policy.

The VSLAN is capable of supporting a network system-wide DAC policy by using the VSLAN principal identifier as a group (i.e., host) identifier. When this group identifier is used as the basis for a network system DAC policy, it must be possible to associate the VSLAN principal identifier with a specific list of users authorized to use that particular host.

In order for the host system to support the VSLAN DAC policy, the host must observe the same convention for host addressing. The VSLAN uses NSD IDs to identify host addresses instead of standard Ethernet addresses, but the data for the NSD ID to Ethernet address mapping is made available to all host systems via the NTCB interface. The host is responsible for providing the addressing information for datagrams to be sent out over the network. The VSLAN does not support multicast or broadcast addresses.

In order for the VSLAN to be incorporated into a trusted network system, every entity in the network system must correctly interpret the sensitivity labels. This can be accomplished either by observing a uniform convention for labeling host datagrams, or by defining a mapping between each host's sensitivity labels and VSLAN labels.

System Integrators should also be aware of the assurance requirements for protocol layers three through seven. For untrusted VSLAN hosts, higher protocol layer software (i.e., layers three through seven) is not security relevant. Consequently, this would allow the use of commercially-available TCP/IP packages. However, for trusted VSLAN hosts, the assurance requirements of the protocol software depend upon the trust requirements of the particular host.

HARDWARE ARCHITECTURE

Introduction

This section describes the hardware architecture of the NSD board as well as that of the NSC workstation. The NSD board is designed and manufactured by Verdix. The NSC workstation is a standard Compaq workstation, modified to contain a special NSD board (the NSD-Prime), as well as Verdix-supplied software in EPROMs and on the hard disk of the workstation. Both the NSD and the NSC use the Intel 80286 microprocessor. The protection features of this processor are described later in this section. Detailed information on the Intel 80286 is contained in Appendix C.

The NSD is designed so that it can be implemented on a variety of different host bus architectures and form factors[1]. All NSD implementations utilize a common hardware design with the exception of the host-specific bus interface logic. Currently Verdix supports the following host bus interfaces: IEEE P796 (Multibus-I), IBM PC (both the PC and PC AT), DEC Q22, IEEE P1014/D1.2 (VME), IEEE 1196 (NuBus), and AT&T 3B2.

The NSC Workstation

The NSC workstation serves as the central control facility for the VSLAN. It is used to provide:

- a permanent storage capability for VSLAN configuration data
- an audit collection facility for VSLAN audit data
- an interface to the network for administrative personnel
- the capability to program authentication keys
- the capability to create a hard-copy of all audit information received

The NSC workstation consists of either the Compaq Deskpro 286 or 286e computer, configured with the Compaq enhanced keyboard, the Compaq color monitor, and an 80 column line printer. In addition the NSC workstation includes the Verdix NSD-Prime board and special EPROMs. The two standard BIOS PROMs for the workstation are removed and four of Verdix's own EPROMs are inserted. These EPROMs contain NSC initialization (boot-up) sofiware. Their use prevents the NSC from being used with standard operating systems like MS-DOS or Unix, and prevents booting from the diskette drive.

When configured as an NSC workstation, the Compaq 286 (an IBM PC/AT compatible microcomputer based on the Intel 80286), includes EPROMs which hold the boot code, 640 KB of RAM, a battery powered CMOS memory used to retain the time and date, a

[1] The term "form factor" refers to the dimensional characteristics os a plug-in board that must be met to insure mechanical compatibility with a particular bus.

Compaq VGA adapter, a fixed disk, a diskette drive, a parallel printer, and the NSD-Prime board. The NSD-Prime board is identical to the other NSD boards on the VSLAN; only its (EPROM) software is different (see page 28, "NSD and NSD-Prime Software"). It is used to program authentication keys and to provide the NSC with an interface to the network.

The NSC's fixed disk is a 40 MB hard disk, and its 5.25 inch diskette drive is capable of storing 1.2 MB of data per diskette. The 80 column printer is used as the NSC audit paper trail generator. It must have a Centronics interface and the speed must be at least 240 characters per second.

The NSC hardware is made up mostly of off-the-shelf components. Quality assurance tests are run on the hardware to demonstrate that the necessary capabilities exist for the proper execution of the NSC software. The tests are used to show the correct operation of the CPU (in protected mode), memory, disk subsystem, the interrupt controller and programmable timer, and all I/O controllers and devices.

The NSD Board

The NSD, a single printed circuit board, contains circuit components grouped into ten units. These are: the processor unit, triple port RAM, EPROM, function select and ready unit, timer and interrupt unit, bus buffer unit, ciphering unit, network interface unit, host interface unit, and the key interface unit. The interconnection of the units is shown on the next page in Figure 4, located on following page.

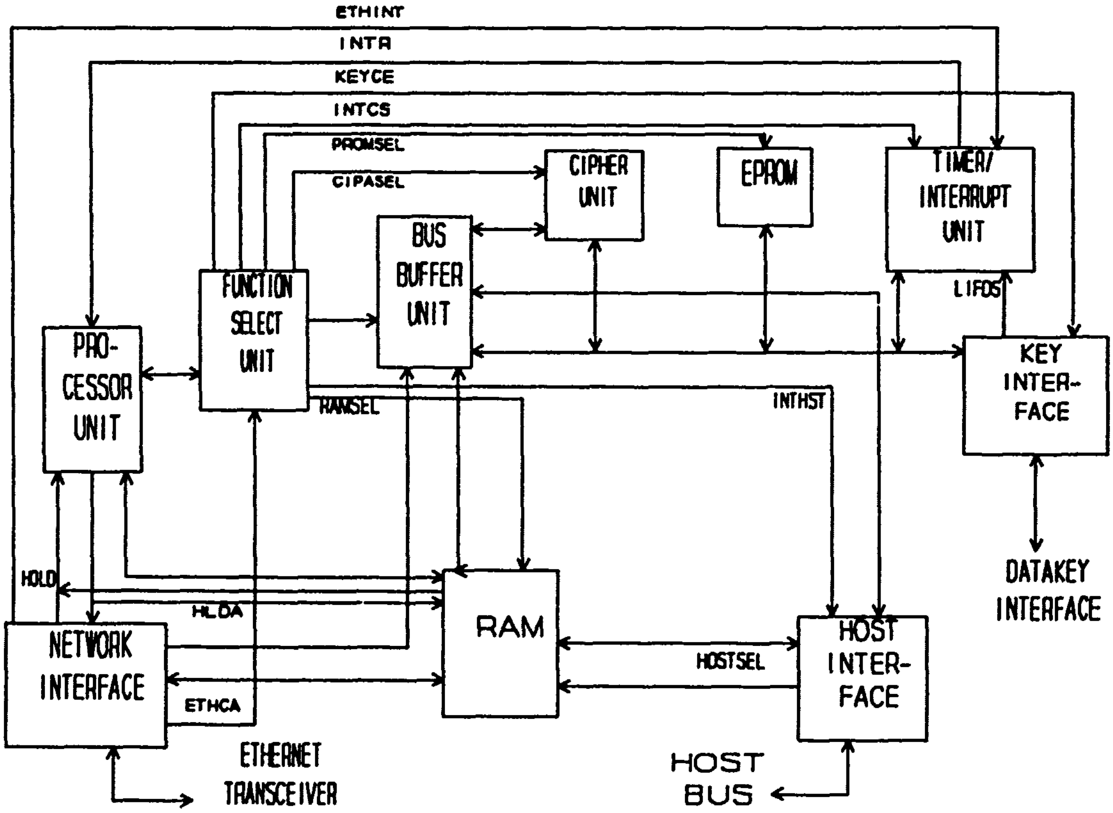

Figure 4 NSD Hardware Modules Interconnection Diagram

Processor Unit

The processor unit is made up of an Intel 80286 microprocessor and control logic. The Intel 80286 has built-in memory protection which supports the separation of programs from data areas within tasks, as well as isolation between tasks, and between each task and the operating system. It has four levels of privilege which are used to provide isolation. Gates are used to assure controlled, well-defined access points into more privileged routines. These protection features are described in an appendix (see page C-1, "Intel 80286 Hardware Overview").

The processor unit is the only module of the VSLAN that can access all of the memory space and I/O units on the NSD board, and it can do this only while in protected mode. When in real mode, the Intel 80286 is restricted from accessing memory addresses greater than one Megabyte, except during initialization following a hardware reset, and then, only until the value of the program location counter is changed from its initialization value. The real mode memory map is shown in Figure 5. In the VSLAN, real mode is used only for the first stages of initialization.

Address Range (in Hex)	Description
000000 - 00FFFF	Bank 1: Local RAM
010000 - 01FFFF	Bank 2: Host/NSD Shared
020000 - 07FFFF	ILLEGAL ADDRESSES
080000 - 080FFF	Real Mode I/O Port Addresses
081000 - 0FFFFF	ILLEGAL ADDRESSES
100000 - FEFFFF	Not Accessible in Real Mode
FF0000 - FFFFFF	EPROM (start-up only)

Figure 5 Real Mode Address Map

Following initialization, the VSLAN switches the processor to protected virtual address mode (protected mode). Through privileged instructions, protected mode provides memory protection to isolate the operating system and insure the privacy of each task's programs and data. The protected mode memory map is shown in Figure 6.

Address Range (in Hex)	Description
000000 - 00FFFF	Bank 1: Local RAM
010000 - 01FFFF	Bank 2: Host/NSD Shared
020000 - 07FFFF	ILLEGAL ADDRESSES
080000 - 080FFF	Real Mode I/O Port Addresses
081000 - 0FFFFF	ILLEGAL ADDRESSES
100000 - 10FFFF	Bank 3: Local RAM
110000 - 11FFFF	Bank 4: NSD/Network Shared
120000 - 1FFFFF	ILLEGAL ADDRESSES
200000 - 20FFFF	Bank 5: Local RAM
210000 - 21FFFF	Bank 6: Local RAM
220000 - 2FFFFF	ILLEGAL ADDRESSES
300000 - 30FFFF	Bank 7: Local RAM
310000 - 31FFFF	Bank 8: Local RAM
320000 - DFFFFF	ILLEGAL ADDRESSES
E00000 - E00FFF	Datakey EEPROM
E01000 - E3FFFF	ILLEGAL ADDRESSES
E40000 - E40FFF	Ciphering Block
E41000 - E7FFFF	ILLEGAL ADDRESSES
E80000 - E80FFF	Protected Mode I/O Ports
E81000 - FEFFFF	ILLEGAL ADDRESSES
FF0000 - FFFFFF	EPROM (Program Storage)

Figure 6 Protected Mode Memory Map

Triple Port RAM

Eight 64 Kilobyte banks of RAM make up the 512 KB triple port memory. They are grouped into three blocks. These blocks are: 384 KB local RAM, accessible only to the Intel 80286; 64 KB host/NSD shared memory (called host dual port RAM by the vendor), accessible to both the host and the processor; and 64 KB network dual port RAM, accessible to both the network interface unit and the processor. The host/NSD shared memory provides the principal interface to the NTCB; because it is directly modifiable by an external host, it is not considered to be part of the NTCB itself.

The separation of the triple port RAM into three separate blocks, each with its own access rules, provides assurance that the network interface unit is physically prevented from accessing the host/NSD shared memory, and the host is prevented from accessing the network dual-port RAM. Both the network interface unit and the external host are prevented from accessing the local RAM of the NSD. For added assurance of this separation, an address monitoring circuit located in the triple port RAM monitors all memory accesses. Any access from the host interface to any memory other than the host/NSD shared memory, or any access from the network interface unit to other than the network dual port RAM, causes the CPU to stop operation.

The triple port RAM is controlled by the Intel 8207 Advanced Dynamic RAM Controller. The controller allows two different buses to access memory independently by use of a dual-port interface. The 8207 also provides the signals necessary to refresh, address, and directly drive the dynamic RAM. A single parity bit is associated with each byte in the triple port RAM. If a parity error is detected, the unit generates a signal to halt the CPU. The address range (as seen by the host) of the host/NSD shared memory is set by jumpers in the host interface unit[1] which specify the beginning address of the 64 KB block.

EPROM

The EPROM of the NSD resides at the highest physical address (FF0000H - FFFFFFH). This unit contains 64 Kilobytes of program memory. All NSD software resides in this EPROM, giving assurance that software integrity is maintained, provided physical access to the NSD board is restricted.

Function Select and Ready Unit

This unit controls the other units on the NSD board. Based on addresses generated by the CPU, this unit enables the control signals to the other units, and then waits until the selected unit has finished its read or write command before generating a ready signal[1] to the CPU to continue processing.

This unit also includes a status display latch that controls four LEDs located on the board. These LEDs give visual indication of program status or errors.

Timer and Interrupt Unit

The timer and interrupt unit contains the programmable timer and the programmable interrupt controller.

The Intel 8259 Programmable Interrupt Controller handles up to eight vector priority interrupts for the CPU, each of which can be masked individually. The interrupt with the highest priority (interrupt zero) comes from the network interface unit. The interrupt with secondary priority (interrupt one) comes from the host, while interrupts two and three are not used. Thereafter, the interrupts, in descending order of priority, are: Datakey insertion, Datakey removal, and two timer interrupts.

The Advanced Micro Devices 9513 Programmable Timer Device provides five independently-programmable hardware timers, two of which can generate interrupts to the CPU.

[1] For the NuBus, a configuration ROM is used rather than jumpers. For the AT&T 3B2, the memory address range is determined by the board's bus slot rather than by jumpers.

[1] The triple port RAM is the only unit that generates its own ready signal.

Bus Buffer Unit

This unit is used to buffer the local address bus, data bus, and control signals. All units, except for the triple port RAM, host interface, and network interface unit, are connected to the bus buffer unit. The buffered address is a latched signal that remains valid for an entire machine cycle.

Ciphering Unit

The ciphering unit performs the encryption and decryption for the VSLAN. The primary chip used in this module is the Advanced Micro Devices AM9568 Data Ciphering Processor, which implements the DES algorithm.

Network Interface Unit

This module consists of the Intel 82586 LAN co-processor and a SEEQ 8023 Ethernet Data Encoder. This unit performs the exchange of data between the 64 KB network dual port RAM and the network. It can support both the IEEE 802.3 Media Access Control Protocol interface and the DEC-Intel-Xerox standard Ethernet interface [34]. However, the DEC-Intel-Xerox standard Ethernet interface is not in the evaluated configuration.

The 82586 communicates directly with the NSD's CPU using the channel attention (CA) and interrupt (INT) signals. The CA signal is used by the processor unit to indicate to the 82586 that a datagram has been placed into shared memory for transfer onto the network. The INT signal is used by the 82586 to indicate to the processor unit that a datagram has been placed into shared memory.

In addition to transmitting and receiving datagrams, the 82586 handles link management algorithms according to the IEEE 802.3 standard, with minor exceptions (see page 9, "Communications"). The 82586 performs the CSMA/CD link access, framing, preamble generation and stripping, source address generation and checking, and CRC generation and checking. These mechanisms help provide data packet integrity.

Host Interface Unit

The host interface unit attaches to the host bus and allows the external host to access the host/NSD shared memory. The unit uses a host interrupt signal to notify the host that the NSD has placed data into the RAM, has accepted host data, or to signal an error, and uses a host I/O port or memory-mapped I/O for the host to notify the NSD that the host has placed data into the RAM.

This is the only unit of the NSD that is unique to a particular host bus. It contains the jumpers[1] that are used to specify the memory address range that the host uses in accessing the host/NSD shared memory, and to specify the address and interrupt line used for signalling between the NSD and the host.

[1] See the footnote on page 16, "the NuBus".

The host interface to the RAM is disabled when the triple port RAM parity checking is turned off. If the host system attempts to read any shared memory location when parity checking is off, the host interface responds with random data. The NSD software can also disable the host interface to the RAM by activating a signal.

Key Interface Unit

This unit interfaces the NSD to the processor key system, which consists of a portable memory device (the Datakey) and the Datakey KCPKA16KS Keyceptacle. The Datakey must be physically inserted into the Keyceptacle in order for the NSD to operate. The Datakey contains 2 KB of EEPROM.

SOFTWARE ARCHITECTURE

All software in the VSLAN is considered part of the NTCB. VSLAN software can be divided into five distinct groups:

1. The NSD and NSD-Prime Separation Kernel (SK)
2. The NSD Application Tasks (AT)
3. The NSD-Prime Application Tasks (AT-Prime)
4. The Verdix Operating System (VOS)
5. The NSC Application Processes (NAP)

This section discusses the relationships between these five groups to provide a context for the discussions of each individual group that follow.

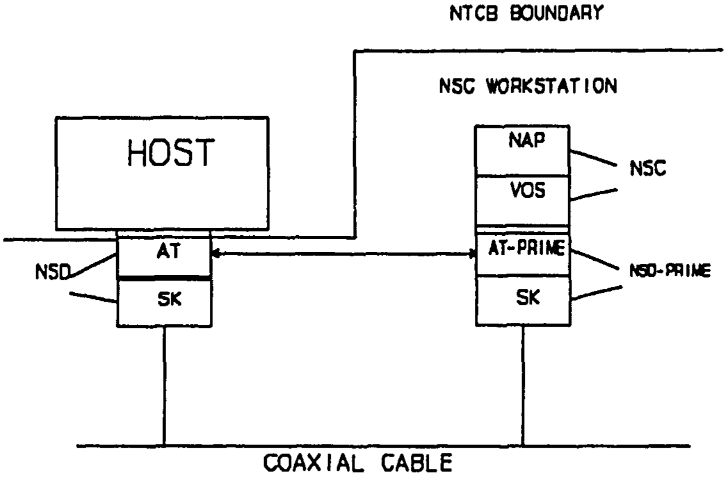

Figure 7 VSLAN Software Organization

Figure 7 shows how the various groups interact with each other. The core of the VSLAN is the NSD and the software which runs on it. Each NSD has a separation kernel; this kernel is a set of procedure calls that provide a simple multi-tasking, message-passing monitor for use by application tasks running on the NSD. For a more thorough description of an Intel 80286 task, see page C-2, "Security Features". All communication among VSLAN processes (both in the NSDs and the NSC) takes place by message-passing. The ATs (and the AT-Primes, since they are largely the same set) are a group of tasks that each perform a specific NSD function. The tasks communicate using the message-passing primitives provided by the SK.

One of the tasks in the AT-Prime and in each AT is the network control task. The NSD network control task is the central controller for each NSD. The NSD-Prime network control task maintains a connection with each NSD network control task, as indicated by the dotted line. It uses this connection to issue commands to the NSD network control task. The recipient acts on those commands to set and change the NSD's MAC, DAC, and operational parameters, as well as turn itself on and off. These control commands, in turn, originate from the NSC.

The NSC is composed of the NSC applications processes (NAPs) and the Verdix Operating System (VOS). The NAPs provide external control of the network control task mentioned above. The NAPs do not have direct access to the hardware; instead, they work through VOS. VOS provides the NSC with a multitasking operating system interface with message-passing facilities similar to those used for inter-task communication on the NSDs. It also provides interfaces to the additional devices present on the NSC workstation.

The host/NSD shared memory is also used by each NSD to communicate with its associated host computer. Communication between the NSC and the AT-Prime takes place using similar shared memory. A more detailed explanation of host/NSD shared memory usage is provided later in this section.

VOS and the SK are related software components that control the Intel 80286 hardware for the NSC Workstation and the NSD/NSD-Prime board, respectively. Both use the 80286 facilities for process isolation and privilege to provide strong data hiding and least privilege among the tasks they control. The specific mechanisms used are explained more completely in the descriptions of VOS and the SK. A more detailed explanation of the hardware usage is explained in the following section.

Protection features of the Intel 80286

The NSD uses a number of protection features inherent to the 80286 microprocessor[1]. These features include Privilege Level Separation, Module Data Privacy, Data Segment Length Specification, Data Segment Access Limitation, and Code/Data Access Attribute.

All tasks that execute on the NSD are trusted. However, the VSLAN still takes advantage of all four privilege levels that the 80286 provides. The main system control and management functions (including most of the NSD Kernel procedures like the Memory Management procedures, the Inter-Task Communications procedures, etc.) are placed at privilege levels 0 and 1. Functions in need of the most protection (e.g., Boot Task) are located in level 0. Gates are established for the application tasks to access the kernel code. The gates provide controlled, well defined access points into more-privileged modules. This prevents an application task from entering a more-privilege module in the middle of the procedure (or within an instruction). The hardware control procedures are placed at privilege level 2 because of the actual hardware interaction between the code and the NSD board that is required. The NSD's application tasks are at level 3, the least privileged level, to allow the NSD kernel to keep its data structures hidden from view. A

[1] For a more detailed discussion of the 80286 and these protection features, see page C-1, "Intel 80286 Hardware Overview".

task changes its current privilege level (CPL) when it shifts execution to a procedure at a different privilege by a CALL through a gate.

Each task has its own private data segment and a stack segment for each privilege level at which it executes. These segments, along with the task's code segment, are declared in the local descriptor table (LDT) of the task. As a result, the data segment of a particular task can be accessed only by that task and by no other. Any other attempts at access will result in a protection violation. This feature enhances data separation and integrity for the application tasks. It also supports least privilege, in cooperation with software, by not allowing a task to have access to any data other than that which it needs.

Separate segments are used for different data structures (such as tables). The lengths of these data structures are specified in the segment declaration, thereby preventing any accidental overflow into a neighboring data structure. Data structures are usually accessed by use of index pointers. An erroneous index parameter, or a problem in the software, could lead to an attempt to access beyond the structure limits, thereby destroying other data. Because the exact length of the data structure is specified, any such erroneous access attempt will result in a protection violation.

Critical data structures (e.g., encryption tables, DAC tables, security window tables, etc.) which are used by multiple tasks are installed in the LDT of each task that has need to access it. Data structures residing in the global descriptor table (GDT) can be accessed by all of the tasks. Any attempt by a task to access a data structure which is not in its LDT, nor in the GDT, will cause a protection violation.

The VSLAN assigns to each segment one of the following Intel 80286 access attributes: read/write or read/execute. These attributes are then checked by the 80286 hardware against the access desired for the target address range. Code segments are marked as read/execute and any attempt to write into such a code area will result in a protection violation. Data segments are marked as read/write. A protection violation will occur if an attempt is made to execute such a segment as code.

NSD and NSD-Prime Software

The NSD and the NSD-Prime contain three of the five groups of software. The Separation Kernel is the same on both the NSD and NSD-Prime boards and serves as a standard mechanism for memory management, inter-task communication, and hardware control. The NSD and NSD-Prime application tasks perform the data communication functions and enforce the VSLAN security policy.

Separation Kernel Service Groups

As mentioned above, the NSD Separation Kernel (SK) is a set of procedure calls which provide the basis for a multi-task environment in which the tasks communicate with each other by passing messages. The SK consists of a set of procedures broken into eight service groups. Each service group provides a number of related services to client tasks.

Tasks are related sets of procedures, each of which performs a specific function. The service groups are distributed between three Intel 80286 execution rings (0-2).
Ring Zero Software

The service groups executing in ring zero are the utilities group and the memory management group. The utilities group consists of 9 procedures. These procedures are in this group because they are necessary to the proper functioning of the SK and the APs, but do not logically belong under any of the other groups. Included in this group are procedures: for initializing the other utility procedures, handling fatal errors, enabling and disabling interrupts, identifying a calling procedure's privilege level, and halting the Intel 80286 processor.

The memory management group consists of nine procedures to manipulate the data structures relevant to managing memory on the NSD. The basic unit of memory on the NSD is the buffer, which is a fixed-size block of memory. Dynamic user memory is organized as a queue of buffers, which can be allocated and deallocated very quickly. The SK makes full use of the Intel 80286 protection features for memory management by using only the LDT of a task to allow that task access to a particular buffer. As the GDT is not used, there is less opportunity for undesired data flows. Included in this group are procedures: for initializing memory management services; initializing and expanding the buffer pool; initializing, installing, and removing LDT entries; and management of memory buffers.

