CSC-EPL-86/003

2

NATIONAL COMPUTER SECURITY CENTER

AD-A208 003

FINAL EVALUATION REPORT OF CONTROL DATA CORPORATION NETWORK OPERATING SYSTEM SECURITY EVALUATION PACKAGE

28 May 1986

Approved For Public Release:
Distribution Unlimited

FINAL EVALUATION REPORT

CONTROL DATA CORPORATION

NOS SECURITY EVALUATION PACKAGE

NATIONAL
COMPUTER SECURITY CENTER

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

May 28, 1986

CSC-EPL-86/003
Library No. S228,358

This page intentionally left blank.

CSC-EPL-86/003
Library No. S228,358

FOREWORD

This publication, the Final Evaluation Report, Control Data Corporation, NOS Security Evaluation Package, 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 CDC's NOS Security Evaluation Package operating system. The requirements stated in this report are taken from the DEPARTMENT OF DEFENSE TRUSTED COMPUTER SYSTEM EVALUATION CRITERIA dated 15 August 1983.

Approved:

Eliot Sohmer
Chief, Office of Product Evaluations
and Technical Guidelines
National Computer Security Center

May 28, 1986

ACKNOWLEDGEMENTS

Team Members

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

Michael W. Hale
Frank L. Mayer
Grant M. Wagner

National Computer Security Center
Fort George G. Meade, Maryland

Jeff Glass
Karina Hasler

MITRE Corporation
Bedford, Massachusetts

Jerzy W. Rub

The Aerospace Corporation
Los Angeles, California

W. Olin Sibert

Oxford Systems, Inc.
Arlington, Massachusetts

Further Acknowledgements

Acknowledgement is given to the following individuals for their contributions to this document: Paul Woodie, Tom Anderson, Kimberly Kirkpatrick, and David Wills.

CONTENTS

EXECUTIVE SUMMARY

The security protection provided by Control Data Corporation's (CDC) Network Operating System (NOS) Security Evaluation Package (see page C-1, "Evaluated Software") running on CDC CYBER 170 mode compatible machines (see page B-1, "Evaluated Hardware") has been evaluated by the National Computer Security Center (NCSC). The security features of NOS were evaluated against the requirements specified by the Department of Defense Trusted Computer System Evaluation Criteria (the Criteria) dated 15 August 1983.

The NCSC evaluation team has determined that the highest class at which NOS satisfies all the requirements of the Criteria is class C2 and therefore NOS has been assigned a class C2 rating.

A system that has been rated as being a C2 class system provides a more finely grained discretionary access control than C1 systems, making users individually