Ring One Software

The service groups executing in ring 1 are the task management group and the inter-task communication group. The task management service group consists of six procedures which provide the mechanisms for initializing and switching between tasks. The process isolation features of the Intel 80286 are used extensively in the SK for task management. Each task is assigned its own local descriptor table (LDT) for memory accesses. Along with the careful design of the tasks themselves, this provides effective support for least privilege and data hiding. Task code, stack, and data segments must already be in memory when task management is invoked to start running the task. Only 32 tasks may be running simultaneously. No task scheduling is done by task management services.

Included in this service group are procedures: for initializing task management services at system startup; making a task known to the kernel by allocating and initializing a task control block and LDT for it; identifying the currently running task; putting the current task to sleep and initiating the next scheduled task; and signalling the hardware to perform a task switch.

The inter-task communications (ITC) service group consists of five procedures that implement an inter-task messaging system. Scheduling in the NSD operates as follows: There are nine levels of priority assignable to inter-task messages. The lowest priority is level one, and the highest priority is level nine. Those tasks that have higher priority ITC messages waiting for them will be executed ahead of those with lower priority messages. Those with the same priority will be executed in a FIFO queue. ITC messages may contain timeout messages, network commands received from the NSC, or regular NSD datagrams received from the network. Timeout messages are priority six, NSC commands are priority level three, regular NSD traffic is priority two, and a wait message (sent by a Null task to itself) is priority one. This ensures that those tasks that process timeouts and control messages will be executed ahead of those that process data transfers. If there are no NSC commands and no data traffic (i.e., the network is not in use), the NSD enters a busy-wait state. In this state, a task called the Null task will continually send a message to itself at priority one, so that there is always at least one message being processed and one task active.

Additionally, some scheduling is performed not with ITC messages but with interrupts. Those tasks that process data traffic through the host/NSD shared memory receive interrupts at priority three if receiving data or priority two if transmitting data. This ensures that incoming traffic is processed sooner than outgoing traffic. If congestion arises, it is easier for the NSD to tell the local host to slow down than to tell the remote host to transmit more slowly. Those tasks that control the Ethernet interface receive interrupts at priority three if receiving traffic and at priority four if transmitting traffic. This ensures that the NSD will always attempt to transmit available messages, without being stifled by incoming data.

In addition to the priority field, messages also have a buffer pointer field. The buffer pointer field specifies a buffer that is associated with the message. As a part of the message delivery, the descriptor for that buffer is moved from the source task's LDT to the destination task's LDT. This assures that only the task working with the buffer has access to it.

Included in this group are procedures: for initializing inter-task communications structures at startup time; sending and receiving inter-task messages; blocking inter-task messages below a specified priority; and determining the destination task of the message at the head of the queue.

Ring Two Software

The software in ring two consists of three service groups: the initialization group, the timer services group, and the hardware services group. The initialization service group is one procedure whose purpose is to drive the initialization of the kernel. It calls procedures in other service groups to do the specific initialization tasks. When it has finished its processing, the kernel will be in a known state. This guarantees that when the other kernel services are invoked again, they will not find spurious data that could be misinterpreted.

The timer services service group consists of procedures which allow tasks to work with alarm timers. A calling task can cause a timer to be started; when the timer expires, timer services formats an inter-task message, with a priority that was specified by the calling task, and sends it to the calling task. Included in this service are procedures: for initializing timer services so that there are no outstanding timers; starting and cancelling timers; and handling the interrupt from the hardware timer.

The hardware service group consists of procedures which serve as device driver interfaces between the base hardware and the rest of the application tasks. They are provided so that if the NSD hardware is ever changed, only a small, well-defined set of procedures need be changed. Included in this service group are procedures to interface to the programmable interrupt controller; procedures to interface to the timer controller chip; procedures for exchanging signals with the host computer; procedures for memory control; procedures to interface to the DataKey hardware; procedures to interface with the network hardware; and routines for turning LEDs on and off and for setting up the NSD's interrupt descriptor table.

Conforming Code Segments

The NSD software also includes a number of conforming code segments. These are routines that reside at privilege level 0 but may be executed by software at any level. When called, the routines execute at the privilege level of the caller. These particular routines are used for queue services. Queueing services consist of procedures which provide a consistent mechanism for initializing and using a queue data-structure. It should be noted that the queues in the NSD are implemented as arrays in the caller's address space and that the only functions performed by queueing service are initializing, ordering, adding, and deleting the elements in the array. The client procedure is responsible for copying data to and from actual memory. Included in this group are procedures: for creating the ordered queue, adding and deleting queue elements, and stepping through the elements of the queue.

NSD and NSD-Prime Data Structures

The NSD and NSD-Prime have the same set of data structure types. These data structures are the encryption protocol table, the decryption protocol table, the LLC transmit table, the LLC receive table, and the interface control block (ICB). In addition, the NSD has four distinct data structures concerned with MAC and DAC. These are the transmit association table, the transmit window, the receive association table, and the receive window. The transmit association table contains the list of NSD IDs of other hosts on the LAN to which the local host may send datagrams. The receive association table contains the NSD IDs of the remote hosts from which the local host may receive datagrams. The transmit window and the receive window each contain the minimum and maximum security levels for the NSD. These security levels designate the range at which the local host may transmit and receive data. The LLC transmit table and the LLC receive table are used to support the reliable data path between the network control tasks on the local NSD and the NSD-Prime. The encryption protocol table contains information used to encrypt an outgoing datagram, and the decryption protocol table contains information used to decrypt an incoming datagram.

The ICB is the data structure through which the host and the NSD communicate with each other. The format of the ICB is described in Figure 8. A hardware interrupt mechanism is provided for each side to get the other's attention. Upon receiving an interrupt, the receiving side then examines the ICB to ascertain what data has been placed in the shared memory, where it was placed, and its length. The ICB contains pointers for two structures called the transmit frame list and the receive frame list. These lists temporarily store frames to be transmitted or to be received by the local host. A frame is the physical representation of a datagram, containing user data and header information necessary for transmission from its source to its destination, including Ethernet network addresses, frame length, and security level. The host interfaces with the LAN by appending frames to the transmit frame list for transmission on the network, and receives incoming datagrams by reading from the receive frame list. The ICB also contains a copy of all security information relevant to the local host: maximum and minimum security levels; discretionary association lists; principal identification; and local NSD ID. This information is available for the host to determine its own security parameters. The remainder of the ICB is a set of status and control bytes used for signalling between the host and NSD.

Field	Beginning Address
Initialization Pattern	00H
NSD Status Interrupt	02H
Status Interrupt	03H
Receive Frame Ready	04H
Frame Transmitted	05H
Start Transmit List	06H
Continue Transmit List	07H
Start Receive List	08H
Continue Receive List	09H
Transmit List Offset	0AH
Receive List Offset	0CH
NSD-ID	0EH
Maximum Transmit Security Classification	10H
Minimum Transmit Security Classification	12H
Allowable Transmit Category Set	14H
Required Transmit Category Set	1CH
Maximum Receive Security Classification	24H
Minimum Receive Security Classification	26H
Allowable Receive Category Set	28H
Required Receive Category Set	30H
Transmit Discretionary List	38H
Receive Discretionary List	78H
Principal Identification	B8H
Ethernet Vendor ID	C0H
Ethernet Address Display Table	C4H

Figure 8 Interface Control Block Fields

NSD Application Tasks

The NSD tasks implement the datagram transmission and reference validation mechanism functions for the VSLAN NTCB. The tasks are grouped into four broad categories: NSD startup, basic communications flow, network control, and audit.

NSD Startup Tasks

The tasks involved in the NSD initialization procedure are the system boot task, the system initialization task (SIT), the network control task, the audit task, the network interface transmit task, and the network interface receive task. Initialization is performed both at system power-on and each time the Datakey is used. Initialization takes place as explained later in this report (see page 51, "Secure Initialization and Shutdown") .

Basic Communications Flow

The primary purpose of the NSD is to transfer data between its own host/NSD shared memory and the host/NSD shared memory of another NSD. This section discusses the path that a datagram takes through the NSD.

Transmit Flow:	Receive Flow:
Host/NSD Shared Memory	Network Interface Interrupt Task
External Interface Interrupt Task	Network Interface Receive Task
Ext. Interface Transmit Task	Decryption Protocol Task
Transmit Policy Control Task	(Network Control Task)
(Network Control Task)	Receive Policy Control Task
Encryption Protocol Task	External Interface Receive Task
Network Interface Transmit Task	Host/NSD Shared Memory

Figure 9 Basic Communications Flow in an NSD

The local host initiates the process of transmitting a datagram by placing a linked list of frames and control information for that list into the host/NSD shared memory and informing the NSD that it has done so. This invokes the external interface interrupt task, which notifies the external interface transmit task that there is work to do. Unless it has been told to suspend processing by the network control task or the LLC transmit task, the external interface transmit task then takes the next frame in the list and copies it into a buffer in the NSD local memory. It then attaches information to the datagram regarding the security level and category set and the source and destination NSD IDs for the datagram, all of which was supplied by the host in the frame header. It also attaches the length of the datagram read from the shared memory. The datagram is then passed to the transmit policy control task. The host is notified that a frame has been read, and that the space is available for more frames. The external interface transmit task repeats this process for each frame in the linked list.

The transmit policy control task is responsible for performing the DAC and MAC mediation on all datagrams leaving the NSD. Upon receiving a datagram from the local host, it first checks to see if that host has a transmit association with the destination host. If such a transmit association exists, it then checks the host-supplied sensitivity label against the allowed range of security levels. If both checks are passed, it then calculates a checksum for the security header field and inserts it into the datagram. The frame is then passed to the encryption protocol task for delivery to the network. If any security mediation checks fail, the datagram is not delivered and the failure is audited.

When the encryption protocol task receives a datagram, it gets the next encryption protocol sequence number for that association from the encryption protocol table, inserts it into the datagram, and increments the sequence number. It then calculates two checksums (one for the datagram header and another for the header, its checksum, and the data portions) and inserts them into reserved fields. Next, it acquires a new buffer, into which it places source and destination network addresses and either the NSD ID for a host datagram or the local Principal ID for a control datagram. The datagram is then encrypted and is placed into the new buffer. Encryption is performed with the AM9568 Data Ciphering Processor using NIST-approved DES algorithm. The VSLAN uses the Cipher Block Chaining mode. This mode operates on 64-bit blocks and includes a feedback step. If there have been no problems, the new buffer is passed to the network interface transmit task for transmission on the LAN. If the cipher unit cannot obtain the key (this may occur if an add association command has been carried out, but a key distribution command has not), an audit message is generated and the datagram is destroyed.

The network interface transmit task is responsible for moving the datagrams from local to network shared memory and invoking the network interface chip to send them. To do this, it copies the datagrams from its outstanding inter-task messages into a network transmit linked list in network shared memory until it runs out of either shared memory or datagrams. It then instructs the network interface unit to send the messages and suspends itself until either a success or failure response is received. If the send is successful, the network interface transmit task then starts the process again with any other outstanding datagrams. If the send fails because the network interface unit reported a fatal transmission error, the processor is halted.

At the receiving end, the network puts packets destined for the local host into the NSD/network shared memory. In this case the flow is basically the reverse of the transmission side. The network interface chip sends an interrupt which is converted by the network interface interrupt task into an inter-task message for the network interface receive task.

The network interface receive task first determines whether the received datagram is from the NSC or another NSD. It then tries to allocate a local buffer using the SK. If the network control or LLC transmit tasks have suspended the network interface receive task, or there are no buffers available, host datagrams (datagrams from another NSD) are discarded. Control datagrams (from the NSC) are allocated a buffer from a reserved buffer area. This reserved area is made sufficiently large to make discarding a control datagram unlikely. If there are available buffers, the datagram is copied from the network/NSD shared memory into the allocated buffer and passed to the decryption protocol task. The decryption protocol task is responsible for decrypting and validating the datagram received from the network. Its first task is to decrypt the datagram, which it does by accessing the decryption protocol table based on the NSD ID or Principal ID. If the NSD ID yields a valid table entry, the data portion of the datagram is decrypted and written into a newly acquired buffer; otherwise, the datagram is discarded and the header information is written into an audit record. Next the encryption protocol sequence number is checked against the expected sequence number. No packet will be discarded because of a sequence number discrepancy; however, sequence numbers that are not within an expected range will cause an audit record to be generated. Regardless, the decryption protocol table "next sequence number" field will be updated. After checking the sequence number, the header and data packet checksums generated by the remote encryption protocol task are checked. If there is any discrepancy, an audit record is generated and the datagram is discarded. If the checksums are correct, the NSD IDs inserted by the remote encryption protocol are checked; if either is incorrect, the header information goes into an audit record and the datagram is discarded. After passing the previous checks, a datagram is passed to either the receive policy control task, if it is a host datagram, or the LLC Receive task, if it is a control datagram. When there are no more inter-task messages for it, the decryption protocol task is suspended by the SK.

If the datagram is a control datagram, the LLC receive task first determines whether it is of type data or acknowledgement. If it is an acknowledgement, the datagram is passed to the LLC transmit task for processing and the next datagram message is requested. If it is data, the LLC receive task first gets a buffer and formats a datagram acknowledgment, which it then sends via inter-task message to the encryption protocol task for transmission to the NSD-Prime. Next the control datagram sequence number is checked and the LLC receive table "next sequence number" field is updated accordingly. If the sequence number is what is expected, the control datagram is passed to the network control task; otherwise, the datagram is discarded by releasing the buffer. The LLC receive task continues until all control datagrams have been processed. It then suspends itself.

If the datagram is not a control datagram, it is passed to the receive policy control task for DAC and MAC mediation. Upon receiving a datagram, it first checks the NSD receive association table to see if the local host has an association with the source host. If it is, the task then checks the host-supplied sensitivity label against the allowed range found in the receive window table. If any security mediation checks fail, appropriate audit data will be generated and passed to the audit task and the datagram discarded. If both checks are passed, it then calculates a checksum for the security header fields and checks it against the one inserted by the source host. If the checksums do not match, an audit record is generated and the datagram is discarded. If all checks are passed, the datagram is passed to the external interface receive task for delivery to the host.

The external interface receive task takes the datagrams passed to it and places them in host-defined frames in the host/NSD shared memory. If there are no frames available, it suspends host datagram processing, and waits for the host to provide more frames. Upon successful transfer of a datagram from NSD local memory to host/NSD shared memory, the local buffer is released back to the SK.

Network Control

The NSD software also provides several mechanisms to support network control. These include mechanisms for setting and updating the NSD tables that are the basis of the NSD access mediation, suspending and resuming host datagram processing, and turning statistical auditing on and off. Each NSD locally supports these mechanisms by a set of three tasks which work together to provide a reliable communication path between the NSD and the NSD-Prime.

The largest and most important of these three tasks is the network control task. The primary purpose of this task is to interpret commands that have been sent from the NSC and act upon them. The network control task receives commands in a control datagram from the LLC receive task. The control datagram may contain commands from the NSC for initialization, auditing, altering security windows, etc. The complete list of commands is in the VSLAN Trusted Facility Manual [16]. The actions taken as a result of these commands are usually updates to the NSD data structures. This task is also responsible for suspending or shutting down processing under certain conditions. Specifically, if a suspend order is issued, the NSD will tell the external interface transmit and network interface receive tasks to stop processing host datagrams. If a status request order is not received from the NSC within a certain time period or a shutdown order is received from the NSC or the authentication key interrupt task sends an inter-task message indicating that the DataKey is no longer in its receptacle, the network control task will shut down the NSD.

The other two tasks which support the network control task are the logical link control (LLC) transmit and receive tasks, which provide a reliable, end-to-end communications path between the NSDs and the NSD-Prime.

Audit

Audit on the NSD is accomplished primarily by the individual application tasks where auditable events occur. When an auditable event occurs in one of the tasks, it writes information relevant to the event into the audit storage area and sends an inter-task message to the audit task to let it know that it should read data from this area. The tasks that can generate audit data are the transmit and receive policy control tasks and the encryption and decryption protocol tasks. The network control task regulates auditing with a flag telling the transmit and receive policy tasks whether to produce statistical audit data.

When the audit task receives a message that there is audit data, it reads the audit storage area, formats the data into a datagram, and passes the audit datagram to the LLC transmit task for delivery to the NSD-Prime. If the number of audit messages exceeds a pre-set threshold, the audit task sends an internal suspend message to the network control task. The threshold counter is reset by the network control task each time it receives a status poll from the NSC. Note that thresholds do not apply to statistical auditing. Once a packet has been sent to the LLC transmit task for transmission, the audit task sends a "stop" message to the network control task (NCT), which then sends messages to the interface tasks to tell them to stop processing host datagrams. When the NSC acknowledges receipt of the audit datagram, the NCT commands the interface tasks and the audit task to resume regular processing.

NSD-Prime Task Organization

The NSD-Prime's chief function is to provide a reliable network interface for the NSC and to supplement network management and control functions. The task structure of the NSD-Prime is almost the same as that of the NSD. The transmit and receive policy control tasks however, are not present since traffic to and from the NSC is not labeled and hence need not be mediated. Also, some of the tasks on the NSD-Prime have been enhanced to communicate with processes residing in the NSC workstation. In the NSD-Prime, the datagram flow is as shown in Figure 10.

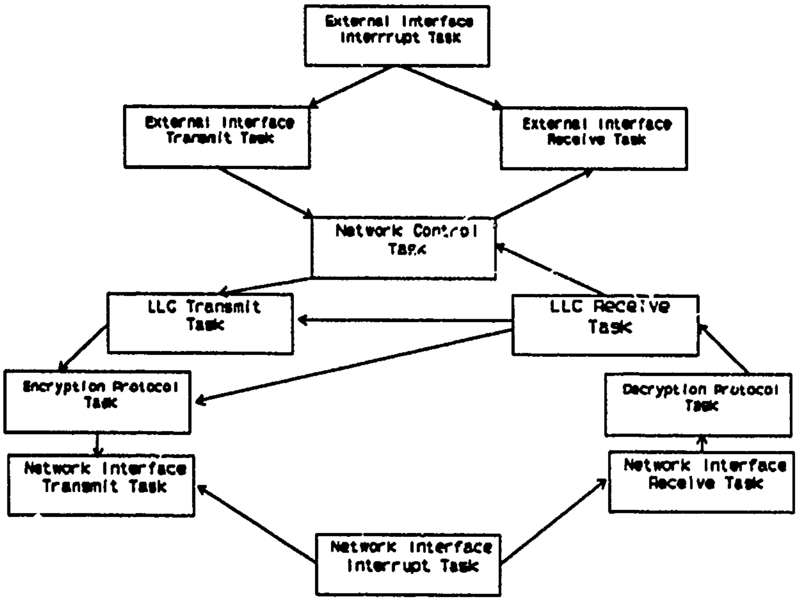
External Interface Interrrupt Task
External Interface Transmit Task
External Interface Receive Task
Network Control Task
LLC Transmit Task
LLC Receive Task
Encryption Protocol Task
Decryption Protocol Task
Network Interface Transmit Task
Network Interface Receive Task
Network Interface Interrupt Task

Figure 10 Datagram Flow in the NSD-Prime

The tasks which are different on the NSD-Prime are the authentication key interrupt task and the network control task. The authentication key interrupt task is enhanced to recognize the insertion and removal of a Datakey. The network control task is enhanced to recognize extra commands from the NSC, as well as to forward commands to other NSDs. The additional commands recognized by the NSD-Prime network control task are: set time order, write key order, and read key order. The set time order tells the NSD-Prime to set its internal clock to the time contained in the order. The write key order tells the network control task to write the data in the order to the Datakey. If a key is inserted, the data is written to it and a positive write key response is formatted and sent back to the NSC; otherwise, a write key response indicating an error condition is returned to the NSC. The read key order processing is identical to the write key order processing except that the Datakey data is read and returned in a positive response. If audit data comes in from another NSD, it is placed in a low-priority queue to be sent to the NSC. If a negative acknowledgement is received in response to an NSD-Prime datagram, a shutdown response is sent to the NSC on behalf of that NSD.

The network control task can also receive standard commands from the NSC that are addressed to both itself and remote NSDs. However, these commands are received from the NSC/NSD-Prime shared memory, instead of across the network. If the command is for a remote NSD, it is simply passed on to the LLC transmit task. Commands intended for the local network control task are acted upon in the same manner as in a standard NSD network control task.

Network Security Center Software

The NSC is a Compaq 286 or 286e dedicated to VSLAN network control. The NSD-Prime communications board resides in the NSC workstation and, for purposes of this report, is considered part of it. It provides the communications capabilities used by the NSC applications processes (NAPs) to monitor and control VSLAN operations.

The NAPs perform the actual monitor and control functions associated with VSLAN operations. They are responsible for manipulation of administrative databases, audit collection, enabling/disabling network communications, and enabling/disabling individual NSDs. The NAPs and the data structures which they manipulate reside in ring three of the Intel 80286 protection domains when they reside in RAM. In order to provide process management capabilities, the NAPs make use of the Verdix Operating System (VOS). VOS resides in the inner three rings (zero through two) of the Intel 80286 protection domains and provides an underlying multi-tasking environment, inter-process communications and device interfaces for the NSC. VOS is based on a ported version of the NSD Separation Kernel which is enhanced to interface to the NSC workstation peripherals.

Verdix Operating System

In order to ensure that the system services provided to the NSC by the workstation's underlying operating system can be trusted, Verdix has written VOS. VOS is written in PLM-286 [27] and ASM-286 [28] for use on a Compaq 286 or 286e computer. It is intended to provide both a limited multi-tasking operating system with inter-task communication facilities and an interface to the NSC workstation peripherals. VOS is not a general purpose operating system and does not support additional functionality outside of the services required to run the NSC applications processes. The kernel services are a ported version of the NSD separation kernel. A complete description of the separation kernel can be found in the NSD and NSD-Prime software section of this report (see page 21, "Separation Kernel Service Groups"). The peripheral interface services include the timer, video, printer, and keyboard services which are described below.

Peripheral Interfaces

The timer, video, printer, keyboard and disk services all reside in ring two. The primary purpose of these service groups is to provide an interface to the underlying Compaq workstation hardware in order to eliminate the need for performing direct I/O from within the applications processes.

The timer services consist of a set of routines which give the NSC the ability to set/cancel timers and to read/set the date and time. They provide support for timers used by tasks in the NSC for event timing purposes. Timer services in the NSC are based on the NSD timer services. The additional features of set/read date and set/read time are provided in the NSC in order to meet the audit requirements.

The video, printer, keyboard, and disk services provide standard interfaces to the various hardware components of the workstation. The video services are a set of routines which provide an interface to the VGA hardware. The printer services are a set of routines which provide the NSC with an interface to the audit printer. They provide processing to support writes and status polls of the audit printer. The keyboard services provide processing to support reads and status polling of the NSC keyboard. The disk services are a set of routines which provide an interface to the disk subsystem of the workstation.

NSC Data Structures

To better understand the software architecture of the NSC application processes, the reader must first understand the databases used by the NSC to perform its tasks. These databases are manipulated by the NAPs and reside in ring three when in RAM.

Configuration Database

The configuration database is a disk resident set of files that define information about the security officers, each of the NSDs, each of the principals, and security level mapping. The specific files within the configuration database include: the security officer file, NSD file, principal file, security map file, and NSD association key file.

The security officer file contains an entry for each security officer on the system. The security-relevant information contained in this file includes each security officer's name, ID, clearance level, password, and role (operator or administrator). Each field in this file is initially null. A maximum of 32 security officers can be defined.

The NSD file contains information about each of the 128 NSDs and the NSD-Prime (129 entries in all). The type of information stored for each NSD includes the NSD ID, state of activity (active or inactive), audit flags, the principals who may use it, and text fields for recording other relevant information.

The principal file contains entries for each of the 256 principals that can be configured. Each entry includes the principal's ID, clearance range, assigned NSD, next session transmit and receive key/initialization vector pairs, auditing flags, clearance range, and allowable transmit and receive windows (both levels and category set). Each field in the file is initially null.

The security map file associates a logical, human-readable, eight-character strings with each level and with each category bit.

The NSD association key file contains the cipher keys and initialization vectors necessary for NSDs to communicate with other NSDs. The file is organized into 258 key/vector pairs for each NSD (129 Transmit and 129 Receive).

Dynamic Database

The dynamic database is a memory-resident set of tables built from the corresponding Configuration Database files with the following changes:

1. The security officer table contains only the ID and password fields.

2. The NSD table contains the NSD state, audit flag, and NSD ID field with the addition of a two byte field to contain the current principal.

3. The principal table contains the principal ID, assigned NSD, and audit flags fields with the addition of total audit and number of alarms fields.

4. The security map file remains the same as its configuration database version.

Key Database

The key database consists of a key count file and a key file. The key count file contains one record specifying the number of keys remaining. The Key File contains encryption keys, decryption keys, and initialization vectors to be used by the NSC and stored on the fixed disk in the NSD association key file. These keys are transferred from a floppy disk onto the NSC's fixed disk drive.

Audit Database

The audit database consists of all audit records received by the NSC from the NSDs and NSD-Prime or generated by the NSC. NSD audit records consist of the type of auditable event, the time of occurrence, the length of audited packet, security level of the audited packet, the source and destination NSDs, and the principal of the audited packet. A complete list of auditable events can be found in the trusted facility manual [16].

System Queues

The system queues include the printer, audit, and alarm queues. The printer queue is a fixed length queue of characters to be written to the printer. It is accessible only to the security officer and audit processing task. The audit queue is a fixed length queue that stores audit data from both the NSC and NSDs before being written to the audit file. It is accessible only to the security officer and audit processing task (SOAPT). The alarm queue consists of all real time alarms sent to the NSC prior to their acknowledgement by the security officer.

NSC Application Processes

This section discusses the organization and function of the individual application processes running on thՒ NSC under VOS. All NAPs execute within ring three of the Intel 80286 protection dc..ains. There are six AP functional groups in the NSC: the security officer and audit processing task, the LAN interface transmit task, the receive message interpreter task, the system monitoring task, and the key distribution task.

Security Officer and Audit Processing Task

The security officer and audit processing task (SOAPT) provides the human-machine interface between the security officer and the VSLAN. It is responsible for translating security officer commands into inter-task communications messages and passing them on to the other tasks (such as LAN interface transmit task) and for translating inter-task communications messages. It also accepts messages from other tasks (such as receive message interpreter task) and converts them into human readable form on the NSC display or performs additional processing (i.e., write audit record to disk).

The VSLAN architecture supports separate operator and administrator roles. A person must be logged in at the NSC to assume one of these roles. The NSC software uses a login identifier and password to determine which role that person may assume. A person in the Security Administrator role may perform all of the allowable commands at the NSC while someone in the Security Operator role may perform only a subset of those commands. Only Security Administrators may execute commands which modify security officer data, set security windows and association lists, define security levels, and affect the state of LAN auditing. A more detailed description of all of the commands that can be selected at the NSC can be found in the trusted facility manual [16].

SOAPT provides the means for the security officer to: control the VSLAN operation and configuration (including programming of datakeys), review the operation of the LAN, be alerted when there are problems, and backup audit data collected by the VSLAN to a diskette. The functions provided by this task can be grouped into five categories: Security Officer, LAN Control, LAN Management, Principal Control, and NSD Management.

Security officer commands provide identification and authentication for the NSC such as login, real-time audit acknowledgement capability, and the ability to add, change, and remove security officers. Commands in this category include login and logout, alarm acknowledge and extend, and modify/list/add/remove security officer.

LAN Control commands that control actions that affect the entire VSLAN as opposed to a single NSD. Commands in this category include start/stop/suspend/resume/audit LAN.

LAN Management commands update and backup the databases used by the NSC to configure the VSLAN and set global definitions for the VSLAN such as meanings for security levels. Commands in this category include defining security levels, archiving the audit file to diskettes for post processing, backing up and restoring the NSC Databases, and loading the key file.

Principal Control commands allow the Security Administrator to configure principal specific information such as security windows and association lists. Commands that are grouped into this category include modify/list/add/remove principal, program Datakey, set/view security window, and set/view association list.

NSD Management commands modify the NSC dynamic database of NSDs and have an immediate effect on affected NSDs. Commands in this category are change cipher key, shutdown, suspend, resume, modify, view, add, and remove NSD.

SOAPT provides a means by which the NSC provides a time stamped audit trail of security relevant events, such as security officer commands, and the capability to audit all NSD transfers on an NSD to NSD basis. SOAPT places locally generated audit messages into the audit queue while the receive message interpreter task places remote (e.g., from NSDs on the network) messages into the audit queue. SOAPT is also responsible for writing records from the audit queue into the disk-resident audit file and to the audit printer.

A real-time alarm capability is also built into SOAPT. When predefined audit alarm thresholds for security-relevant events have been exceeded, an alarm entry is written to the workstation screen in addition to the standard audit records.

LAN Interface Transmit Task

The LAN interface transmit task (LITT) is a single application process that takes messages sent to it from other NSC application processes (such as SOAPT) and places them in the NSC/ NSD-Prime shared memory buffer for either transmission on the LAN or local processing on the NSD-Prime. The messages transmitted to the LAN are control messages meant to initiate status changes in the NSD as opposed to inter-host datagrams carrying user information which is simply stripped of headers and passed on to the host.

Receive Message Interpreter Task

The receive message interpreter task (RMIT) provides a means to validate NSD initialization requests and configure that NSD with its operating security windows, associations, encryption keys, and other parameters necessary for its operation. Upon reception of an initialization request, RMIT accesses the dynamic database and will send the appropriate initialization datagrams to the NSD-Prime for forwarding to the requesting NSD. RMIT is also responsible for accepting management commands from SOAPT and formatting the appropriate datagrams to carry out the command. It also passes audit datagrams and network status packets to SOAPT which allows the maintenance of a current audit log and provides the security officer with an up-to-date screen display noting the state of all NSDs.

System Monitoring Task

The system monitoring task provides a means by which the NSC polls all NSDs to determine if the NSD is still in the on-line state. To accomplish this, it sends NSD status poll orders to LITT which forwards them to the specified NSDs.

Key Distribution Task

The key distribution task distributes encryption keying material for authorized NSD to NSD associations.

VSLAN SUBJECTS AND OBJECTS

The VSLAN's subjects and objects are defined in accordance with the connection-oriented philosophy of protection as stated in the TNI.

The entire VSLAN, excluding the host/NSD shared memory, constitutes an NTCB partition. The VSLAN is assumed to be protected at the highest classification of any data that it handles. The hosts which are attached to the NSDs are external to the VSLAN NTCB partition. The VSLAN NTCB partition enforces the VSLAN security policy. The LAN cable must not be accessible by any untrusted component.

The datagrams are objects. The datagrams are exchanged between a pair of NSDs over the VSLAN. The datagrams in the form of frames are exchanged between an NSD and its associated host.

The combination of a valid Datakey and the host which is associated with the NSD that is identified by the Datakey is a subject (with a specific range of security levels which are associated with the Datakey). A host associated with an NSD cannot use the VSLAN unless a valid Datakey is inserted into the NSD. A Datakey is considered valid for the NSD if and only if it is programmed beforehand for the indicated NSD by the NSC workstation. A programmed Datakey is valid for exactly one NSD. More than one Datakey can be associated with an NSD. A valid Datakey denotes a principal in the context of the VSLAN. A valid Datakey is uniquely identified by its principal identifier. Indirectly, a host is associated with each principal identifier (i.e., each valid Datakey).

This page intentionally left blank.

SOFTWARE PROTECTION MECHANISMS

The VSLAN software protection mechanisms are discussed in this section in the following order: VSLAN startup and shutdown, identification and authentication, mandatory access control, discretionary access control, audit, and network service protection mechanisms.

Secure Initialization and Shutdown

The NSC boot-up software is resident in EPROM on the motherboard and is responsible for loading the software from the hard disk into the NSC's RAM. The software performs an integrity check on the lower 640 KB of RAM. If this test fails, an error code is generated to the console and the processor is halted. After successful completion of this check, the NSC boot-up software then initializes the NSC's hardware peripherals, and loads the NSC software into RAM from the hard disk. A checksum is completed on the NSC software and compared to a checksum written to the disk earlier during the format/installation of the system. If the checksums do not match, an error code is generated to the console and the processor is halted. If the checksums match, the software then jumps to the first instruction of the NSC software and begins execution. The system initialization task (SIT) resets the NSD-Prime hardware. SIT checks the timer in CMOS memory to determine if the date and time are valid values. If not, then the audit file is read and the current date and time are set corresponding to the last audit record in the file. The correct time and date are later prompted for by the security officer and audit processing task (SOAPT) after the security officer login. SIT invokes the system monitor task (SMT) and SOAPT and suspends itself. The system is then ready to begin operation.

The first task in the NSC software that is executed is the protected mode switch task. This task switches the processor from real to protected mode and invokes the SIT. SIT initializes the interfaces to the hardware peripherals and the operating system services. SIT configures the dynamic database in RAM by reading the data from the configuration database on the hard disk. If this data cannot be read from the hard disk, an error code is generated to the console and the processor is halted.

In order to shut down the VSLAN, a security officer must issue the "stop LAN" command. This command causes the NSC to scan the dynamic database and, for each NSD that is either on-line or suspended, issue a shutdown order. When the network control task of an NSD receives the shutdown order, it suspends processing, sends its audit data to the NSC, and then sends a shutdown response to the NSC. When the NSC has received all of the responses, it shuts the NSD-Prime down and the network is then completely shut down. The only way to bring the network back up is for the security officer to issue a "start LAN" command.

NSC Login and Logout Procedure

When no security officer is logged into the NSC, the system displays a login prompt on the screen of the workstation. In order to login, the security officer must provide his login

ID and password. If either the login ID or password is incorrect, the login attempt is audited and access is denied. If the login is successful, the security officer is granted the access privileges which have been assigned to him. The password data does not appear on the screen, either during login or modification. The identification and authentication data for the security officer is stored in the security officer tables in both the dynamic and static configuration databases. This data can be accessed and modified only by an administrator.

Principal Identification and Authentication

As part of NSD initialization, the VSLAN provides identification and authentication of principals by means of the Datakey device. The Datakey is a physical device that contains the following information:

- Principal ID (Unique for each programmed Datakey)
- NSC encryption key
- NSC encryption initialization vector
- NSC decryption key
- NSC decryption initialization vector
- Key checksum
- NSC Ethernet address

In order to enforce individual identification and authentication, the management of the VSLAN and its hosts must procedurally identify the list of users for each principal. Only the most recently programmed Datakey is valid for that principal.

If a single-user host is connected to an NSD, then only one user may be associated with each principal at any one time. If a multi-user host is connected to an NSD, then the correspondence of principals to users is one-to-many. This means the VSLAN can identify users only to the granularity of groups. Each group, in this case, consists of all users authorized to use that host while the Datakey of the principal associated with the group is inserted into the NSD. In this case, in addition to the VSLAN identification and authentication at the group level, the host must provide individual accountability for each user authorized to use that host.

The network control task on the NSD recognizes that the Datakey is present and a principal is requesting access to the VSLAN. First, it verifies the data on the Datakey using the key checksum. If the checksum is incorrect, the NSD records an error by setting the Datakey flag, and incrementing the counter which indicates the number of faulty Datakeys inserted since the last valid initialization of that NSD. The NSD then records the time of occurrence and waits for a valid Datakey. If the data on the Datakey are correct, the NSD's network control task then writes the information necessary for communication with the NSC into the control tables on the NSD. This information includes: the principal's ID, the keys and initialization vectors used for control datagrams, and the NSC address. Finally, the NSD creates an NSD initialization request containing the principal's ID and the network address of the NSD.

The data in the NSD initialization request, with the exception of the NSC address and the request packet's checksum, is then encrypted using the NSC encryption key and is sent to the NSC as a control packet. Next, the Datakey is erased which prevents its reuse until it can be reprogrammed using new data provided by the NSC. An LED on the key socket indicates whether the NSD is in the process of initialization in order to warn the principal not to remove the Datakey during this process. Should the user ignore this warning and remove the Datakey prematurely, the Datakey will be useless, since it will not have the new initialization data from the NSC and will have to be reprogrammed by the security administrator.

When the NSC receives the NSD initialization request, it searches the principal table in its dynamic database and determines if the principal is authorized to use the requested NSD. If not, the NSC generates an "invalid initialization request" audit record and will not allow the principal to initialize any NSD until the Datakey is reprogrammed. If the principal is permitted to use the NSD, the NSC issues an initialization order, a key distribution order, and an on-line order to the NSD. The initialization order contains the principal's transmit and receive security window, the principal's transmit and receive association lists, and the master keys and initialization vectors with which the principal will be authenticated at his next login to the VSLAN. The key distribution order contains the master keys and initialization vectors for each NSD to which the NSD has a transmit or receive association.

When the NSD receives the orders described above, it can complete initialization. Upon receipt of the initialization order, the network control task initializes its tables with the security window and NSD association information and also writes the next session master keys and initialization vectors to the Datakey. Upon receipt of the key distribution order, the network control task updates its internal encryption and decryption control tables with the keys and initialization vectors contained in the datagram. When the on-line order is received, the network control task issues resume messages to the external interface transmit task and the network interface receive task and then sends an on-line response to the NSC. At this point, the communications paths between the NSD and both the NSC and other NSDs are established. The host associated with the principal can now make use of the VSLAN to communicate with other hosts connected to the VSLAN.

The NSC will regularly poll each NSD to ensure that it is present on the network. The NSD then responds by sending a status poll order datagram back to the NSC, and resets the audit exception counter, the audit exception threshold, and the polling countdown timer. If a given NSD is not polled by the NSC within the time interval, the NSD performs halt processing via a call to the separation kernel. If an NSD does not respond to the NSC's status poll, the NSC assumes that NSD has gone off-line and marks it as such.

Mandatory Access Control

The VSLAN MAC policy controls access between subjects and objects of the VSLAN. Once a Datakey has been inserted for an NSD and the security windows have been downloaded from the NSC to the NSD, mandatory access control is enforced by the NSD for the datagrams that the host exchanges with the NSD. The security window information is also made available to the host in the ICB.

To initiate a transmission of a datagram, the host places the datagram into the host/NSD shared memory and signals the NSD. The NSD copies the datagram from the NSD/host shared memory into the NSD's local memory buffer and checks that the host supplied sensitivity label is within the current allowable send window. If it is not, an audit record is generated and the NSD disallows the transmission. If the datagram is within the allowable send window and if the discretionary access control check is satisfied, then the datagram is forv arded to the receiving NSD.

Similarly, the host can receive a datagram after the reception is signaled by the NSD to the host. Prior to this signalling the NSD checks that the sensitivity label associated with the datagram is within the current allowable receive window, checks the discretionary access control lists, and places the datagram in the host/NSD shared memory in a frame indicated by the host. If any checks fail, then the datagram is not placed in shared memory, and the event is audited.

The NSD makes the checks not against the send/receive window information residing in the ICB, but against the information residing within the NSD's internal database.
The transmit and receive security windows of a given principal can be reset by the security officer. This cannot happen while the associated NSD is on-line.

Discretionary Access Control

The VSLAN DAC policy controls access between subjects and datagrams by means of the transmit association list and receive association list.

A transmit association list and a receive association list are associated with each principal. When a principal becomes active (i.e., the corresponding NSD is initialized), the association lists are made known to the NSD by the NSC. The association lists are also made available to the host in the ICB.

The transmit association list denotes a list of NSDs to which the principal is allowed to transmit datagrams. The receive association list denotes a list of NSDs from which the principal is allowed to receive datagrams.

When a subject transmits a datagram, the NSD copies the datagram from the shared memory and checks the transmit association list to ensure that the subject may transmit the datagram to the destination NSD indicated in the datagram. Before a subject receives a datagram, the NSD checks the receive association list to ensure that the subject may receive the datagram from the source NSD indicated in the datagram and copies the datagram into the host/NSD shared memory. If the transmit or receive check fails, the NSD generates an audit record and discards the datagram. In the VSLAN, these checks represent the discretionary access control checks made between the subject and datagrams.

The transmit and receive association lists of a given principal can be reset by the administrator, provided that the NSD is not in a suspended state. The host cannot modify the association lists.

Audit

There are two levels of audit: security-relevant auditing and statistical auditing. The security-relevant auditing comprises the auditing of all security officer actions and the following security-relevant events: attempted MAC violations, attempted DAC violations, packet integrity errors, illegal initialization attempts, sequence number errors and faulty Datakey error. The statistical auditing allows the security officer to perform a selective audit on one or more specific NSDs. When statistical auditing is enabled for an NSD, the NSD generates an audit event for each datagram that it processes.

The NSC system monitoring task periodically polls each NSD by writing a status poll order control packet to the NSD-Prime which distributes it to the NSDs. After the NSD network control task receives this poll form the NSD-Prime, the NSD makes an LLC-level acknowledgment to the NSD-Prime. The NSD then resets the security audit event counter and updates the audit event suspend ceiling (which is the threshold of security-relevant audits beyond which the NSD will stop processing host datagrams) with the new value found in the status poll order. Finally, the NSD network control task restarts its status poll timeout timer. If this timer expires, the NSD network control task shuts the NSD board down on the assumption that the NSC is down. In this case, the Datakey must be removed and reinserted to restart the NSD.

In the event that an NSD does not receive a status poll order message that has already been transmitted by the NSD-Prime, the LLC protocol between the NSD-Prime and the NSD will timeout. This will cause the NSD-Prime to send a proxy shutdown response to the NSC for the NSD that did not respond.

Data interchange between the NSD and its host is blocked when the NSD sends an audit message to the NSC until it receives an acknowledgement from the NSC. When the NSC receives an audit message from an NSD, the receive message interpreter task (RMIT) places the message into the audit queue so that the message can be written to the audit database by the SOAPT. After this, the RMIT sends an audit response order message to the NSD. SOAPT runs at the lowest priority. The audit queue can hold up to 4700 audit records. In the unlikely event that the audit queue gets full, RMIT will not be able to send an audit response order message to the NSD, and the NSD will not resume processing of its host data traffic.

In the NSC there is an indicator on the display which tells the security officer how full the audit file has become. Should the file become 90 percent full, then the NSC forces the VSLAN into a suspended state (i.e., inactive). The system will stay suspended until the audit file has been archived onto floppy disks and the disk space released.

SOAPT is responsible for the real time alarm capability of the NSC. It continually reads the audit file, searching for security-relevant audits from an NSD. If it finds one, it is placed on the printer queue. Based on the audit type (e.g., MAC, DAC, etc.), it increments the principal's count of the number of audits. It then checks the audit type against the threshold setup by the security officer. If the counter is greater than the threshold, then SOAPT will format an alarm data holder and place it in the alarm queue to be viewed by the security officer. Alarms from the alarm queue are displayed on the NSC screen. Each alarm must be acknowledged by the security officer.

Audit post-processing begins when the security officer backs-up and clears the audit file onto floppy disks. He must then take the diskettes to a PC/AT-style machine running a copy of MS-DOS supplied by Verdix for post-processing of audit data[1].

Verdix licenses a generic copy of MS-DOS from Microsoft that contains the minimum programs necessary to operate a PC. This is similar to the copies that are licensed to IBM, Zenith, and other PC manufacturers that bundle MS-DOS with their hardware. The vendors have the opportunity to provide additional functionality with these versions of MS-DOS. In Verdix's case, they remove all unnecessary programs provided by Microsoft, as opposed to adding functionality, leaving only the routines required by their post-processing tool.

The Verdix supplied post-processing tool is able to choose records on the basis of security-relevant audits, statistical audits, state change audits, security officer commands, faulty Datakey audits, and all audits. These may be broken down to a finer granularity.

For security-relevant audits and statistical audits, the security officer can specify the following: a list of reporting principal IDs, a list of originating principal IDs, a list of source NSDIDs, a list of destination NSD IDs, the range of sensitivity levels, user ID, and time period. In addition, for security-relevant audits, any combination of the following is specified: MAC violations, DAC violations, packet integrity errors, sequence number errors, and illegal initialization attempts.

For state change audits, the security officer specifies a list of NSD IDs, and any combination of the following: on-line, shutdown, suspend, resume, add association, remove association, and key distribution. Security officer audits may be selected by security officer ID. The selection of all audits creates a report including all the above types of audits. Each audit type is reported completely before the next type is displayed. Therefore, all security-relevant audits are displayed, then statistical audits, state change audits, and finally security officer commands.

Network Service Protection Mechanisms

The protocols used in the VSLAN provide several of the network security services described in Part II of the TNI. These services are performed within the VSLAN's NTCB partition (i.e., services related to communications between each NSD and the NSC, and

[1] A Verdix version of MS-DOS-based software is supplied and maintained under configuration control in accordance with the Verdix configuration management plan [2].

among the NSDs).

The protocols supported by the VSLAN are the encryption protocol (EP), IEEE 802.3 Media Access Control Protocol (MACP), modified IEEE 802.2 logical link control protocol (LLCP), and network control protocol (NCP). Taken as a whole, the protocols provide communications field integrity, protocol-based denial of service protection, and denial of service protection for the management of VSLAN network.

The EP and MACP provide communications field integrity. Communications field integrity is provided both by the encryption of the EP and by the use of cyclic redundancy checksums supported in the MACP. The NSD software places a 16-bit checksum on the datagram. The datagram also contains a 16-bit header checksum on the sequence number, source and destination NSDs, source principal ID, service access point, and length of data field. The sensitivity label is protected with an additional 16-bit checksum. Cyclic redundancy checksums are calculated over the entire datagram (which includes the security label, the header, and the data portion of the packet). Because of these mechanisms, the VSLAN can detect the modification, insertion, deletion, or replay of a packet. Since these protocols are used to provide datagram service, no recovery is attempted. That is, when a packet is noted as having been modified during transmission, no retransmission or error correction is attempted.

The LLCP is used for communications between each NSD and the NSC. The LLCP provides guaranteed delivery of network control packets via the acknowledgements of received packets and packet retransmission when necessary. The LLCP, when combined with the EP and MACP, provides a reliable communications service. For all network control-related communications, the VSLAN can attempt recovery by requesting retransmission of a packet that has been corrupted.

The NCP provides protocol-based denial of service protection and network management denial of service protection. The NCP is used for control-related communications between the NSDs and NSC via the NSD-Prime. A status poll order is generated periodically by the NSC and sent to each NSD. If an NSD does not respond within a certain period, the LLCP software times out and the NSC is notified. Similarly, if an NSD does not receive a status poll order within a certain period, the NSD suspends operations. This protocol-based denial of service protection enables the security officer to be made aware of actual or potential problems and to initiate appropriate actions. If the NSD audit task logs too many events between status poll orders, the NSD network control task suspends the NSD. This has the effect of suspending a node that may be creating problems on the network.

This page intentionally left blank.

ASSURANCES

This section outlines various additional assurances that Verdix has implemented on the VSLAN. These include a formal security policy model, a covert channel analysis, system integrity mechanisms, and configuration management.

Formal Security Policy Model

The Formal Security Policy Model (FSPM) provided by Verdix for the VSLAN is an access control model, which elaborates the connection-oriented abstraction as set out in the TNI. It is an external model that covers the VSLAN communications phenomena and security requirements imposed on them.

Verdix has not provided internal security models for the NSDs or NSC as separate individual components of the VSLAN. Such internal models need not be provided. When considering each NSD or the NSC separately, we note that all the subjects (processes) within each NSD or the NSC is trusted. They are within the VSLAN NTCB partition. To have a useful internal model for an individual component, there would have to be some untrusted subjects within the component whose access to the resources of the component is controlled by the NTCB of the component. In contrast, the external model that is provided covers the entire VSLAN and the subjects external to the VSLAN, and the control of access to its resources by these external subjects.

The external FSPM for the VSLAN is formally based on the Bell-La Padula (BLP) model, though it is not truly a BLP model. It includes modified forms of the simple security property, the *-property, and the discretionary property as well as rules of operation. The security model for the VSLAN is necessarily a modification of the original BLP model, since it is treating LAN phenomena and security requirements and not the phenomena and security requirements usually associated with conventional stand-alone computer systems. In particular, the subjects and objects are different in the two cases.

In the formalism of the security model, the subjects are equated with the hosts of the VSLAN. More precisely, the subjects should be considered as pairs, each consisting of a host and a valid Datakey. In the model documentation the pairs are briefly called "hosts". From the perspective of the VSLAN, the hosts are the active entities, hence, subjects, that are trying to communicate with one another using datagrams. The hosts are often multilevel entities and effectively need to be trusted to send datagrams to the VSLAN in accordance with well-defined security policies for them. The objects are equated with the datagrams. From the perspective of the VSLAN, the datagrams are the passive entities used by the hosts for communication vehicles. The datagrams are transient entities, unlike the objects usually considered in stand alone computer systems. Elaborations of the definitions of the concepts of subject and object as they pertain to the VSLAN are given elsewhere (see page 37, "VSLAN Subjects and Objects"). The primary security modeling aim is exactly to define the control of subject-to-subject (i.e., host-to-host) communication with the datagrams providing the communication vehicles. Hence, the VSLAN FSPM is an external communications-oriented security model. The VSLAN is modeled as a state machine. The network state is defined to incorporate the current discretionary and mandatory access control information for the hosts and information about the security levels of the datagrams, and the rules are laid down to provide for only those state changes that are secure.

The VSLAN FSPM is represented in the Gypsy specification language [33]. The FSPM consists of four Gypsy scopes. The Network_Global_Definitions scope contains the global type and function declarations used by the other three scopes. In particular, it defines the network state and the form of the state-changing rules and their execution. The Network_Security_Policy scope describes the rest of the framework for the VSLAN and asserts the basic security theorem and related lemmas. The Network_Top_Level_Specification scope defines the VSLAN initial state and rules of operation. The Network_FTLS_Model_Proof scope provides the lemmas for the detailed steps to prove that the rules of operation result in a secure system.

The principal security properties given in the VSLAN FSPM are derived from the VSLAN security policy (see page 5, "Security Policy"). Formally the properties are presented as modifications of the properties of BLP. The BLP terminology is carried over to the FSPM. Thus we have the simple security property, the *-property, and the discretionary property in the following forms:

- The simple security property holds for a state if and only if, for every object being read by a subject, i.e., every datagram being received by a host from the VSLAN, the maximum receive security level of the host dominates the security level of the datagram.

- The *-property holds for a state if and only if, for every object being written by a subject (i.e., every datagram being sent by a host to the VSLAN) the security level of the datagram dominates the minimum transmit security level of the host.

- The discretionary property holds for a state if and only if, for every object being read by a subject, i.e., for every datagram being received by a host from the VSLAN, the receiving host is authorized to receive from the sending host by a receive association; and, for every object being written by a subject, i.e., for every datagram being transmitted by a host to the VSLAN, the sending host is authorized to send to the receiving host by a transmit association.

In addition, the VSLAN FSPM includes a tranquility property that states that the security level of each object (datagram) does not change during its existence.

To model the operation and implemented security policy of the VSLAN a set of 12 rules of operation have been given in the FSPM. These fall into three groups of four rules each. For writing, i.e., transmitting, a datagram by a host to the VSLAN, we have the following four rules, which are to be executed in sequence:

- create_comm_rule. This rule models the creation of a datagram (communication) at a host for transmission to the VSLAN.

- get_comm_write_rule. This rule models the checking of the transmit security window, i.e., the mandatory access control transmission mechanism, and the transmit associations, i.e., the discretionary access control transmission mechanism, to determine if the datagram can be transmitted securely. Thus the datagram for transmission from a host is checked to determine whether its security level is dominated by the maximum security level and dominates the minimum security level of the transmit window and whether there is a transmit association from the host to the destination.

- write_comm_rule. This rule models the transmission of the datagram to the VSLAN. (It is actually a no-op in the model.)

- rescind_comm_write_rule. This rule models the removal of permission to transmit another datagram under the current access checks. Thus only one datagram is to be transmitted under the given access checks.

For reading (i.e., receiving) a datagram by a host from the VSLAN, we have the following four rules, which are to be executed in sequence:

- get_comm_read_rule. This rule models the checking of the receive security window, i.e., the mandatory access control reception mechanism, and the receive associations, i.e., the discretionary access control reception mechanism, to determine if the datagram can be received securely. Thus the datagram for reception by a host is checked to determine whether its security level is dominated by the maximum security level and dominates the minimum security level of the receive window and whether there is a receive association from the source to the host.

- read_comm_rule. This rule models the reception of the datagram from the VSLAN. (It is actually a no-op in the model.)

- rescind_comm_read_rule. This rule models the removal of permission to receive another datagram under the current access checks. Thus only one datagram is to be received under the given access checks.

- delete_comm_rule. This rule models the deletion of a datagram from the VSLAN.

For changes by the Security Administrator to the VSLAN discretionary security policy, we have the following four rules, which can be executed individually and in any order:

- add_auth_send_to_rule. This rule models the addition of a transmit association, i.e., the modification of part of the discretionary access control transmission mechanism, for a host by the security administrator.

- remove_auth_send_to_rule. This rule models the removal of a transmit association, i.e., the modification of part of the discretionary access control transmission mechanism, for a host by the security administrator.

- add_auth_receive_from_rule. This rule models the addition of a receive association, i.e., the modification of part of the discretionary access control transmission mechanism, for a host by the security administrator.

- remove_auth_receive_from_rule. This rule models the removal of a receive association, i.e., the modification of part of the discretionary access control transmission mechanism, for a host by the security administrator.

It is necessary to provide an argument why the VSLAN FSPM is adequate for guaranteeing the security of the VSLAN (or similar systems). This argument is given informally. The guarantee of security of the model rests on the main security properties. The *-property requires that no datagram is transmitted by a host to the VSLAN with a security level that fails to dominate the minimum transmit security level of the sending host, i.e., the host cannot write down. The simple security property requires that no datagram is received by a host from the VSLAN with a security level that fails to be dominated by the maximum receive security level of the receiving host, i.e., the host cannot read up. In addition, the discretionary security property requires that the transmission of datagrams and the reception of datagrams are in accordance with the permitted discretionary associations. Also the datagrams cannot change security levels within the VSLAN according to tranquility property. Assuming the security levels of datagrams are given accurately by the hosts, then it is not possible for one host to transmit information covertly to another according to the FSPM. No datagram will be delivered to a host without passing all the access control checks. If according to the FSPM a datagram is not delivered, no information is given to the intended receiving host. Hence, in the model no information is given covertly.

The inductive demonstration of the security of the VSLAN system consists of proving that the initial state is secure and that any state change leads from secure state to secure state. Hence, at any time the system is always in a secure state. This demonstration is reduced to proving that the rules of operation satisfy the stated security and tranquility properties. These proofs have been carried out with the help of the theorem prover of the Gypsy Verification Environment.

A correspondence between the implemented VSLAN interface and the VSLAN FSPM has been developed. This shows that the FSPM provides a good modeling of the security policy implemented in the system.

Covert Channel Analysis

Verdix has selected Kemmerer's Shared Resource Matrix (SRM) methodology [30] to perform the VSLAN covert storage channel analysis. The covert channel analysis of the VSLAN was performed on the basis of the VSLAN DTLS, other Verdix design documentation, and knowledge of the VSLAN implementation. The analysis contains a detailed argument of the completeness of the shared resource matrix used. This argument was based on a functional decomposition of VSLAN resources.

The SRM methodology is based on the fact that there is some set of resource attributes subject to modification and observation and some set of operations that can be performed by the system subjects resulting in the modification and observation of those resource attributes. VSLAN subjects are represented by hosts that are connected to the VSLAN, while the shared resource attributes are those entities within the VSLAN that can be manipulated by one or more hosts in such a way as to be detected by one or more different hosts.

At a primitive level, the following operations can be performed by a host: I/O port read, I/O port write, shared memory read, shared memory write. The I/O port operations are used to cause or reset interrupts. The shared memory read is of relevance only in the case of fields which the NSD writes, and the shared memory write is only relevant if it influences the NSD and (by virtue of the discussion below about the network environment) only if it does so in such a way as to affect packets which the NSD will later put onto the LAN.

It should be noted that the model (described in the preceding subsection) represents the reading of a datagram by a sequence of four operations, and the writing of one similarly. These operations cannot be carried out separately in the particular implementation of the model found in the VSLAN. It is therefore not necessary to consider them individually for the purposes of the covert channel analysis; instead the composite operations of reading and writing a datagram (called Send Datagram and Receive Datagram) are considered. The remaining operations in the model describe actions by the Security Administrator, and are therefore not relevant to the covert channel analysis.

The SRM methodology had to be adapted for the network environment. Each host can both refer to and modify the shared memory of its own NSD, and so the composite resource consisting of host/NSD shared memory, or any particular field of host/NSD shared memory, can be modified and referred to by all hosts.

However, this alone cannot result in communication between the hosts. Thus the notion of "shared resource" was, in the case of NSD resources, limited to resources which can be referred to or modified by a distant host as well as by the local host.

The result was that three operations were identified: Send Datagram, Receive Datagram, and Initialize Interface (consisting of shared memory reads to determine security windows and association lists sent from the NSC, and shared memory writes to set up the transmit and receive frame lists and buffers), which were found not to affect any shared resources.

Several of the covert channels depend on the host detecting that its NSD has become suspended, which can be deduced from the failure of the receive datagram operation; however the status of the NSD can be more easily detected by the host by reading the NSD status field in the Interface Control Block in the shared memory. All other shared memory fields which are written by the NSD are written upon receipt of a datagram, upon sending a datagram, upon initialization, or as a result of actions by the security officer at the NSC. Thus it is adequate to characterize the range of relevant host operations as the three used in the analysis.

The VSLAN covert channel analysis discovered four potential covert storage channels. Engineering estimates have been made of the maximum band widths of each of them. The largest of these band widths is estimated to be only 7.1 bps and only in the extreme case of the sender (or several collaborating senders) collaborating with several other of the NSDs attached to the VSLAN which act as receivers. Furthermore, this channel requires the involvement of the human principal of each receiving NSD, since he must reset the NSD (by removing and reinserting the Datakey) whenever the NSD becomes suspended. Thus, the information cannot be received by malicious host software alone. At most 9 bits can be transmitted for each occasion that a human principal takes this action. This maximum is achieved when only one NSD is receiving; the figure is less if the channel is using multiple receivers. The channel is audited, with at least one audited event per bit signalled.

The estimated band widths of the remaining potential covert storage channels are less than 0.01 bit per second. The guidelines used in governing covert storage channel band width during this evaluation are the following:

> 100 bits/sec shall not exist

10 - 100 bits/sec shall be documented and shall have all their real or potential uses audited

1 - 10 bits/sec shall be documented and may be audited

< 1 bit/sec may be documented, may be audited.

As part of this analysis, Verdix divided covert channels into two distinct classes: covert storage channels and covert timing channels. Covert storage channels involve the direct or indirect writing of a storage location by one process and the direct or indirect reading of the location by another process. A covert timing channel is a covert channel in which one process signals information by modulating its use of system resources so as to affect the response time observed by the second process. (The sending process's actions necessarily cause some change to the state of the system, but such change cannot be observed by the receiving process other than through response time). Analysis of covert timing channels is not required for a B2 network component.

Verdix's Shared Resource Matrix analysis identified resources which can be observed by hosts independent of class of covert channel (i.e., storage vs. timing). For example, packets sent by a remote host can occupy the LAN cable or cause the CPU of the local NSD to become busy allowing a local host to detect this by a delay in response time, thus creating a covert timing channel. Therefore, while covert timing channels can be identified using the Shared Resource Matrix methodology, band width analysis was not required. Verdix has documented five potential covert timing channels. Three of these channels are heavily audited, with many audits per bit sent (though in some scenarios these could be statistical audits). One is not a VSLAN covert channel but a network system covert channel, since it requires certain actions by a multilevel host which re.ult in the downgrading of information within that host's range. The other relies on collision of packets on the LAN cable.

It is important to note that the VSLAN is a network component provi'ing information flow between hosts (not host processes or users), and that the Verdix covert channel analysis is concerned only with the identification and analysis of unauthorized information flow between pairs of host systems. When the VSLAN is included as part of a network system, new covert channels may be introduced that were not evident from the evaluation of the VSLAN NTCB partition alone. It is beyond the scope of the VSLAN covert channel analysis to address covert channels that occur at the network system level (e.g., between two processes in different hosts or the same host, where the hosts are authorized to pass information, but the processes are not). Thus when multi-level hosts are connected to the VSLAN, the interface software on the host must be able to deliver the labeled packets to the appropriate processes running on the host system. In the case of single-level hosts, no assumptions are made about the correct operation of the interface software.

A network system (which may include evaluated network components) must always be evaluated as a whole to ensure that the components together enforce the overall policy. Although the VSLAN covert channel analysis cannot address information flow between host processes, the results of this type of analysis (i.e., of host-to-host channels) can form part of an analysis of covert channels in a network system.

System Integrity

The integrity of the hardware and firmware associated with the NSC, NSD-Prime, and NSDs is checked throughout the operation of the VSLAN.

When bringing up the NSC the workstation bootstrap task checks the integrity of the 640 KB of RAM of the Compaq 286 or the lower 640 KB of RAM of the Compaq 286e an' initializes and tests various hardware peripherals. Provided these tests are successfully completed, the NSC software is loaded from the hard disk and a checksum test is performed on it. The process then switches the processor from real to protected mode and a jump is made to the system initialization task (SIT) of the NSC software. This task performs diagnostics on the hardware peripherals of the NSC workstation. It initializes various hardware peripherals and checks the timer chip.

There is a system boot task for initializing each of the NSDs and NSD-Prime. It performs a checksum integrity check on EPROM to verify that the EPROM-based code for each of the NSDs and NSD-Prime is not corrupted. Provided this test succeeds, memory diagnostics are carried out and appropriate segments are copied from EPROM to RAM. The task switches the processor from real to protected mode and invokes the system initialization task. The system initialization task controls overall initialization of hardware and software, and the execution of the diagnostics tasks. This task invokes each of the following diagnostic tasks:

- The memory diagnostics task checks all memory that was not checked by the system boot task for correct functioning.
- The ciphering diagnostics task tests the cipher block chaining mode of the ciphering device (i.e., AM 9568 chip) for correct operation.

- The network interface diagnostics task tests the Intel 82586 802.3 device for correct operation. The functionality of the 82586 network controller is tested by operating the network interface in internal loopback mode (within the chip).

- The timer device diagnostics task checks the AMD 9513 timer device for correct operation. The timer device diagnostics processing requires extensive use of the 8259 Programmable Interrupt Controller, and so the timer device diagnostics task also supports interrupt device diagnostics.

The system monitoring task (SMT) of the NSC periodically polls the on-line NSDs via the NSD-Prime to determine their status, i.e, to see if they are active (see page 40, "Principal Identification and Authentication" and page 43, "Audit").

By localizing failures to single nodes, the VSLAN is more likely to maintain secure operation. The only network-wide failure that can occur is when the NSC workstation has a failure. Because the NSC workstation processes audit information and supports the security policy, the network cannot operate properly if there is such a failure.

Configuration Management

Verdix has implemented a configuration management plan which maintains control of changes to a well defined set of configuration items. All configuration items fall into one of five groups: Project Management Plans and Reports (indicated by the prefix PR in the list of references at the end of this report), Specifications and Design Documentation (SP), Technical Reports (TR), Hardware Products, and Software Products. Each individual item within each category is assigned a unique serial number that identifies it as part of the evaluation project.

Verdix has implemented a Configuration Control system that tracks all changes to any of the configuration items. For each of the configuration items, a baseline is established and recorded. Thereafter, a Configuration Control Board must approve all changes based on submission of a Problem Report. Once the change is analyzed and approved, an engineer is assigned to make the necessary changes in the appropriate configuration item. The revised version is then passed to the CM administrator and entered into the configuration management files using automated tools. The automated tools include the Polytron Version Control System (PVCS) used on a PC/AT local area network with a central file server running the LOCUS product, and the Revision Control System (RCS) used on a UNIX system (4.2 BSD). The latter is used to track changes to the documentation, and the former to track changes to the hardware and software design and implementation.

In order to ensure a consistent mapping among all NTCB code and documentation, the configuration management plan includes periodic design and code reviews, and configuration testing to assure that changed versions work as required. Design review consists of comparing each Program Design Language (PDL) module with its corresponding functional specification and task structure. The reviewers will seek to ensure that the PDL and its specifications match, and that the PDL continues to meet the project's programming standards. Code review consists of comparing the PDL of each module with its corresponding implementation in the source code. Here also, the reviewers will seek to maintain correspondence between the two, and to maintain the project's programming standards on the source code.

The tools used to generate new versions of the NTCB from source code are the Intel PLM286 and PLM86 compilers, Intel ASM286 and ASM86 assemblers, Microsoft MASM 5.0 assembler, Microsoft LINK, Intel Bind286 and Build286 utilities, and the Real Time Computer Science Corporation's Universal Development Interface (which enables running the Intel tools on a DOS-based PC). The PVCS utilities enable the comparison of new and old versions of the source code, in order to ascertain that only intended changes have been made to the NTCB.

This page intentionally left blank.

SECURITY TESTING

Vendor Testing

The Verdix test plan includes 25 test scenarios, divided into three categories: VSLAN security policy, accountability, and operational assurance. Security policy testing includes eight scenarios directed at testing the mandatory and discretionary access controls of the VSLAN, as well as label integrity and object reuse. Accountability testing includes eight scenarios aimed at exercising the audit functionality of the VSLAN, including auditing of the security officer logins, auditing of access control violations, statistical auditing, and packet integrity auditing. Operational assurance testing includes nine scenarios directed at testing a variety of network assurances, such as security officer commands, audit integrity, network control under loaded conditions, and control of memory shared with attached hosts. In particular, one of these scenarios measures certain VSLAN parameters used in the engineering analysis of the estimated band widths of the covert storage channels.

The tests described in the test plan are complete relative to the security requirements levied upon the VSLAN. Each Verdix test scenario includes multiple annotated test_scripts, the external support required for the test configuration, inputs, expected outputs, and the criteria for a successful test. Testing has been carried out by Verdix according to the test plan and all tests were completed successfully as reported in the test results documentation [20].

Since the VSLAN is being evaluated as a datagram-level network component, testing requires additional hardware in addition to the traditional software support. This is because the test software must execute on host systems that are connected to the VSLAN, rather than executing directly on the VSLAN.

The following additional hardware support items were chosen by Verdix to test the VSLAN product: a VSLAN packet analyzer, a multi-node simulator, attached hosts connected to their corresponding NSDs, and a PC AT for the post processing of audit data. The VSLAN Packet Analyzer is a special device that monitors packet activity on the network transmission medium by transmitting, capturing, storing, and analyzing packets. The hardware, operation, and command structure of the Packet Analyzer are documented in the Verdix test plan [3]. The multi-node simulator is used to test the VSLAN, specifically the NSC and NSD-Prime, under heavy loading conditions. Each simulator can simulate the network control actions of up to 16 different NSDs. They are capable of generating large volumes of audit messages directed to the NSC. The Verdix test plan includes a description of the simulator and its operation. Depending on the type of NSD board being tested, the attached host may be either an IBM (or compatible) PC-AT or PC-XT workstation, a Multi-bus single-board computer residing in a Multi-bus chassis, a DEC MicroVAX, a Sun Workstation, an AT&T 3B2, or an Apple Macintosh II.

Team Testing

Using the Verdix test plan as a basis, the team took the following approach to the security testing of the Verdix VSLAN. First, it carefully examined the test procedures devised by Verdix and the test results already obtained by Verdix. In particular, it considered the Verdix tests with reference to the TNI requirements. The Verdix Test Plan provided evidence that Verdix had systematically developed testing procedures with respect to the TNI requirements, and the Verdix test results showed that these procedures had been executed according to the plan. Hence, during security testing, the team primarily sampled the Verdix tests, deriving further results, to make sure that the Verdix test results were accurate. These further results confirmed the ones Verdix had derived. Second, the team developed some additional tests, which were mainly concerned with testing the VSLAN under loaded conditions. The full capability of 128 NSDs was tested using simulators. The audit capability was tested to capacity by filling up the audit storage area, in order to determine if audit data could be lost. Also performance was observed in a more realistic environment using the TCP/IP protocols and a connection to a Unix machine. In addition, the team tested the VSLAN under some extreme conditions. The Verdix audit post-processing tool was also examined by the team to determine its adequacy and correctness.

Test Configuration

Initial testing was conducted on the vendor's premises over a period of one week. A follow-on test to verify that problems found by the team had been corrected took one day. The vendor supplied the following systems for the test configuration:

- One NSC (a Compaq 286 and 286e)

- Ten IBM PC ATs, running PC-DOS, Versions 3.0, 3.1 and 3.3 (IBM PC bus)

- One IBM PC XT, running PC-DOS Version 3.1 (IBM PC bus)

- One Macintosh II, running A/UX Version 1.1.1 (NuBus)

- One AT&T 3B2 400, running System V Version 3.1 and MLS Version 1.1 (3B2 bus)

- One Sun 3/260, running SunOS MLS 1.0, beta release (VME bus)

- One Multibus chassis with a single-board computer from Zenix, running a ROM-based, Verdix-supplied test program (Multibus-I)

- One DEC MicroVAX II, running VMS Version 5.0 (DEC Q22 bus)

Thus, the test configuration included at least one of each type of bus supported by the VSLAN. Each of the systems (other than the NSC) contained an NSD connected to its bus. Several additional NSDs for the IBM PC-bus were available to facilitate testing.

In addition to the above equipment, Verdix supplied several EPROMs for the NSD for conversion of an NSD into a packet analyzer, as well as EPROMs for conversion of an NSD to a simulator (see page 58, "Team Testing")[1]. Finally, Verdix supplied test driver programs for each system, programs to drive the packet analyzer and simulator, and a copy of the audit post-processing program. The team was thus able to configure the system so as to be able to conduct any of the tests in the vendor's test plan as well as its own tests.

Team Test Results

The team tests were based on conducting tests that were not performed by Verdix, on performing a load stress on the system, and on the rerun of selective scripts from each test scenario stated in the Verdix test report.

Two tests were found that were not addressed in the Verdix test report[2]. The first of these tests addresses the filling of the audit storage area to capacity to try to lose audit data. When the audit storage area is 90% full the VSLAN is suspended as specified. The second test addresses the functional correctness of the VSLAN audit post processing tool.

Based on all the tests conducted, the VSLAN was found to operate according to the security specification. A few minor errors were discovered. Verdix has fixed these errors satisfactorily.

[1] Note that the Packard Analyzer is used not only to receive, decrypt, monitor, and display selected LAN packets, but also toread, copy, and even corrupt Datakeys. It can monitor selected types of transactions between selected NSDs, and can track key-change orders sent from the NSC to the NSDs. The simulator is able to simulate up to 16 different host-NSD pairs, and to selectively generate datagrams.

[2] In addition to these two security-related test, a file transfer test was conducted to obtain a feel for VSLAN performance under real operating conditions. The file transfer used the TCP/IP protocols between UNIX-based operating systems (a SUN Microsystems' machine and a SEQUENT machine) over a LAN with and without the VSLAN. The relative performance degradation was found to be approximately 7%. Verdix indicated that if only the data link layer protocol is used, the performance degradation due to using the VSLAN is approximately 20%.

Penetration Testing

The penetration testing of the VSLAN followed the security testing. The team used the method of flaw hypothesis generation [31]. Some of these were related to flaws that had actually been discovered during the team's security testing. (These were flaws which could not be exploited by users.)

The number of possible flaw hypotheses was limited by the fact that the user's interaction with the NTCB is limited to attempting to send or receive a datagram, and to inserting and removing a Datakey from the Keyceptacle.

The team tested for most of the flaws hypothesized; a few were found to be impossible or impractical. Those tested for were found not to be present. No further flaws were discovered during the penetration testing.

EVALUATION AS A B2 MDIA NETWORK COMPONENT

Because the VSLAN has been evaluated against the TNI, rather than the TCSEC, the format of this section is structured differently from that of reports of TCSEC-based evaluations. Each requirement section will cite the requirement from the TCSEC as well as the interpretation from the TNI. This is followed by additional interpretations applicable to network components, where such additional interpretations exist. Finally, the applicable features are enumerated.

Because of printing errors, there is a discrepancy between the words which the TNI claims to be from the TCSEC and the actual wording of the TCSEC. Therefore, the authors of this report have used the original text of the TCSEC, instead of the wording claimed by the TNI to be the text of the TCSEC. There is one consistent change made throughout the TCSEC text in the TNI which does not seem to be the result of a printing error. The phrase "sensitivity level" has been used in the TNI for the TCSEC's "security level." Although the original term "security level" has been restored in the requirements section, the term "sensitivity level" has been retained in the interpretations quoted from the TNI.

According to Appendix A of the TNI, there are some TCSEC requirements which do not apply to network components such as the VSLAN (e.g., network components which do not directly support user input). This has been so noted in the Conclusion section of these requirements.

Discretionary Access Control Requirement

The TCB shall define and control access between named users and named objects (e.g., files and programs) in the ADP system. The enforcement mechanism (e.g., self/group/public controls, access control lists) shall allow users to specify and control sharing of those objects by named individuals or defined groups of individuals, or both, and shall provide controls to limit propagation of access rights. The discretionary access control mechanism sha'l, either by explicit user action or by default, provide that objects are protected from unauthorized access. These access controls shall be capable of including or excluding access to the granularity of a single user. Access permission to an object by users not already possessing access permission shall only be assigned by authorized users.

Interpretation

The discretionary access control (DAC) mechanism(s) may be distributed over the partitioned NTCB in various ways. Some part, all, or none of the DAC may be implemented in a given component of the network system. In particular, components that support only internal subjects (i.e., that have no subjects acting as direct surrogates for users), such as a public network packet switch, might not implement the DAC mechanism(s) directly (e.g., they are unlikely to contain access control lists).

Identification of users by groups may be achieved in various ways in the networking environment. For example, the network identifiers (e.g., inter-net addresses) for various components (e.g., hosts, gateways) can be used as identifiers of groups of individual users (e.g., "all users at Host A," "all users of network Q") so long as the individuals involved in the group are implied by the group identifier. For example, Host A might employ a particular group-id, for which it maintains a list of explicit users in that group, in its network exchange with Host B, which accepts the group-id under the conditions of this interpretation.

For networks, individual hosts will impose need-to-know controls over their users on the basis of named individuals -- much like (in fact, probably the same) controls used when there is no network connection.

When group identifiers are acceptable for access control, the identifier of some other host may be employed, to eliminate the maintenance that would be required if individual identification of remote users was employed. In class C2 and higher, however, it must be possible from that audit record to identify (immediately or at some later time) exactly the individuals represented by a group identifier at the time of the use of that identifier. There is allowed to be an uncertainty because of elapsed time between changes in the group membership and the enforcement in the access control mechanisms.

The DAC mechanism of a NTCB partition may be implemented at the interface of the reference monitor or may be distributed in subjects that are part of the NTCB in the same or different component. The reference monitor manages all the physical resources of the system and from them creates the abstraction of subjects and objects that it controls. Some of these subjects and objects may be used to implement a part of the NTCB. When the DAC mechanism is distributed in such NTCB subjects (i.e., when outside the reference monitor), the assurance requirements (see the Assurance section) for the design and implementation of the DAC shall be those of class C2 for all networks of class C2 or above.

When integrity is included as part of the network discretionary security policy, the above interpretations shall be specifically applied to the controls over modification, viz, the write mode of access, within each component based on identified users or groups of users.

Applicable Features

The VSLAN is capable of supporting DAC in a complete network system. The VSLAN resides at the Data Link Layer, so it does not have subjects which act as direct surrogates for users and thus does not mediate access between named users and named objects. Instead, the VSLAN mediates access between groups of users (using each host denoted by the principal identifier) and datagrams, leaving it to the hosts to further mediate access to objects residing on that host, on the basis of named individuals. All VSLAN audit records will identify the principal responsible for generation of the record. From this information, it is possible to identify the individuals associated with the principal identifier at the time of its use, using procedures described in the network security architecture document [19].

Conclusion

The VSLAN satisfies the B2 Discretionary Access Control requirement.

Object Reuse Requirement

All authorizations to the information contained within a storage object shall be revoked prior to initial assignment, allocation or reallocation to a subject from the TCB's pool of unused storage objects. No information, including encrypted representations of information, produced by a prior subject's actions is to be available to any subject that obtains access to an object that has been released back to the system.

Interpretation

The NTCB shall ensure that any storage objects that it controls (e.g., message buffers under the control of a NTCB partition in a component) contain no information for which a subject in that component is not authorized before granting access. This requirement must be enforced by each of the NTCB partitions.

Applicable Features

The host has access to the host/NSD shared memory. This is the only memory on the NSD to which the host has access. The memory is cleared when any of the following occurs:

- 1) The Datakey is inserted.
- 2) The Datakey is removed.
- 3) The power is switched off.

Upon reset (from Datakey insertion or power on) the NSD disables all host access to the board until diagnostics have been completed. The memory diagnostics clear memory by writing specific patterns to the memory: first, each byte is written with the pattern 55H (alternating zeroes and ones) and then with the pattern AAH (alternating ones and zeroes). Then, each word (2 bytes) is written with a number in ascending order, beginning with 1111H.

When the Datakey is removed, the NSD writes the value 0FFH to all bytes of the shared memory.

The NSD and NSD-Prime Null task, upon receiving the Null message, invokes the Kernel Memory Management Service to scrub free internal data buffers. The Null message is repeatedly sent by the Null task to itself, so the scrubbing function is invoked whenever the Null task runs. Scrubbing is performed by writing each bit with a fixed pattern.

Conclusion

The VSLAN satisfies the B2 Object Reuse requirement.

Labels Requirement

Sensitivity labels associated with each ADP system resource (e.g., subject, storage object, ROM) that is directly or indirectly accessible by subjects external to the TCB shall be maintained by the TCB. These labels shall be used as the basis for mandatory access control decisions. In order to import non-labeled data, the TCB shall request and receive from an authorized user the sensitivity level of the data, and all such actions shall be auditable by the TCB.

Interpretation

Non-labeled data imported under the control of the NTCB partition will be assigned a label constrained by the device labels of the single-level device used to import it. Labels may include secrecy and integrity[1] components in accordance with the overall network security policy described by the network sponsor. Whenever the term "label" is used throughout this interpretation, it is understood to include both components as applicable. Similarly, the terms "single-level" and "multilevel" are understood to be based on both the secrecy and integrity components of the policy. The mandatory integrity policy will typically have requirements, such as the probability of undetected message stream modification, that will be reflected in the label for the data so protected. For example, when data is imported its integrity label may be assigned based on mechanisms, such as cryptography, used to provide the assurance required by the policy. The NTCB shall assure that such mechanism are protected from tampering and are always invoked when they are the basis for a label.

If the security policy includes an integrity policy, all activities that result in message-stream modification during transmission are regarded as unauthorized accesses in violation of the integrity policy. The NTCB shall have an automated capability for testing, detecting, and reporting those errors/corruptions that exceed specified network integrity policy requirements. Message-stream modification (MSM) countermeasures shall be identified. A technology of adequate strength shall be selected to resist MSM. If encryption methodologies are employed, they shall be approved by the national security agency.

All objects must be labeled within each component of the network that is trusted to maintain separation of multiple levels of information. The label associated with any objects associated with single-level components will be identical to the level of that component. Objects used to store network control information, and other network structures, such as routing tables, must be labeled to prevent unauthorized access and/or modification.

[1] See, for example, Biba, K. J., *Integrity Considerations for Secure Computer Systems*, ESD-TR-76-372, MTR-3153, The MITRE Corporation, Bedford, MA, April 1977.

Applicable Features

Each datagram received by the NSD from the subject contains a label which is placed in the datagram by the subject. The NSD reads the label and enforces the mandatory access control policy based upon this label. The label and data will be linked in this manner throughout the time that the datagram is within the VSLAN.

The VSLAN NTCB partition accepts no unlabeled data.

Conclusion

The VSLAN satisfies the B2 Labels requirement.

Label Integrity Requirement

Sensitivity labels shall accurately represent security levels of the specific subjects or objects with which they are associated. When exported by the TCB, sensitivity labels shall accurately and unambiguously represent the internal labels and shall be associated with the information being exported.

Interpretation

The phrase "exported by the TCB" is understood to include transmission of information from an object in one component to an object in another component. Information transferred between NTCB partitions is addressed in the System Integrity Section. The form of internal and external (exported) sensitivity labels may differ, but the meaning shall be the same. The NTCB shall, in addition, ensure that correct association of sensitivity labels with the information being transported across the network is preserved.

As mentioned in the Trusted Facility Manual Section, encryption transforms the representation of information so that it is unintelligible to unauthorized subjects. Reflecting this transformation, the sensitivity level of the cipher text is generally lower than the clear text. It follows that clear text and cipher text are contained in different objects, each possessing its own label. The label of the clear text must be preserved and associated with the cipher text so that it can be restored when the clear text is subsequently obtained by decrypting the cipher text. If the clear text is associated with a single-level device, the label of that clear text may be implicit. The label may also be implicit in the key.

When information is exported to an environment where it is subject to deliberate or accidental modification, the TCB shall support the means, such as cryptographic checksums, to assure the accuracy of the labels. When there is a mandatory integrity policy, the policy will define the meaning of integrity labels.

Applicable Features

Each host datagram processed by an NSD has an associated sensitivity label. The label format is a 16-bit word which ranges in value from 0 to 15 (thus 16 levels) and an 8-octet (64-bit) string that comprises the 64 categories (1 bit per category).

From the point at which the NSD copies the datagram from the host/NSD shared memory into its internal memory, the NSD ensures that the label value cannot be changed. The NSD hardware has parity-checked RAM to reduce the likelihood that RAM failures will cause the value of the level or category to be changed while it resides in the NSD memory. The likelihood of errors during transmission is reduced through the use of checksums. The frame header (which includes sensitivity label, principal id, and length) has a checksum that is calculated by the transmit policy control task after the packet has successfully completed transmit mediation. The encryption protocol task places a checksum of the entire NSD packet (including the label and label checksum) at the end of the packet before the packet is encrypted. When the NSD transmits the packet onto the LAN, the Intel 82586 also places its own CRC at the end to detect transmission errors.

When the NSD puts the datagram into the host/NSD shared memory for the host, the label associated with the datagram is the same label which had been associated with the datagram throughout its existence in the NTCB.

Conclusion

The VSLAN satisfies the B2 Label Integrity requirement.

Exportation of Labeled Information Requirement

The TCB shall designate each communication channel and I/O device as either single-level or multilevel. Any change in this designation shall be done manually and shall be audit able by the TCB. The TCB shall maintain and be able to audit any change in the sensit vity level or levels associated with a communications channel or I/O device.

Interpretation

Each communication channel and network component shall be designated as either single-level or multilevel. Any change in this designation shall be done with the cognizance and approval of the administrator or security officer in charge of the affected components and the administrator or security officer in charge of the NTCB. This change shall be auditable by the network. The NTCB shall maintain and be able to audit any change in the device labels associated with a single-level communication channel or the range associated with a multilevel communication channel or component. The NTCB shall also be able to audit any change in the set of sensitivity levels associated with the information which can be transmitted over a multilevel communication channel or component.

Applicable Features

For the VSLAN, there are no I/O devices; therefore, the requirement is implicitly satisfied for I/O devices. The communication channels in the VSLAN are the host/NSD shared memory segments connecting hosts with their NSDs. These communication channels are noted as single-level or multilevel by the send and receive window of the hosts attached to their respective NSD. These windows can be changed only by the security officer at the NSC, and this action is auditable.

Conclusion

The VSLAN satisfies the B2 Exportation of Labeled Information requirement.

Exportation to Multi-level Devices Requirement

When the TCB exports an object to a multilevel I/O device, the sensitivity label associated with that object shall also be exported and shall reside on the same physical medium as the exported information and shall be in the same form (i.e., machine-readable or human-readable form). When the TCB exports or imports an object over a multilevel communication channel, the protocol used on that channel shall provide for the unambiguous pairing between the sensitivity labels and the associated information that is sent or received.

Interpretation

The components, including hosts, of a network shall be interconnected over "multilevel communication channels," multiple single-level communication channels, or both, whenever the information is to be protected at more than a single sensitivity level. The protocol for associating the sensitivity label and the exported information shall provide the only information needed to correctly associate a sensitivity level with the exported information transferred over the multilevel channel between the NTCB partitions in individual components. This protocol definition must specify the representation and semantics of the sensitivity labels (i.e., the machine-readable label must uniquely represent the sensitivity level).

The "unambiguous" association of the sensitivity level with the communicated information shall meet the same level of accuracy as that required for any other label within the NTCB, as specified in the criterion for Label Integrity. This may be provided by protected and highly reliable direct physical layer connections, or by traditional cryptographic link protection in which any errors during transmission can be readily detected, or by use of a separate channel. The range of information imported or exported must be constrained by the associated device labels.

Applicable Features

The NSD can import and export datagrams through the host/NSD shared memory. The protocol for communication with the NSD through this shared memory interface specifies the ICB and frame headers that include the sensitivity label and the offset in the shared

memory of where the datagram resides.
A detailed explanation of this interface is provided in the NSD external interface specification (see page 24, "NSD and NSD-Prime Data Structures").

Conclusion

The VSLAN satisfies the B2 Exportation to Multi-Level Devices requirement.

Exportation to Single-Level Devices Requirement

Single-level I/O devices and single-level communication channels are not required to maintain the sensitivity labels of the information they process. However, the TCB shall include a mechanism by which the TCB and an authorized user reliably communicate to designate the single security level of information imported or exported via single-level communication channels or I/O devices.

Interpretation

Whenever one or both of two directly connected components is not trusted to maintain the separation of information of different sensitivity levels, or whenever the two directly connected components have only a single sensitivity level in common, the two components of the network shall communicate over a single-level channel. Single-level components and single-level communication channels are not required to maintain the sensitivity labels of the information they process. However, the NTCB shall include a reliable communication mechanism by which the NTCB and an authorized user (via a trusted path) or a subject within an NTCB partition can designate the single sensitivity level of information imported or exported via single-level communication channels or network components. The level of information communicated must equal the device level.

Applicable Features

The NSD may have a communication interface wherein both the minimum and maximum security levels are the same. Such an interface is otherwise implemented exactly the same as a multilevel interface.

Conclusion

The VSLAN satisfies the B2 Exportation to Single-Level Device. requirement.

Labeling Human-Readable Output Requirement

The ADP system administrator shall be able to specify the printable label names associated with exported sensitivity labels. The TCB shall mark the beginning and end of all human-readable, paged, hard copy output (e.g., line printer output) with human-readable sensitivity labels that properly(1) represent the sensitivity of the output. The TCB shall, by default, mark the top and bottom of each page of human-readable, paged, hard copy output (e.g., line printer output) with human-readable sensitivity labels that properly represent the overall sensitivity of the output or that properly represent the sensitivity of the information on the page. The TCB shall, by default and in an appropriate manner, mark other forms of human-readable output (e.g., maps, graphics) with human-readable sensitivity labels that properly represent the senritivity of the output. Any override of these marking defaults shall be auditable by the TCB.

Interpretation

This criterion imposes no requirement to a component that produces no human-readable output. For those that do produce human-readable output, each sensitivity level that is defined to the network shall have a uniform meaning across all components. The network administrator, in conjunction with any affected component administrator, shall be able to specify the human-readable label that is associated with each defined sensitivity level.

Applicable Features

The VSLAN produces no exportable human-readable output.

Conclusion

The B2 Labeling Human-Readable Output requirement is not applicable to the VSLAN.

Subject Sensitivity Labels Requirement

The TCB shall immediately notify a terminal user of each change in the security level associated with that user during an interactive session. A terminal user shall be able to query the TCB as desired for a display of the subject's complete sensitivity label.

Interpretation

An NTCB partition shall immediately notify a terminal user attached to its component of each change in the sensitivity level associated with that user.
Additional Network Component Interpretation

An M-Component need not support direct terminal input in which case this requirement is not applicable. Any M-Component which does support direct terminal input must meet the requirement as stated.

Applicable Features

The VSLAN does not support direct terminal input.

Conclusion

The B2 Subject Sensitivity Labels requirement is not applicable to the VSLAN.

Device Labels Requirement

The TCB shall support the assignment of minimum and maximum security levels to all attached physical devices. These security levels shall be used by the TCB to enforce constraints imposed by the physical environments in which the devices are located.

Interpretation

This requirement applies as written to each NTCB partition that is trusted to separate information based on sensitivity level. Each I/O device in a component, used for communication with other network components, is assigned a device range, consisting of a set of labels with a maximum and minimum. (A device range usually contains, but does not necessarily contain, all possible labels "between" the maximum and minimum, in the sense of dominating the minimum and being dominated by the maximum.)

The NTCB always provides an accurate label for information exported through devices. Information exported or imported using a single-level device is labelled implicitly by the sensitivity level of the device. Information exported from one multilevel device and imported at another must be labelled through an agreed-upon protocol, unless it is labelled implicitly by using a communication link that always carries a single level.

Information exported at a given sensitivity level can be sent only to an importing device whose device range contains that level or a higher level. If the importing device range does not contain the given level, the information is relabelled upon reception at a higher level within the importing device range. Relabelling should not occur otherwise.

Applicable Features

There are no I/O devices within the VSLAN used for communication. The hosts are not devices under the control of the VSLAN.

Conclusion

The B2 Device Labels requirement is not applicable to the VSLAN.

Mandatory Access Control Requirement

The TCB shall enforce a mandatory access control policy over all resources (i.e., subjects, storage objects, and I/O devices) that are directly or indirectly accessible by subjects external to the TCB. These subjects and objects shall be assigned sensitivity labels that are a combination of hierarchical classification levels and non-hierarchical categories, and the labels shall be used as the basis for mandatory access control decisions. The TCB shall be able to support two or more such security levels. (See the Mandatory Access Control guidelines.) The following requirements shall hold for all accesses between all subjects external to the TCB and all objects directly or indirectly accessible by these subjects: A subject can read an object only if the hierarchical classification in the subject's security level is greater than or equal to the hierarchical classification in the object's security level and the non-hierarchical categories in the subject's security level include all the non-hierarchical categories in the object's security level. A subject can write an object only if the hierarchical classification in the subject's security level is less than or equal to the hierarchical classification in the object's security level and all the non-hierarchical categories in the subject's security level are included in the non-hierarchical categories in the object's security level. Identification and authentication data shall be used by the TCB to authenticate the user's identity and to ensure that the security level and authorization of subjects external to the TCB that may be created to act on behalf of the individual user are dominated by the clearance and authorization of that user.

Interpretation

Each partition of the NTCB exercises mandatory access control policy over all subjects and objects in its component. In a network, the responsibility of an NTCB partition encompasses all mandatory access control functions in its component that would be required of a TCB in a stand-alone system. In particular, subjects and objects used for communication with other components are under the control of the NTCB partition. Mandatory access control includes secrecy and integrity control to the extent that the network sponsor has described in the overall network security policy.

Conceptual entities associated with communication between two components, such as sessions, connections and virtual circuits, may be thought of as having two ends, one in each component, where each end is represented by a local object. Communication is viewed as an operation that copies information from an object at one end of a communication path to an object at the other end. Transient data-carrying entities, such as datagrams and packets, exist either as information within other objects, or as a pair of objects, one at each end of the communication path.

The requirement for "two or more" sensitivity levels can be met by either secrecy or integrity levels. When there is a mandatory integrity policy, the stated requirements for reading and writing are generalized to: A subject can read an object only if the subject's sensitivity level dominates the object's sensitivity level, and a subject can write an object only if the object's sensitivity level dominates the subject's sensitivity level. Based on the integrity policy, the network sponsor shall define the dominance relation for the total label, for example, by combining secrecy and integrity lattices[1].

Applicable Features

The NSD mediates access between subjects and datagrams based upon the labels of the datagrams. All data is labeled; the NSD supports up to 16 hierarchical levels and 64 non-hierarchical categories. The transmit and receive windows of an NSD are based upon the allowable security levels of the principal associated with that NSD.

A datagram which is to be transmitted must first pass through the Transmit Policy Control Task (TPCT), which compares the label that is associated with the datagram with the transmit window of the NSD. If the security label is within this window (i.e., the datagram level dominates the minimum security level and is dominated by the maximum security level prescribed by the security administrator), then TPCT allows the datagram to be transmitted to the destination NSD.

A datagram which is received must pass through the Receive Policy Control Task (RPCT) before it is delivered to the host system. This task compares the label that is associated with the datagram with the receive window of the NSD. If the security label is within this window (i.e., the datagram level dominates the minimum security level and is dominated by the maximum security level prescribed by the security administrator), then RPCT allows the datagram to be received by the host system.

Conclusion

The VSLAN satisfies the B2 Mandatory Access Control requirement.

Identification and Authentication Requirement

The TCB shall require users to identify themselves to it before beginning to perform any other actions that the TCB is expected to mediate. Furthermore, the TCB shall maintain authentication data that includes information for verifying the identity of individual users (e.g., passwords) as well as information for determining the clearance and authorizations of individual users. This data shall be used by the TCB to authenticate the user's identity and to ensure that the security level and authorizations of subjects external to the TCB that may be created to act on behalf of the individual user are dominated by the clearance and authorization of that user. The TCB shall protect authentication data so that it cannot be accessed by any unauthorized user. The TCB shall be able to enforce individual accountability by providing the capability to uniquely identify each individual ADP system user. The TCB shall also provide the capability of associating this identity with all auditable actions taken by that individual.

[1] See, for example, Grohn. M. J., *A Model of a Protected Data Management System*, ESD-TR-76-289m I. P. Sharp Association Limited, June 1976; and Denning, D. E., Lunt, T. F., Neuman, P. R., Schell, R. R., Heckman, M. and Shockely, W., *Secure Distributed Data Views, Security Policy and Interpretation for a CLass A1 Multilevel Secure Relational Database System*, SRI International, November 1986.

Interpretation

The requirement for identification and authentication of users is the same for a network system as for an ADP system. The identification and authentication may be done by the component to which the user is directly connected or some other component, such as an identification and authentication server. Available techniques, such as those described in the Password Guideline[1], are generally also applicable in the network context. However, in cases where the NTCB is expected to mediate actions of a host (or other network component) that is acting on behalf of a user or group of users, the NTCB may employ identification and authentication of the host (or other component) in lieu of identification and authentication of an individual user, so long as the component identifier implies a list of specific users uniquely associated with the identifier at the time of its use for authentication. This requirement does not apply to internal subjects.

Authentication information, including the identity of a user (once authenticated) may be passed from one component to another without reauthentication, so long as the NTCB protects (e.g., by encryption) the information from unauthorized disclosure and modification. This protection shall provide at least a similar level of assurance (or strength of mechanism) as pertains to the protection of the authentication mechanism and authentication data.

Applicable Features

The VSLAN is capable of supporting the identification and authentication requirement for a complete network system by identifying and authenticating principals. A principal is associated with a specific host/Datakey pair and denotes a collection of users authorized to use that particular host while a valid Datakey is inserted in its associated NSD. Because the VSLAN does not directly support users (i.e., human beings), the VSLAN is capable of mediating only the actions of attached hosts, each of which implies a group of users.

In order to provide individual accountability for the complete network system, each host is responsible for providing identification and authentication for all users authorized to use that host. In addition, the administrative personnel responsible for incorporating the VSLAN into their network system must also procedurally maintain a list of authorized users that is associated with each host connected to the network at the time the NSD is initialized. This responsibility is outside the scope of the VSLAN, but within the scope of the network system that has incorporated the VSLAN.

The VSLAN requires that principals identify and authenticate themselves to it, via Datakey insertion, before beginning to perform any other actions that the NTCB is expected to mediate. The NSC maintains authentication data, in the form of the principal table. This table, along with default profiles, ensures that principals are assigned the proper security levels, and that all auditable actions taken by principals, as well as the group of users that the principal represents, are associated with the individual principal

[1] *Department of Defense Password Management Guideline*, CSC-STD-002-85.

identifier.

Conclusion

The VSLAN satisfies the B2 Identification and Authentication requirement.

Trusted Path Requirement

The TCB shall support a trusted communication path between itself and user [sic] for initial login and authentication. Communications via this path shall be initiated exclusively by a user.

Interpretation

A trusted path is supported between a user (i.e., human) and the NTCB partition in the component to which the user is directly connected.

Additional Network Component Interpretation

An M-Component need not support direct user input (e.g., the M-Component may not be attached to any user I/O devices such as terminals) in which case this requirement is not applicable. Any M-Component which does support direct communication with users must meet the requirement as stated. In addition, an M-Component with directly connected users must provide mechanisms which establish the clearance of users and associate that clearance with the users current session.

Applicable Features

The VSLAN does not support direct user communication.

Conclusion

The B2 Trusted Path requirement is not applicable to the VSLAN.

Audit Requirement

The TCB shall be able to create, maintain, and protect from modification or unauthorized access or destruction an audit trail of accesses to the objects it protects. The audit data shall be protected by the TCB so that read access to it is limited to those who are authorized for audit data. The TCB shall be able to record the following types of events: use of identification and authentication mechanisms, introduction of objects into a user's address space (e.g., file open, program initiation), deletion of objects, actions taken by computer operators and system administrators and/or system security officers, and other security relevant events. The TCB shall also be able to audit any override of human-readable output markings. For each recorded event, the audit record shall identify: date and time of the event, user, type of event, and success or failure of the event. For identification/authentication events the origin of request (e.g., terminal ID) shall be included in the audit record. For events that introduce an object into a user's address space and for object deletion events the audit record shall include the name of the object and the object's security level. The ADP system administrator shall be able to selectively audit the actions of any one or more users based on individual identity and/or object security level. The TCB shall be able to audit the identified events that may be used in the exploitation of covert storage channels.

Interpretation

This criterion applies as stated. The sponsor must select which events are auditable. If any such events are not distinguishable by the NTCB alone (for example those identified in Part II), the audit mechanism shall provide an interface, which an authorized subject can invoke with parameters sufficient to produce an audit record. These audit records shall be distinguishable from those provided by the NTCB. In the context of a network system, "other security relevant events" (depending on network system architecture and network security policy) might be as follows:

1. Identification of each access event (e.g., establishing a connection or a connectionless association between processes in two hosts of the network) and its principal parameters (e.g., host identifiers of the two hosts involved in the access event and user identifier or host identifier of the user or host that is requesting the access event)

2. Identification of the starting and ending times of each access event using local time or global synchronized time

3 Identification of security-relevant exceptional conditions (e.g., potential violation of data integrity, such as misrouted datagrams) detected during the transactions between two hosts

4. Utilization of cryptographic variables

5. Changing the configuration of the network (e.g., a component leaving the network and rejoining)

In addition, identification information should be included in appropriate audit trail records, as necessary, to allow association of all related (e.g., involving the same network event) audit trail records (e.g., at different hosts) with each other. Furthermore, a component of the network system may provide the required audit capability (e.g., storage, retrieval, reduction, analysis) for other components that do not internally store audit data but transmit the audit data to some designated collection component. Provisions shall be made to control the loss of audit data due to unavailability of resources.

In the context of a network system, the "user's address space" is extended, for object introduction and deletion events, to include address spaces being employed on behalf of a remote user (or host). However, the focus remains on users in contrast to internal subjects as discussed in the DAC criterion. In addition, audit information must be stored in machine-readable form.

The capability must exist to audit the identified events that may be used in the exploitation of covert storage channels. To accomplish this, each NTCB partition must be able to audit those events locally that may lead to the exploitation of a covert storage channel which exist because of the network.

Applicable Features

The VSLAN is capable of generating audit records for user datagram information, network status information, and all security officer actions. The following types of events cause audit data to be generated: initialization and termination of each NSD, host transmission and reception of datagrams[1], all security officer actions, and other security relevant events (see page 43, "Audit"). Only VSLAN NTCB software is capable of generating and processing raw audit data. All audit data is collected and maintained in the NSC and is protected so that only security officers are allowed access.

The VSLAN has the capability to audit every datagram transmitted or received by a host when statistical auditing is turned on. In doing so, the security level of the datagram is recorded along with the source and destination NSDs. In addition, the initialization and termination of every NSD generates an audit record. The format of an NSD audit record includes the date and time of the event, the source and destination NSD and principal, and the type of event that was generated (see page 32, "NSC Data Structures"). A failure of an event is identified by the audit event type (e.g., MAC violation, DAC violation, faulty Datakey insertion).

By using the VSLAN Audit Post Processing (VAPP) tool, The VSLAN security officer has the capability to select audit records consisting of all datagrams processed by one or more hosts. In addition, the VAPP software allows the security officer to select audit records by any one of the fields in the audit record, including the NSD ID and the security level (see page 43, "Audit").

All known events that may be used in the exploitation of covert channels are also audited by the VSLAN.

Conclusion

The VSLAN satisfies the B2 Audit requirement.

[1] Host transmission and reception of datagrams is audited only when statistical auditing is turned on.

System Architecture Requirement

The TCB shall maintain a domain for its own execution that protects it from external inteˀference or tampering (e.g., by modification of its code or data structures). The TCB shall maintain process isolation through the provision of distinct address spaces under its control. The TCB shall be internally structured into well-defined largely independent modules. It shall make effective use of available hardware to separate those elements that are protection-critical from those that are not. The TCB modules shall be designed such that the principle of least privilege is enforced. Features in hardware, such as segmentation, shall be used to support logically distinct storage objects with separate attributes (namely: readable, writable). The user interface to the TCB shall be completely defined and all elements of the TCB identified.

Interpretation

The system architecture criterion must be met individually by all NTCB partitions. Implementation of the requirement that the NTCB maintain a domain for its own execution is achieved by having each NTCB partition maintain a domain for its own execution. Since each component is itself a distinct domain in the overall network system, this also satisfies the requirement for process isolation through distinct address spaces in the special case where a component has only a single subject.

The NTCB must be internally structured into well-defined largely independent modules and meet the hardware requirements. This is satisfied by having each NTCB partition so structured. The NTCB controls all network resources. These resources are the union of the sets of resources over which the NTCB partitions have control. Code and data structures belonging to the NTCB, transferred among NTCB subjects (i.e., subjects outside the reference monitor but inside the NTCB) belonging to different NTCB partitions, must be protected against external interference or tampering. For example, a cryptographic checksum or physical means may be employed to protect user authentication data exchanged between NTCB partitions.

Each NTCB partition must enforce the principle of least privilege within its component. Additionally, the NTCB must be structured so that the principle of least privilege is enforced in the system as a whole.

Each NTCB partition provides isolation of resources (within its component) in accord with the network system architecture and security policy so that "supporting elements" (e.g., DAC and user identification) for the security mechanisms of the network system are strengthened compared to C2, from an assurance point of view, through the provision of distinct address spaces under control of the NTCB.

As discussed in the Discretionary Access Control section, the DAC mechanism of a NTCB partition may be implemented at the interface of the reference monitor or may be distributed in subjects that are part of the NTCB in the same or different component. When distributed in NTCB subjects (i.e., when outside the reference monitor), the assurance requirements for the design and implementation of the DAC shall be those of class C2 for all networks of class C2 or above.

Additional Network Component Interpretation

An M-Component must meet the requirement as stated. In this interpretation the words "The user interface to the TCB shall be completely defined..." shall be interpreted to mean the interface between the reference monitor of the M-Component and the subjects external to the reference monitor shall be completely defined.

Applicable Features

Except for the host/NSD shared memory, all VSLAN hardware and software resides within the NTCB. The NSD hardware is designed in such a way that it prohibits the mediation software from being circumvented or from being affected by any external untrusted software. This is accomplished by the separation of memory sections accessible by different active processing entities. The attached host can access only the host/NSD shared memory, while the network interface can access only a different area of NSD memory. Only the NSD processor can access both of these areas. Because the processor is the only NSD entity which can access both the network and host RAMs, all movement of data between the two must be mediated by the NSD software. There is also a special address monitoring circuit on the NSD that independently verifies that the host and network interfaces access only their designated areas of memory.

As described in the software architecture overview (see page 19, "Software Architecture"), the NSD software is organized as a group of tasks, each performing a specific function relative to the communications and security requirements of the VSLAN. The tasks communicate with each other and with the hardware only through a well defined set of procedure calls. A queued message-passing system is the mechanism used for inter-task communication and scheduling. The Intel 80286 facilities for process separation are used extensively to further ensure that an individual task cannot inadvertently corrupt another's code, data, or stack space. By using each task's LDT as the mechanism for accessing datagram buffers, and allowing the buffer address to be in only one LDT at a time, the NSD software ensures that the buffer will not be affected in an unexpected manner by two tasks.

The code, stack, and data segments and the TSS definitions for each task are stored in the NSD EPROMs. During NSD initialization, the task segment descriptors are loaded into RAM, and then the tasks are started. The tasks themselves are all part of the NTCB.

The interface between the VSLAN and the subjects external to it is completely defined in the external interface document [11], which describes the data structures, the protocol, and the host bus interface conventions used by a host to communicate with an NSD.

For the NSC, Verdix has chosen to develop its own operating system. This system, the Verdix Operating System (VOS), uses the same design and much of the same code as that used in the NSD. The applications supporting the administrator and operator run on the NSC. The VOS kernel is the same as the NSD separation kernel except for the addition of some device drivers.

Conclusion

The VSLAN satisfies the B2 System Architecture requirement.

System Integrity Requirement

Hardware and/or software features shall be provided that can be used to periodically validate the correct operation of the on-site hardware and firmware elements of the TCB.

Interpretation

Implementation of the requirement is partly achieved by having hardware and/or software features that can be used to periodically validate the correct operation of the hardware and firmware elements of each component's NTCB partition. Features shall also be provided to validate the identity and correct operation of a component prior to its incorporation in the network system and throughout system operation. For example, a protocol could be designed that enables the components of the partitioned NTCB to exchange messages periodically and validate each other's correct response. The protocol shall be able to determine the remote entity's ability to respond. NTCB partitions shall provide the capability to report to network administrative personnel the failures detected in other NTCB partitions.

Intercomponent protocols implemented within a NTCB shall be designed in such a way as to provide correct operation in the case of failures of network communications or individual components. The allocation of mandatory and discretionary access control policy in a network may require communication between trusted subjects that are part of the NTCB partitions in different components. This communication is normally implemented with a protocol between the subjects as peer entities. Incorrect access within a component shall not result from failure of an NTCB partition to communicate with other components.

Applicable Features

The NSD and NSD-Prime software, which are stored in EPROM, have an associated checksum of the code which is checked at initialization for firmware integrity. There are also a variety of tests used to verify the correct operation of the hardware. The majority of these tests are in the form of diagnostic tasks, which are listed in another section of this report (see page 53, "System Integrity"). These tests are performed automatically by the NSD software after each reset of the board.

The functionality of the Intel 82586 LAN co-processor can be tested by operating the network interface in internal loopback mode, which is part of the initialization diagnostic testing on system reset.

In the VSLAN, the NSC periodically sends out a status poll message from the NSD-Prime to each on-line NSD. If the NSD does not acknowledge the message, it is assumed by the NSC that the NSD has gone to the off-line state. If the NSD does not receive the status poll from the NSC, the NSD will assume that the NSC has gone to the off-line state and shut itself down.

Conclusion

VSLAN satisfies the B2 System Integrity requirement.

Covert Channel Analysis Requirement

The system developer shall conduct a thorough search for covert storage channels and make a determination (either by actual measurement or by engineering estimation) of the maximum band width of each identified channel. (See the Covert Channels Guideline section.)

Interpretation

The requirement, including the TCSEC covert channel guideline, applies as written. In a network, there are additional instances of covert channels associated with communication between components.

Additional Network Component Interpretation

An M-Component must meet the requirement as stated. In addition, if the analysis indicates that channels exist that need to be audited (according to the Covert Channel Analysis Guideline), the M-Component shall contain a mechanism for making audit data (related to possible use of covert channels) available outside of the M-Component (e.g., by passing the data to an audit collection component).

Applicable Features

Verdix's approach to conducting covert channel analysis is discussed earlier in this report (see page 50, "Covert Channel Analysis"). Verdix has produced documentation describing its approach and an overview of the SRM methodology as it applies to the VSLAN, as well as its covert channel analysis and the results of it. Verdix describes in detail each identified covert channel and how it can be used, the engineering estimates for the band widths of each identified channel, how it derived those estimates, and the actions that can be taken to limit the use of these channels.

Conclusion

The VSLAN satisfies the B2 Covert Channel Analysis requirement.

Trusted Facility Management Requirement

The TCB shall support separate operator and administrator functions.

Interpretation

This requirement applies as written to both the network as a whole and to individual components which support such personnel.

Applicable Features

The VSLAN architecture supports separate Security Operator and Security Administrator roles. A person must be logged in at the NSC to assume one of these roles. The NSC software uses the login identifier and password to determine which role that person may assume. A person in the Security Administrator role may perform all of the allowable commands at the NSC while someone in the Security Operator role may perform only a subset of those commands. A more detailed explanation of the specific commands which a Security Administrator may execute and those which a Security Operator is restricted from executing can be found in the NSC software architecture section of this report (see page 34, "NSC Application Processes").

Conclusion

The VSLAN satisfies the B2 Trusted Facility Management requirement.

Security Testing Requirement

The security mechanisms of the ADP system shall be tested and found to work as claimed in the system documentation. A team of inaıviduals who thoroughly understand the specific implementation of the TCB shall subject its design documentation, source code, and object code to thorough analysis and testing. Their objectives shall be: to uncover all design and implementation flaws that would permit a subject external to the TCB to read, change, or delete data normally denied under the mandatory or discretionary security policy enforced by the TCB; as well as to assure that no subject (without authorization to do so) is able to cause the TCB to enter a state such that it is unable to respond to communications initiated by other users. The TCB shall be found relatively resistant to penetration. All discovered flaws shall be corrected and the TCB retested to demonstrate that they have been eliminated and that new flaws have not been introduced. Testing shall demonstrate that the TCB implementation is consistent with the descriptive top-level specification. (See the Security Testing Guidelines.)

Interpretation

Testing of a component will require a test bed that exercises the interfaces and protocols of the component including tests under exceptional conditions. The testing of a security mechanism of the network system for meeting this criterion shall be an integrated testing procedure involving all components containing an NTCB partition that implement the given mechanism. This integrated testing is additional to any individual component tests involved in the evaluation of the network system. The sponsor should identify the allowable set of configurations including the sizes of the networks. Analysis or testing procedures and tools shall be available to test the limits of these configurations. A change in configuration within the allowable set of configurations does not require retesting.

The testing of each component will include the introduction of subjects external to the NTCB partition for the component that will attempt to read, change, or delete data normally denied. If the normal interface to the component does not provide a means to create the subjects needed to conduct such a test, then this portion of the testing shall use a special version of the untrusted software for the component that results in subjects that make such attempts. The results shall be saved for test analysis. Such special versions shall have an NTCB partition that is identical to that for the normal configuration of the component under evaluation.

The testing of the mandatory controls shall include tests to demonstrate that the labels for information imported and/or exported to/from the component accurately represent the labels maintained by the NTCB partition for the component for use as the basis for its mandatory access control decisions. The tests shall include each type of device, whether single-level or multilevel, supported by the component.

The NTCB must be found relatively resistant to penetration. This applies to the NTCB as a whole, and to each NTCB partition in a component of this class.

Additional Network Component Interpretation

An M-Component must meet the requirement as stated except for the words ''normally denied under the ... discretionary security policy,'' which are not applicable to an M-Component.

Applicable Features

The security mechanisms of the VSLAN were tested by security testing and penetration testing. (see page 58, "Team Testing"). Testing was conducted on an operational VSLAN containing NSDs of each type of supported bus architecture. Both penetration and security testing were designed only to test the VSLAN NTCB and were not intended to exercise security services provided by the host computers. Complete end-to-end testing of this sort is beyond the scope of the evaluation of the VSLAN B2 MDIA network component.

Security testing was designed to test the enforcement of mandatory access controls, discretionary access controls, labeling, identification and authentication, object reuse, and auditing. The team's tests were based largely on the functional tests developed by Verdix and described in their test documentation [3]. These tests comprised both manual and automated means in order to more completely exercise the VSLAN's capabilities. These tests exercised all aspects of the security policy implemented by the VSLAN and needed to be supplemented with only a few team-developed tests. Neither the Verdix test suite nor the team-developed tests uncovered any significant flaws; minor flaws were corrected. The security tests performed demonstrated that the VSLAN functioned as described in the DTLS.

The evaluation team based its penetration testing upon a flaw hypothesis examination of the design documentation and source code of the VSLAN. Penetration tests uncovered only minor flaws in the implementation of the VSLAN software which were subsequently corrected and the corrected version was examined by the evaluation team. No design flaws were found.

Conclusion

The VSLAN satisfies the B2 Security Testing requirement.

Design Specification and Verification Requirement

A formal model of the security policy supported by the TCB shall be maintained over the life cycle of the ADP system that is proven consistent with its axioms. A descriptive top-level specification (DTLS) of the TCB shall be maintained that completely and accurately describes the TCB in terms of exceptions, error messages, and effects. It shall be shown to be an accurate description of the TCB interface.

Interpretation

The overall network security policy expressed in this model will provide the basis for the mandatory access control policy exercised by the NTCB over subjects and storage objects in the entire network. The policy will also be the basis for the discretionary access control policy exercised by the NTCB to control access of named users to named objects. Data integrity requirements addressing the effects of unauthorized MSM need not be included in this model. The overall network policy must be decomposed into policy elements that are allocated to appropriate components and used as the basis for the security policy model for those components.

The level of abstraction of the model, and the set of subjects and objects that are explicitly represented in the model, will be affected by the NTCB partitioning. Subjects and objects must be represented explicitly in the model for the partition if there is some network component whose NTCB partition exercises access control over them. The model shall be structured so that the axioms and entities applicable to individual network components are manifest. Global network policy elements that are allocated to components shall be represented by the model for that component.

The requirements for a network DTLS are given in the design documentation section.

Additional Network Component Interpretation

An M-Component must meet the requirement as stated.

Security Policy is interpreted to mean the MAC Policy supported by the component. Model is interpreted to be those portions of a reference monitor model that are relevant to the MAC Policy supported by the Component (e.g., the representation of the current access set and the sensitivity labels of subjects and objects, and the Simple Security and Confinement Properties of the Bell and La Padula Model).

Applicable Features

The Formal Security Policy Model (FSPM) supported by the VSLAN [21] is described earlier in this report (see page 47, "Formal Security Policy Model"). The VSLAN FSPM is an access control security model that elaborates the connection-oriented abstraction. It is an external model covering the VSLAN communications phenomena and security requirements imposed on them. It is formally based on the Bell-La Padula model. The differences between the VSLAN FSPM and BLP are due to the nature of the VSLAN as a network and have been discussed earlier. The VSLAN FSPM is a satisfactory security model for the VSLAN. The requisite security properties have been proven for the VSLAN FSPM using the Gypsy Verification Environment [33]. The VSLAN FSPM has been maintained over the life cycle of the development of the VSLAN.

The Descriptive Top Level Specification (DTLS) is contained in the NSD external interface specification [11], which defines the interface between the VSLAN and the external host systems, and the NSC software specification [9], which defines the security officers' interface to the VSLAN. Additionally, the trusted facility manual [16] provides further details about the security officers' interface. The DTLS is an accurate description of the NTCB interface; it completely and accurately describes the NTCB in terms of exceptions, error messages, and effects.

Conclusion

The VSLAN satisfies the B2 Design Specification and Verificatio: requirement.

Configuration Management Requirement

During development and maintenance of the TCB, a configuration management system shall be in place that maintains control of changes to the descriptive top-level specification, other design data, implementation documentation, source code, the running version of the object code, and test fixtures and documentation. The configuration management system shall assure a consistent mapping among all documentation and code associated with the current version of the TCB. Tools shall be provided for generation of a new version of the TCB from source code. Also available shall be tools for comparing a newly generated version with the previous TCB version in order to ascertain that only the intended changes have been made in the code that will actually be used as the new version of the TCB.

Interpretation

The requirement applies as written, with the following extensions:

1. A configuration management system must be in place for each NTCB partition.

2. A configuration management plan must exist for the entire system. If the configuration management system is made up of the conglomeration of the configuration management systems of the various NTCB partitions, then the configuration management plan must address the issue of how configuration control is applied to the system as a whole.

Applicable Features

Verdix has implemented a configuration management plan which maintains control of changes to a well-defined set of configuration items. This is described earlier (see page 54, "Configuration Management") in this report. The tools used to generate new versions of the NTCB from source code and to compare new and old versions of the source code are also described there.

Conclusion

The VSLAN satisfies the B2 Configuration Management requirement.

Security Features User's Guide Requirement

A single summary, chapter, or manual in user documentation shall describe the protection mechanisms provided by the TCB, guidelines on their use, and how they interact with one another.

Interpretation

This user documentation describes user visible protection mechanisms at the global (network system) level and at the user interface of each component, and the interaction among these.

Applicable Features

The information required to satisfy this requirement is in an appendix of the trusted facility manual [16]. Since there are no users who directly access the VSLAN, this appendix is targeted at individuals responsible for a particular host's communications over the network. The appendix describes the security functionality of the VSLAN and the principal's role within it. It covers the following topics: identification and authentication, security policy, MAC policy, DAC policy, security officer display, labeling, encryption, and audit and alarms. The authentication process is detailed and the need to guard one's Datakey is emphasized. The features the VSLAN implements beyond the principal's control (except authentication) are described with an emphasis on the manner in which they will affect the principal.

Conclusion

The VSLAN satisfies the B2 Security Features User's Guide requirement.

Trusted Facility Manual Requirement

A manual addressed to the ADP system administrator shall present cautions about functions and privileges that should be controlled when running a secure facility. The procedures for examining and maintaining the audit files as well as the detailed audit record structure for each type of audit event shall be given. The manual shall describe the operator and administrator functions related to security, to include changing the security characteristics of a user. It shall provide guidelines on the consistent and effective use of the protection features of the system, how they interact, how to securely generate a new TCB, and facility procedures, warnings, and privileges that need to be controlled in order to operate the facility in a secure manner. The TCB modules that contain the reference validation mechanism shall be identified. The procedures for secure generation of a new TCB from source after modification of any modules in the TCB shall be described.

Interpretation

This manual shall contain specifications and procedures to assist the system administrator(s) maintain cognizance of the network configuration. These specifications and procedures shall address the following:

1. The hardware configuration of the network itself;

2. The implications of attaching new components to the network;

3. The case where certain components may periodically leave the network (e.g., by crashing, or by being disconnected) and then rejoin;

4. Network configuration aspects that can impact the security of the network system; (For example, the manual should describe for the network system administrator the interconnections among components that are consistent with the overall network system architecture.)

5. Loading or modifying NTCB software or firmware (e.g., down-line loading).

6. Incremental updates; that is, it must explicitly indicate which components of the network may change without others also changing.

The physical and administrative environmental controls shall be specified. Any assumptions about security of a given network should be clearly stated (e.g., the fact that all communications links must be physically protected to a certain level).

The components of the network that form the NTCB must be identified. Furthermore, the modules within an NTCB partition that contain the reference validation mechanism (if any) within that partition must be identified.

The procedures for the secure generation of a new version (or copy) of each NTCB partition from source must be described. The procedures and requirements for the secure generation of the NTCB necessitated by changes in the network configuration shall be described.

Additional Network Component Interpretation

An [M-Component/D-Component/I-Component] must meet the requirement as stated except for the words "The procedures for examining and maintaining the audit files as well as..." These words are interpreted to mean "the mechanisms and protocols associated with exporting of audit data must be defined." Also, the words "...to include changing the security characteristics of a user", shall not be applicable to an M-Component.
Applicable Features

The information required to fulfill the trusted facility manual requirement is contained within a single document [16].

Section four ("Audit and Alarm Functions") of the trusted facility manual describes the audit mechanism in terms of auditable actions and the structure of audit records. Appendix D describes the post-processing tool used to view audit records. The maintenance of audit files is described under the section on operator functions.

Security Operator and Security Administrator functions are described in Sections eight and nine respectively. The procedure for using all VSLAN management functions are described within these sections. This includes the procedures for altering the configuration of the network by adding additional NSDs to the network.

Sections two ("Overview of Security Mechanisms") and ten ("Facility Warnings") provide the network administrator with caveats for secure operations. These include a description of:

- possible hardware configurations of the VSLAN

- procedures for and implications of adding new components to the network configuration of the VSLAN in a secure manner

- the effects of hosts periodically leaving and rejoining the network as a result of principal activity

- the inter-relationships of network components.

The TCB modules which implement the reference validation mechanism are described within section five of the document. This section also includes further descriptions of the hardware configuration of the network and a discussion of implications of adding new components to the network. Additionally, the secure generation of a new NTCB is described.

Conclusion

The VSLAN satisfies the B2 Trusted Facility Manual requirement.

Test Documentation Requirement

The system developer shall provide to the evaluators a document that describes the test plan, test procedures that show how the security mechanisms were tested, and results of the security mechanisms' functional testing. It shall include results of testing the effectiveness of the methods used to reduce covert channel band widths.

Interpretation

The "system developer" is interpreted as "the network system sponsor". The description of the test plan should establish the context in which the testing was or should be conducted. The description should identify any additional test components that are not part of the system being evaluated. This includes a description of the test-relevant functions of such test components and a description of the interfacing of those test components to the system being evaluated. The description of the test plan should also demonstrate that the tests adequately cover the network security policy. The tests should include the features described in the System Architecture and the System Integrity sections. The tests should also include network configuration and sizing.

Applicable Features

Verdix compiled a test plan consisting of 25 test scenarios, which has been documented in the test plan [3], and has carried out the tests, the details of which have been documented [20]. See the earlier discussion (page 57, "Vendor Testing") for more information.
Conclusion

The VSLAN satisfies the B2 Test Documentation requirement.

Design Documentation Requirement

Documentation shall be available that provides a description of the manufacturer's philosophy of protection and an explanation of how this philosophy is translated into the TCB. The interfaces between the TCB modules shall be described. A formal description of the security policy model enforced by the TCB shall be proven to show that it is sufficient to enforce the security policy. The specific TCB protection mechanisms shall be identified and an explanation given to show that they satisfy the model. The descriptive top-level specification (DTLS) shall be shown to be an accurate description of the TCB interface. Documentation shall describe how the TCB implements the reference monitor concept and give an explanation why it is tamper resistant, cannot be bypassed, and is correctly implemented. Documentation shall describe how the TCB is structured to facilitate testing and to enforce least privilege. This documentation shall also present the results of the covert channel analysis and the tradeoffs involved in restricting the channels. All auditable events that may be used in the exploitation of known covert storage channels shall be identified. The band widths of known covert storage channels, the use of which is not detectable by the auditing mechanisms, shall be provided. (See the Covert Channel Guideline section.)

Interpretation

Explanation of how the sponsor's philosophy of protection is translated into the NTCB shall include a description of how the NTCB is partitioned. The security policy also shall be stated. The description of the interfaces between the NTCB modules shall include the interface(s) between NTCB partitions and modules within the partitions if the modules exist. The sponsor shall describe the security architecture and design, including the allocation of security requirements among components.

The documentation includes both a system description and a set of component DTLSs. The system description addresses the network security architecture and design by specifying the types of components in the network, which ones are trusted, and in what way they must cooperate to support network security objectives. A component DTLS shall be provided for each trusted network component, i.e., each component containing an NTCB partition. Each component DTLS shall describe the interface to the NTCB partition of its component. Appendix A addresses component evaluation issues.

As stated in the introduction to Division B, the sponsor must demonstrate that the NTCB employs the reference monitor concept. The security policy model must be a model for a reference monitor.

The security policy model for each partition implementing a reference monitor shall fully represent the access control policy supported by the partition, including the discretionary and mandatory security policy for secrecy and/or integrity. For the mandatory policy the single dominance relation for sensitivity labels, including secrecy and/or integrity components, shall be precisely defined.

Additional Network Component Interpretation

All components must meet the requirement as stated. In addition:

> -- the Design Documentation must include a description of the protocol used by the D-Component to communicate Subject permissions (i.e., user ids), where applicable, with other components. This protocol must be shown to be sufficient to support the DAC policy enforced by the D-Component.
>
> -- the Design Documentation must include a description of the protocol used by the I-Component to export Authenticated Subject Identifiers to other components.

-- the Design Documentation must include a description of the protocol used by the A-Component to import Audit Data from other nodes.

Applicable Features

Verdix has supplied a statement of the philosophy of protection for the VSLAN in the trusted facility manual [16] and this is elaborated in the system specification [6]. In addition, Verdix has provided a network security architecture document [19] that augments the information in the system specification and describes network protection mechanisms provided by the VSLAN.

Communication among tasks and between the subsystems (NSC and NSD) is by message passing. The NSD software specification [8] provides details of the interactions within the NSD, and the NSC software specification [9] gives details of the actions within the NSC. The NSD and the NSC software specification also explain the processing performed by the NSD and NSC with respect to the mediation process. The network control interface specification gives details of the method of interaction between the NSDs and the NSC.

The NSD external interface specification [11] defines the interface between the VSLAN and the external host systems. The NSC software specification defines the security officers' interface to the VSLAN. These two documents constitute the Descriptive Top-Level Specification (DTLS). Additionally, the trusted facility manual [16] provides further details about the security officers' interface.

The system specification, NSD hardware specification [7], and the NSD software specification provide the documentation that explains how the VSLAN implements a network reference monitor, is tamper resistant, cannot be bypassed, and is correctly implemented.

A formal security policy model has been provided and relevant security theorems have been proven using the Gypsy Verification Environment. In addition, Verdix has supplied a covert channel analysis report [17] that discusses all covert storage channels that were found. This report describes the methodology that was followed to determine what covert storage channels exist and provides engineering estimates on their band width. This report also documents covert timing channels that were found, though this documentation is not required for the B2 MDIA rating.

Conclusion

The VSLAN satisfies the B2 Design Documentation requirement.

EVALUATION OF PART II REQUIREMENTS

There are nine network security services stated in Part II. These services are divided into three service groups. The service groups and their associated services are:

Communications Integrity
- Authentication
- Communications Field Integrity
- Non-Repudiation

Denial of Service
- Continuity of Operations
- Protocol-Based Protection Mechanisms
- Network Management

Compromise Protection
- Data Confidentiality
- Traffic Flow Confidentiality
- Selective Routing

The security perimeter of the NTCB partition includes the NSDs, NSC, and LAN cable, but excludes the host/NSD shared memory. The VSLAN can be used for classified multilevel secure LAN applications provided that the VSLAN is suitably protected (e.g., physically protected) at the highest classification of data handled by the VSLAN. The authentication and data confidentiality services are implicitly present for communications among the NSDs via the use of DES. However, their evaluation is beyond the scope of the Trusted Product Evaluation Program. The non-repudiation and traffic flow confidentiality services are not offered by the VSLAN. Since the VSLAN is IEEE 802.3 based, the selective routing service is not applicable due to the operating characteristic of local area networks.

A summary of the Part II ratings are shown on the following page in Figure 11.

COMMUNICATIONS INTEGRITY	
AUTHENTICATION	(see report)
COMMUNICATIONS FIELD INTEGRITY	
functionality	GOOD
strength	GOOD
assurance	GOOD
NON-REPUDIATION	(not offered)
DENIAL OF SERVICE	
CONTINUITY OF OPERATIONS	
functionality	MINIMUM
strength	MINIMUM
assurance	GOOD
PROTOCOL-BASED PROTECTION MECHANISMS	
functionality	FAIR
strength	GOOD
assurance	GOOD
NETWORK MANAGEMENT	
functionality	PRESENT
strength	FAIR
assurance	GOOD
COMPROMISE PROTECTION	
DATA CONFIDENTIALITY	(see report)
TRAFFIC FLOW CONFIDENTIALITY	(not offered)
SELECTIVE ROUTING	(not offered)

Figure 11 Summary of Part II Ratings

Communications Integrity

Communications integrity is a collective term for services which are concerned with the accuracy, faithfulness, non-corruptibility, and believability of information transfer between peer entities through the computer communications network. The services are authentication, communications field integrity, and non-repudiation.

Authentication

The network should ensure that a data exchange is established with the addressed peer entity (and not with an entity attempting a masquerade or a replay of a previous establishment). The network should assure that the data source is the one claimed.

Authentication is implicitly present, because the VSLAN uses the NIST-approved DES algorithm to encrypt datagrams and cryptographically bind the addressing and sequencing of datagrams. The strength of authentication is based on the strength of the DES algorithm and implementation. DES is not evaluated because it is outside of the scope of the Trusted Product Evaluation Program.

Communications Field Integrity

Communications field integrity refers to the prevention of unauthorized modification of any of the fields (e.g., user-data or header fields) involved in communications. The network should ensure that information is accurately transmitted from source to destination. The network should be able to counter equipment failure as well as actions by persons and processes not authorized to alter the data.

For Ethernet-level communications among the NSDs (including the NSD-Prime), the VSLAN provides sequence numbers for packets transmitted between each source and destination pair and uniquely encrypts packets for each source and destination pair. For communications between each NSD and NSD-Prime a unique master encryption key is used. For communications from one NSD to another NSD, a unique working encryption key is used per session (i.e., for an active association).

The Ethernet protocol uses a 32-bit frame check sequence on the IEEE 802.3. In addition to this, the NSD software piaces a 16-bit checksum on the datagram enclosed within each IEEE 802.3 packet, and then encrypts the datagram. The datagram also contains a 16-bit header checksum over the sequence number, source and destination NSDs, source principal ID, service access point, and length of data field. The security label is protected with an additional 16-bit checksum.

For NSD-to-NSD connections, the VSLAN can detect the modification, insertion, deletion, or replay of a packet. Since the VSLAN provides a datagram communications service to the associated hosts, it is not meaningful to provide datagram (or packet) level recovery. The VSLAN can also detect the alteration of certain fields (e.g., security label), and reports to the granularity of header or datagram level.

For NSC-to-NSD connections, the Network Control Protocol provides recovery (retransmission and acknowledgement) in addition to the detection capabilities provided above.

In accordance with the guideline provided within the TNI, the VSLAN warrants a rating of GOOD for the functionality concerning communications field integrity service.

The VSLAN also provides capabilities, which are not applicable with respect to the above rating, because unauthorized insertion, deletion, or replay of a packet are not applicable when the VSLAN is used in a classified, operational environment, which must be physically controlled.

It has been indicated already that datagrams have sequence numbers and are encrypted before transmission. If the sequence number of a received packet is outside the expected range of sequence numbers, a playback attempt may have occurred. If a recording is made of an earlier session, it would not be decrypted properly, because the key would be different. Detection of such an attempt is audited. If a recording were made earlier in the same session (i.e., with the correct keys), the sequence number would be outside the range of acceptability, and this would cause an audit of the packet. The packet is not rejected, and the current sequence number is reset to the sequence number in the packet.

The strength of the VSLAN communications field integrity service is based upon the strength of the ciphers, the provision of cryptographically bound checksums, the provision of non-cryptographically bound cyclic redundancy checks, the correctness of the protocol logic, and the adequacy of implementation.

Although the use of cryptography adds to the ability of the VSLAN to detect message stream modification, the evaluation team believes that the VSLAN deserves a strength rating of GOOD based solely on the use of the CRC and packet header checksums, described above.

The assurance rating of GOOD for the VSLAN communications field integrity is derived from the B2 assurance rating of the VSLAN component, as well as from the structured design, security testing, configuration management, and distribution factors as described in the TNI.

Non-Repudiation

The feature of non-repudiation provides unforgeable proof that a message was both sent and received. This prevents the receiver from falsely claiming that a message was never received when it actually was, as well as preventing the sender from falsely claiming that a message was sent when it actually was not.

The VSLAN does not offer non-repudiation of receipt or sending of messages.

Denial of Service

A denial-of-service condition is defined to exist whenever the throughput falls below a pre-established threshold, or when access to a remote entity is unavailable. Denial of service also exists when resources are not available to users on an equitable basis. Denial of service can be caused by software errors, hardware failures, or malicious attacks. If a connection is active, denial-of-service condition can be detected by the maximum waiting time or the predetermined minimum throughput. However, when a connection is quiescent, a protocol entity is unable to detect a denial-of-service attack that completely cuts off the flow of packets.

The services are continuity of operation, protocol-based protection, and network management.

Continuity of Operations

Continuity of operations involves how well a system provides a means to detect denial of service and notify the network manager.

In the VSLAN, if a given host or NSD fails or is subject to a denial of service attack, the remaining NSDs will continue to operate normally. The VSLAN will reconfigure itself when a new (or replacement) NSD is added to the network. When an NSD is initialized with a valid Datakey, it is loaded with all of the parameters necessary for operation. Therefore, a faulty NSD can be replaced, and the replacement NSD can be initialized, without affecting the remainder of the network.

When the NSC (or NSD-Prime) fails, it must be shut down. This will result in all of the NSDs shutting down. Also, if an NSC is under a denial-of-service attack, its throughput may drop below the acceptable limits. This decreased throughput will be detected by the NSC program and reported to the security officer.

Because failure of the NSC will result in the entire network shutting down, the VSLAN earns a functionality rating of MINIMUM for continuity of operations.

The strength rating of MINIMUM of the continuity of operations service is based upon simulated testing of the VSLAN and its internal devices. The TNI enumerates several recommended functions and suggests rating the strength of each of them. An overall rating of strength can then be obtained by averaging the individual strength ratings. These ratings are:

Function	NSC Strength	NSD Strength
Replacement and Redundancy	NONE	FAIR
Reconfigurration	NONE	FAIR
Fault Tolerance	NONE	NONE
Security Controls	GOOD	GOOD
Overall Rating	MINIMUM	FAIR

The overall strength rating would then be MINIMUM for the NSC, which is the rating for this capability, because the NSC rating is weaker than that of the NSD.

The assurance rating of GOOD for the VSLAN continuity of operations is derived from the B2 assurance rating of the VSLAN component, as well as from performance analysis of the VSLAN which demonstrates that a single host system cannot deny service to any other host on the network.

Protocol-Based Protection Mechanisms

These mechanisms insure that there are no hardware/software incompatibilities which would result in denial of service by creating an undefined state. These incompatibilities include such situations as two devices that continually respond to one another's response, thereby tying up the network facilities.

The VSLAN uses three protocol-based mechanisms for protection against denial of service:

- A status poll message is generated periodically from the NSC to all NSDs. If a given NSD does not respond, indicating that it is disconnected or failed, the LLC software in the NSD-Prime will time-out and send a negative acknowledgement to the NSC.

- The status poll is used by each NSD to determine the correct functioning of the NSC. If an NSD is not polled within a predetermined time interval, indicating that the NSC is unable to maintain its audit collection and network control functions, it will then shut itself down.

- Each NSD identifies its status (on-line, off-line, or suspended) to the attached host. Higher-level protocols on the attached host can then use this status to add or remove connections, or to initiate testing of the connection.

Part II of the TNI suggests that the functionality of the protocol-based protection mechanisms can be evaluated as a function of the number of protocols provided. The protocols mentioned above would thereby warrant a functionality rating of FAIR.

The strength rating of GOOD for the VSLAN protocol-based protection mechanisms is based upon testing (with simulated loading), and analysis of the network control protocols.

The assurance rating of GOOD for the VSLAN protocol-based protection mechanisms is derived from the B2 assurance rating of the VSLAN component, as well as from performance analysis of the VSLAN which demonstrates that a single host cannot deny service to the remaining hosts on the VSLAN.

Network Management

Network management takes into consideration aspects of denial of service not covered by continuity of operations or protocol-based protection. These include capacity overloading, network flooding, or protocol retry resulting from excessive noise on the channel, all of which result in denial of service. The services provided by the VSLAN include:

- detection of status (i.e., on-line, off-line) changes of an individual NSD.

- auditing the occurrence of certain packet integrity errors, and the generation of alarms when the number of exceptions exceeds a predetermined threshold.

- shutting down or suspending a specific node that is suspected of creating problems on the network.

These services warrant the functionality rating of PRESENT.

The strength rating of FAIR for the VSLAN network management service is based upon testing, including simulated loading, and analysis of the network control protocols.

The assurance rating of GOOD for the VSLAN network management service is derived from the B2 assurance rating of the VSLAN component, as well as from performance analysis of the VSLAN.

Compromise Protection

Compromise protection is a collective term for a number of security services. These services are all concerned with the secrecy or non-disclosure of an information transfer between peer entities through the computer communications network. Physical security, such as protected wireways, can also provide transmission security. The network manager or sponsor must decide on the balance between physical, administrative, and technical security. The TNI addresses only technical security. Compromise protection services are data confidentiality, traffic flow confidentiality, and selective routing.

Data Confidentiality

Data confidentiality is primarily concerned with the resistance of data compromise to passive wiretapping attacks (obtaining data by observing it as it passes through a communication link). The design of the VSLAN prevents any host from receiving any packets other than those intended for that host.

Data confidentiality is implicitly present because datagrams are encrypted using NIST-approved DES algorithm. The strength of data confidentiality is based on the strength of the DES algorithm and implementation. DES is not evaluated, because it is outside of the scope of Trusted Product Evaluation Program.

Traffic Flow Confidentiality

Traffic flow confidentiality is concerned with the disclosure of information about the data being transmitted (for purposes of traffic analysis). Such information would include characteristics such as message length, transmission frequency, and source and destination addresses.

Traffic flow confidentiality is not offered by the VSLAN.

Selective Routing

Selective routing is a defensive measure wherein specific transmission routes are selected or avoided, the purpose of which is to avoid attacks. This service is helpful in that, if a breach of security is discovered, then traffic need not be transmitted through a vulnerable area.

SeJective routing is not offered by the VSLAN, because of the broadcast nature of local area networks.

EVALUATED HARDWARE COMPONENTS

The VERDIX Secure Local Area Network (VSLAN-200) consists of a variety of user-selected components, each installed with a Network Security Device communications board, all interconnected with user-supplied coaxial cable.

The Subsystems model numbers are given below.

PART NUMBER	DESCRIPTION
VNSC-200-286	VERDIX Network Security Center Includes COMPAQ 286 with 640 KB memory; one high density (1.2MB) 5.25" diskette drive; one 40 MB hard disk; one VNSD-PC-200 Network Security Device; one color terminal; one 80 column medium speed dot matrix printer; network security center management and control software. Includes Datakey hardware device.
VNSC-200-286e	VERDIX Network Security Center Includes COMPAQ 286e with 1 MB memory(1); one high density (1.2MB) 5.25" diskette drive; one 40 MB hard disk; one VNSD-PC-200 Network Security Device; one color terminal; one 80 column medium speed dot matrix printer; network security center management and control software. Includes Datakey hardware device. NOTE: One VNSC Required Per VSLAN-200 Configuration.
VNSD-PC-200	VERDIX Network Security Device Single board Network Security Device for use in IBM PC or IBM PC AT bus systems. Includes Datakey hardware device.
NSD-MB-200	VERDIX Network Security Device Single Board Network Security Device for use in Multibus-I (IEEE P796) systems. Includes Datakey hardware device.
NSD-QB-200	VERDIX Network Security Device Single Board Network Security Device for use in Q-Bus systems. Includes Datakey hardware device.

NSD-VME-200	VERDIX Network Security Device Single Board Network Security Device for use in VME Bus (IEEE P1014) systems. Includes Datakey hardware device.
NSD-NB-200	VERDIX Network Security Device Single Board Network Security Device for use in NuBus (IEEE 1196) systems. Includes Datakey hardware device.
NSD-3B2-200	VERDIX Network Security Device Single Board Network Security Device for use in 3B2-Bus systems. Includes Datakey hardware device.
KCPKA16KS	Datakey Keyceptacle Interface Connects to VNSC and VNSD
SLET-10	Ethernet Transceiver Unit For connection of Ethernet Transceiver Cable to Ethernet Media Cable.
SLEC-10	Ethernet Transceiver Cable For connecting Transceiver Unit to VNSD-PC-200 or VNSD-NB-200.
SLEC-12	Ethernet Transceiver Cable For connecting Transceiver Unit to VNSD-MB-200, VNSD-QB-200, VNSD-VME-200, or VNSD-3B2-200.
(User supplied)	75-ohm Coaxial cable

EVALUATED SOFTWARE COMPONENTS

The evaluated software is the VSLAN Version 5.0. This comprises:

NSC, Version 5.0
NSD, Version 5.0
NSD-Prime, Version 5.0
VOS, Version 5.0

Additional software not included in the NTCB, but which is part of a complete VSLAN package:

VAPP, the VSLAN audit post-processing tool, running under MS-DOS, version 3.0

This page intentionally left blank.

INTEL 80286 HARDWARE OVERVIEW

Architectural Overview

The Intel 80286, also known as the iAPX 286, is a general purpose microprocessor. It supports a 24-bit address bus and a 16-bit data bus, both internal and external.

Registers

The Intel 80286 has fifteen 16-bit registers (AX, BX, CX, DX, BP, SI, DI, SP, F, IP, MSW, CS, DS, SS, ES), which may be grouped into three categories: general registers, status and control registers, and segment registers. Also programmer-visible are several 32-bit registers, including the local descriptor table register, the global descriptor table register, and the interrupt descriptor table register (described later).

General Registers - There are eight general purpose registers (AX, BX, CX, DX, BP, SI, DI, SP) which are used to contain the operands of arithmetic and logic operations. Of these, four (AX, BX, CX, DX) may be used either as 16-bit registers, or may be split into pairs of individually-addressable 8-bit registers. The 8-bit registers are referenced by the byte (low or high) which they occupy in the respective 16-bit register.

Some of the general registers are also committed to specific use by certain instructions and addressing modes, in which case they are referred to as "base and index registers". For example, BX and BP are used to contain the base address; SI and DI are often used to contain the index value. SP contains the stack pointer.

Status and Control Registers - There are three registers which maintain the current state of the processor: the F register contains the flags; the IP, the instruction pointer; and MSW, the machine status word.

Segment Registers - The four segment registers are used to select the segments of memory that are addressable for code, stack, and data. They are: the CS, code segment selector; DS, data segment selector; SS, stack segment selector; and ES, extra segment selector (usually used for data).

Modes of Operation

The Intel 80286 supports two modes of operation: Real Address Mode and Protected Virtual Address Mode (PVAM). In the VSLAN, the Real Address Mode is used only for initialization. With the Intel 80286 in Real Address Mode, any task running on the processor has full and unrestricted access to the entire 1 megabyte address space. The PVAM, however, expands the address space to 16 megabytes and provides protection for memory partitions within that address space.

The processor always starts operating in Real Address Mode. Thereafter, a single microprocessor instruction, LMSW (Load Machine Status Word), can switch the processor into PVAM, and then only a hardware reset can cause a switch back to Real Address Mode. Hence, the rest of this discussion describes the PVAM of operation.

The Intel 80286, in PVAM, supports a set of four hierarchical privilege levels (rings), memory access mediation through a central mechanism (virtual memory), and task separation mechanisms.

List of Terms

The following terms are used throughout this discussion:

Alias	Alternate descriptor, for a segment, with different segment attributes.
Descriptor	Structure used to define a memory segment.
Descriptor Table	Memory resident structure used to store descriptors.
Gate	Special descriptor, used in transferring control.
Interrupt	Break in normal task execution.
Privilege Level (ring)	Hierarchical domain of privilege.
Segment	Finest granularity of memory separation, described by descriptors.
Selector	Used to choose a descriptor.
Task	Single thread of execution.
Trap (exception)	Similar to interrupt, occur when instruction fails to complete normally.

Security Features Segments

The Intel 80286 microprocessor views memory as a collection of segments, which may be defined to overlap each other. Each segment can be from 1 byte to 64 kilobytes in size and can reside anywhere within the 16 megabyte memory address space. Each segment is associated with a segment descriptor, whose inclusion in a descriptor table represents the presence of that segment in the address space defined by that table (see also Descriptors).

Address Translation

Memory addressing is accomplished by the use of 32-bit pointers, each composed of a 16-bit selector field and a 16-bit offset field. In Real Address Mode the 16-bit selector field contains the upper 16 bits of the 20-bit segment address, the lower four bits being zero; while the 16-bit offset field contains the lower 16 bits of the 20-bit offset address, with the upper four bits always zero. The 20-bit physical address is then calculated by adding together these two addresses.

Whereas the Real Address Mode selector field represents the high-order 16 bits of a 20-bit real memory address, in PVAM it represents a 16-bit index into a memory-resident segment descriptor table. The table entry, called a descriptor, contains the 24-bit segment base address and segment length. This base address is then added to the contents of the offset field of the pointer to result in the translation to the indicated physical memory address location. Each segment can be accessed only through a descriptor.

Descriptors

In addition to referencing a memory segment location, the descriptor also contains access control information. This includes a descriptor privilege level (DPL), segment type, access, and segment specific information (e.g., stack expansion direction).

The DPL assigns one of the four Intel 80286 privilege levels (discussed later) to the given descriptor. Because a segment is accessible only through its descriptor, this privilege level can conceptually be associated with the segment as well. However, the fact that one descriptor can refer to a segment (at one privilege level) does not imply that another descriptor cannot refer to another segment (at another privilege level), occupying the same memory space.

The descriptor will define its associated segment as being one of the following types: data segment, code segment, special system segment, or gate. A data segment is used as the operand of instructions, and may have the following access permissions: read only or read/write. The descriptor also defines whether it will expand up or down in the memory address space. This is useful when a data segment is to be used for a stack. A code segment is used as the source of instructions to be executed, and may have the following access permissions: execute or read/execute. The descriptor also defines whether it will be a conforming code segment or not. A conforming code segment is one that acquires the privilege level of the calling task if the task has less privilege. Special system segments are Task State Segments (TSS) and Local Descriptor Tables (LDT). Each will be described in more detail below. Gates are special descriptors used for transferring control indirectly. There are four types of gates: call gate, trap gate, interrupt gate, and task gate.

- Call gate: contains the DPL, base address, and index for a code segment. The index is used to set up an entry point.

- Trap gate: identical to call gate, except that it is used in the Interrupt Descriptor Table (IDT) to specify a trap service routine.
- Int. gate: identical to a trap gate, except that when it is invoked, interrupts are automatically disabled.

- Task gate: contains a descriptor referencing a TSS. Hence use of this gate cause a complete context switch.

All descriptors are physically located in memory resident tables, each a segment itself. The Intel 80286 recognizes three types of descriptor tables: LDT, Global Descriptor Table (GDT), and IDT. LDTs are used for task local data and there can be virtually any number of these in memory, but only one associated with any given task. Multiple tasks may share a common LDT in order that they may share a semi-private, as opposed to completely global, data set. There is exactly one GDT defined; it is used for system-wide, global data.

Descriptor tables are referenced by the Global Descriptor Table Register (GDTR) and Local Descriptor Table Register (LDTR). The GDTR describes the location and size (ie, number of entries) of the GDT, while the LDTR describes the location and size of the LDT associated with the currently running task. The contents of the LDTR change whenever a task switch occurs, so that it is loaded with the address and size of the new task's LDT. The contents of the GDTR can be modified by use of the privileged instructions Load GDT (LGDT) and Store GDT (SGDT). Similarly, the LDTR's contents can be modified by use of the privileged instructions Load LDT (LLDT) and Store LDT (SLDT).

The IDT is used to store interrupt vectors in the form of gates. (These will be discussed further in the section on Interrupts.) Up to 256 such gates may exist in this table and there can be only one such table known tc the Intel 80286 at any given instant. The method of maintaining the Interrupt Descriptor Table is similar to the maintenance of the LDTs and GDT. The IDT is pointed to by the IDT Register (IDTR), which is loaded by the privileged instruction LIDT.

Selectors

Segment selectors are held in the segment registers and used by tasks to select which descriptor to use in order to reference a segment. The selectors are composed of an index, descriptor table selector bit, and a Requested Privilege Level (RPL). The descriptor table selector bit specifies whether to use the GDT or LDT. The index specifies which descriptor within that table to use. The RPL indicates the privilege level of the access request.

An executing task has access to four segment selector registers: Code Segment (CS), Data Segment (DS), Stack Segment (SS), and the Extra Segment (ES). Hence, a task has access to up to four segments at any given instant. When a selector is loaded, the selected descriptor is checked to ensure that it exists and is well formed. Also, the segment is checked to ensure that it is present. Instructions that load selectors into DS and ES must refer to a data segment descriptor or a readable code segment descriptor, and the privilege requirements must be met. Instructions that load a selector into SS must refer to a writable data descriptor. Control transfer is accomplished when a selector is loaded into the CS by a control transfer operation. A transfer can occur only if the operation which loads the selector references the correct type of descriptor. If an attempt is made to load an incorrect type of descriptor or if the privilege check fails, then a general protection exception will occur.

A general protection exception indicates the occurrence of a violation to privilege rules or usage rules. An interrupt handler reads in an error code which is pushed onto the stack after the return address. This error code identifies the sector which is involved, while the return address identifies the instruction which caused the exception. Although most exceptions are restartable after the cause of the exception has been removed, a restart is generally not attempted for general protection exceptions.

Privilege Levels

The Intel 80286 supports four hierarchical levels of privilege (rings), numbered 0 through 3 in decreasing level of privilege. One such level is associated with every task and descriptor, and hence with every segment, subject, and gate.

A set of privilege rules are enforced by the Intel 80286 and are mandatory in nature. They can be summarized as follows:

- Data can be accessed only from the same or a more privileged ring.

- Code can be executed (called) only from the same or a less privileged ring.

After all checks, including privilege access checks, descriptor access checks, and segment bounds checks, a selector may be loaded. Any further references to the same segment require further checks to ensure that the specific reference is still within the bounds of the segment and that no attempt to write to a read-only segment has been made.

Any violations will cause an interrupt to occur before any memory reference is made or any registers are modified. This ensures that the process state remains unchanged in case a restart may be possible.

Stacks

Stacks are referenced through the SS and Stack Pointer (SP), to provide an offset. The stack segment is merely a data segment, possibly configured to expand downward in memory address space.

When a user transfers control to a new (more privileged) ring, the SS:SP pair is placed on the (new) stack of the called task, which is at the new level. When a return from that level occurs, those values are restored, so the task resumes operation at the previous level.

Each privilege level has its own stack which is determined by the Current Privilege Level (CPL) of the task and the TSS (described below), which contains stack segment information for each ring 0-2. There is no need for any stack information to be predefined in this manner for ring 3, because of the fact that ring 3 is the least-privileged level and could not have been transferred into from a less privileged ring. Information can be passed from less-privileged to more-privileged rings, but not the other way. The information is placed onto the stack along with a counter indicating how much information was passed on the stack.

Tasks

A task, with respect to the Intel 80286, is conceptually a single thread of execution, which is defined by a TSS. The TSS defines a save area, for the Intel 80286 registers, and stacks for rings 0-2. It also contains a selector for the task LDT and a selector referencing the calling task (only in the event of a task switching interrupt).

At any given instant, the Intel 80286 Task Register is used to reference a descriptor referencing the currently-executing task's TSS and also indicating the CPL of the task by virtue of the descriptor's DPL. During a task switch (transfer of control) the static portion (stack pointers and LDT selector) of the TSS remains unchanged, but the dynamic portion (the register save area) is automatically updated with the current register contents. These registers are restored automatically when the task regains control.

Control Transfers

Control transfers can be made intra-segment, inter-segment, and inter-level.

Intra-segment control transfers can be done via CALL, JMP, and RET instructions. In the case of the CALL and JMP, only an offset is specified. This offset is used to calculate a new point of execution within the same segment.

Inter-segment control transfers can be done via CALL, JMP, and RET instructions. In the case of the CALL and JMP, either a code segment and offset are specified directly, or a call gate may be specified. In this case, the gate must be accessible (i.e., at the same or less-privileged level) and the code segment is assumed to be at the same level, thus accessible.

Inter-level control transfers may be done via CALL and RET instructions. In the case of the CALL, either a call or task gate must be referenced so that control can be transferred to the new level as dictated by the gate. In this case, the gate must be accessible (i.e., at the same or less-privilege level) and the code segment must also be accessible (i.e., at a greater privilege level). Hence, inter-level control transfers via the CALL instruction can only be to a more privileged ring. The RET instruction allows control to transfer in the other direction, but only after a successful CALL has been completed.

Interrupts and Traps

Interrupts and traps are special cases of control transfers. Interrupts may be internal or external and maskable or non-maskable. In the case of external interrupts, they are independent of the currently executing task. The Intel 80286 also provides the capability for software interrupts. A trap, as opposed to an interrupt, is generated when an instruction fails to complete normally.

Tasks available to service interrupts and traps can be configured to execute in the interrupted task's context, or to do a complete task switch before servicing. The choice is determined by whether the interrupt/trap service routine is referred to by a trap, interrupt, or task gate.
Up to 256 interrupts and traps may be defined to the Intel 80286. 32 of these are defined internally by Intel, the rest may be defined by the system. A gate corresponding to each interrupt/trap is contained within the IDT. The IDT is a variable length segment, but must be minimally large enough to support the 32 internally defined interrupts. It may then grow as needed by the system.

Just as with all code segments, the interrupt/trap service code segment must be at least as privileged as the interrupted code segment's privilege level. Use of trap or interrupt gates may cause the context of the interrupted routine to change, since they execute in the same context. However, use of a task gate leaves the interrupted task's context unchanged, since a complete context switch is made.

Aliasing

The term aliasing refers to the capability of creating more than one descriptor referring to a single segment. This allows a programmer to have access to a segment in multiple modes or even with multiple privilege levels.

Example: One descriptor indicates that read/write access is allowed to some data segment at privilege level 0. Another descriptor indicates that execute access is allowed to some code segment (occupying the same memory address space) at privilege level 1. In this instance, the actual code may be read and modified by ring 0 tasks, while it could only be executed by ring one tasks.

This feature must be used dynamically during system operation in order to create and modify descriptor tables. In order to create new LDTs, some task must have write access to the GDT in order to build the pointer. Thus, a descriptor referring to the GDT address space as a writable data segment must exist. Also, when initially creating the LDT (e.g., specifying its descriptor entries) it must also be accessible as a writable data segment. However, once it is built, the descriptor providing this capability may be destroyed.

This page intentionally left blank.

ACRONYM GLOSSARY

ADP	- Automatic Data Processing
AP	- Applications Process
BIOS	- Basic Input Output System
BLP	- Bell - La Padula
BSD	- Berkeley Standard Distribution
CA	- Channel Attention
CCS	- Conforming Code Segment
CM	- Configuration Management
CMOS	- Complementary Metal Oxide Semiconductor
CPL	- Current Privilege Level
CPU	- Central Processing Unit
CRC	- Cyclic Redundancy Check
CSMA/CD	- Carrier Sense Multiple Access/Collision Detection
DAC	- Discretionary Access Control
DCP	- Data Ciphering Processor
DES	- Data Encryption Standard
DMA	- Direct Memory Access
DoD	- Department of Defense
DPL	- Descriptor Privilege Level
DTLS	- Descriptive Top-Level Specification
EEPROM	- Electrically Erasable Programmable Read Only Memory
EP	- Encryption Protocol
EPL	- Evaluated Products List
EPROM	- Erasable Programmable Read Only Memory
EPT	- Encryption Protocol Task
FIFO	- First In First Out
FSPM	- Formal Security Policy Model
GDT	- Global Descriptor Table
GDTR	- Global Descriptor Table Register
ICB	- Interface Control Block
ICS	- Inter-task Communications Service
ID	- Identifier
IDT	- Interrupt Descriptor Table
IDTR	- Interrupt Descriptor Table Register
IEEE	- Institution of Electrical and Electronic Engineers
	IPAR - Initial Product Assessment Report
ISO	- International Standards Organization
IV	- Initialization Vector
I/O	- Input/Output
ITC	- Inter-tast Communications
KB	- Kilobyte
Kbps	- Kilobits per second
KDC	- Key Distribution Center

KGB	- Kilogigabyte
LAN	- Local Area Network
LDT	- Local Descriptor Table
LDTR	- Local Descriptor Table Register
LED	- Light Emitting Diode
LLC	- Logical Link Control
MAC	- Mandatory Access Control
MB	- Megabyte
MFDDC	- Multipurpose Fixed Disk Drive Controller
NAP	- NSC Applications Processes
NCP	- Network Control Protocol
NCSC	- National Computer Security Center
NIST	- National Institute of Standards and Technology
NSC	- Network Security Center
NSD	- Network Security Device
NSD-3B2	- Network Security Device - 3B2-bus
NSD-MB	- Network Security Device - Multibus
NSD-NB	- Network Security Device - Nu-Bus
NSD-PC	- Network Security Device - PC-bus
NSD-QB	- Network Security Device - Q-bus
NSD-VME	- Network Security Device - VME-bus
NTCB	- Network Trusted Computing Base
OSI	- Open Systems Interconnect
PDL	- Program Design Language
PE	- Protection Enable
PLA	- Programmable Logic Array
PROM	- Programmable Read Only Memory
PS	- Protocol Server
PVAM	- Protected Virtual Address Mode
PVCS	- Polytron Version Control System
RAM	- Random Access Memory
RCS	- Revision Control System
RMIT	- Receive Message Interpreter Task
ROM	- Read Only Memory
RPCT	- Receive Policy Control Task
RPL	- Requested Privilege Level
SA	- Security Administrator
SAP	- Service Access Point
SI	- Source Index
SIT	- System Initialization Task
SK	- Separation Kernel
SMT	- System Monitoring Task
SLAN	- Secure Local Area Network
SO	- Security Officer

SOAPT	- Security Officer and Audit Processing Task
SRM	- Shared Resource Matrix
TCB	- Trusted Computing Base
TCP	- Transmission Control Protocol
TCSEC	- Trusted Computer System Evaluation Criteria
TF	- Trusted Facility
TNI	- Trusted Network Interpretation
TPCT	- Transmit Policy Control Task
TSS	- Task State Segment
UDP	- User Datagram Protocol
UHF	- Ultra High Frequency
VAPP	- Verdix Audit Post-Processor
VGA	- Video Graphics Array
VME	- Versabus Module Eurocard
VOS	- Verdix Operating System
VSLAN	- Verdix Secure Local Area Network

This page intentionally left blank.

REFERENCES

[1] *Project Plan, Verdix Secure Local Area Network,* Version 2, February 13, 1987, document # PR-2001.

[2] *Configuration Management Plan, Verdix Secure Local Area Network,* Version 2, March 18, 1987, document # PR-2002.

[3] *Test Plan, Verdix Secure Local Area Network,* Version 7, April 6, 1990, document # PR-2003.

[4] *Training Plan, Verdix Secure Local Area Network,* Version 1, February 4, 1988, document # PR-2004.

[5] *Training Manual, Verdix Secure Local Area Network,* Version 1, March 22, 1989, document # PR-2006.

[6] *System Specification, Verdix Secure Local Area Network,* Version 4, August 23, 1989, document # SP-2001.

[7] *Network Security Device (NSD) Hardware Specification, Verdix Secure Local Area Network,* Version 4, August 30, 1989, document # SP-2002.

[8] *Network Security Device (NSD) Software Specification, Verdix Secure Local Area Network,* Version 7, August 25, 1989, document # SP-2003.

[9] *Network Security Center (NSC) Software Specification, Verdix Secure Local Area Network,* Version 7, August 28, 1989, document # SP-2004.

[10] *Network Security Device-Prime Software Specification, Verdix Secure Local Area Network,* Version 6, August 30, 1989, document # SP-2005.

[11] *Network Security Device (NSD) External Interface Specification, Verdix Secure Local Area Network,* Version 8, February 21, 1990, document # SP-2006.

[12] *Network Control Interface Control Document, Verdix Secure Local Area Network,* Version 7, April 3, 1990, document # SP-2007.

[13] *Separation Kernel External Specification (Includes Timer/Hardware Services) for Network Security Device, Verdix Secure Local Area Network,* Version 6, April 6, 1990, document # SP-2008.

[14] *NSC System Services (Includes Timer, Hardware, Video, Printer, Keyboard, and Disk Services) for Network Security Center, Verdix Secure Local Area Network,* Version 3, August 31, 1989, document SP-2009.

[15] *Network Security Center (NSC) Hardware Specification, Verdix Secure Local Area Network,* Version 4, February 21, 1990, document # SP-2010.

[16] *Trusted Facility Manual, Verdix Secure Local Area Network,* Version 6, May 23, 1990, document # TR-2002.

[17] *Covert Channel Analysis, Verdix Secure Local Area Network,* Version 4, February 14, 1990, document # TR-2005.

[18] *Network Security Device (NSD) Hardware Reference Manual, Verdix Secure Local Area Network,* Version 7, February 2, 1990, document # TR-2006.

[19] *Network Security Architecture Document, Verdix Secure Local Area Network,* Version 4, May 24, 1990, document # TR-2007.

[20] *Test Resuits, Verdix Secure Local Area Network,* October 2, 1989, document # TR-2008.

[21] *Secure Local Area Network, Design Verification Results,* Version 3, May 22, 1990, document # TR-0002.

[22] *Department of Defense, Trusted Computer System Evaluation Criteria,* December 1985, DOD 5200.28-STD.

[23] *Trusted Network Interpretation of the Trusted Computer System Evaluation Criteria,* 31 July 1987, NCSC-TG-005, Version 1, National Computer Security Center.

[24] *Technical Rationale Behind CSC-STD-003-85: Computer Security Requirements -- Guidance for Applying the Department of Defense Trusted Computer System Evaluation Criteria in Specific Environments,* 25 June 1985, CSC-STD-004-85.

[25] *International Standard, ISO 7498, Information processing systems - Open Systems Interconnection - Basic Reference Model,* Ref. No. ISO7498-1984(E).

[26] *Local Area Networks, ANSI/IEEE Standard Draft International Standard, 802.3, Carrier Sense Multiple Access with Collision Detection,* ANSI/IEEE Std 802.3-1985; ISO/DIS 8802/3.

[27] PLM-286 *User's Guide,* Intel Corporation, 1982, Order Number 121945-001.

[28] ASM286 *Assembly Language Reference Manual*, Intel Corporation, October, 1984, Order Number 121924-003.

[29] *Trusted Product Evaluations, A Guide For Vendors*, 1 March 1988, NCSC-TG-002, Version-1, National Computer Security Center.

[30] Kemmerer, R. A., *Shared Resource Matrix Methodology: An Approach to Identifying Storage and Timing Channels*, ACM Transactions on Computer Systems, vol. 1, no. 3, August 1983.

[31] Linde, R. H., *Operating system penetration, "Proceedings of the National Computer Conference*, 1975"

[32] Bell, D. E. and La Padula, L. J., *Secure Computer System: Unified Exposition and Multics Interpretation*, MTR-2997, The MITRE Corporation, Bedford, MA, July 1975.

[33] Good, D. I. et al, *Using the Gypsy Methodology, Institute for Computer Science*, June 6, 1984.

[34] *The Ethernet, A Local Area Network: Data Link Layer and Physical Layer Specifications*, Version 1.0, September 30, 1980, Digital Equipment Corporation, Intel Corporation, Xerox Corporation.

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

3. DISTRIBUTION/AVAILABILITY OF REPORT
UNLIMITED DISTRIBUTION

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
CSC-EPL-90/001

5 MONITORING ORGANIZATION REPORT NUMBER(S)
S235,898

6a. NAME OF PERFORMING ORGANIZATION
National Computer Security Center

6b. OFFICE SYMBOL *(If applicable)* C71

7a NAME OF MONITORING ORGANIZATION

6c. ADDRESS *(City, State and ZIP Code)*

9800 Savage Road
Ft. George G. Meade, MD 20755-6000

7b ADDRESS *(City, State and ZIP Code)*

8a. NAME OF FUNDING/SPONSORING ORGANIZATION

8b. OFFICE SYMBOL *(If applicable)*

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS *(City, State and ZIP Code)*

10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO	TASK NO.	WORK UNIT NO

11. TITLE *(Include Security Classification)*
Final Evaluation Report Verdix Corporation VSLAN 5.0

12. PERSONAL AUTHOR(S)
Thomas A. Ambrosi, Ronald J Bottomly, Paul A Olson, Shawn M Rovansek, Frank Belvin, Manilal Daya, Dale M Johnson, Jeremy E Dawson

13a. TYPE OF REPORT
Final

13b. TIME COVERED
FROM ____ TO __

14 DATE OF REPORT *(Yr/ Mo/ Day)*
90,07,25

15. PAGE COUNT
132

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB. GR.

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
NSA, Verdix Corporation, VSLAN 5.0, TCSEC, TNI, B2

19. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

The National Security Agency's (NSA) Trusted Product and Network Security Evaluations Division examined the security protection mechanisms provided by Verdix Corporation's VSLAN 5.0. It was evaluated against the *DoD Trusted Computer System Evaluation Criteria (TCSEC) and the Trusted Network Interpritation (TNI)* and the evaluation team determined that the system meets all criteria for the B2 level of trust.

This report documents the findings of the evaluation.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
UNCLASSIFIED/UNLIMITED

21 ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL
PATRICIA L. MORENO

22b TELEPHONE NUMBER *(Include Area Code)* (301)859-4458

8b OFFICE SYMBOL
C71

EDITION OF 1 JAN 73 IS OBSOLETE.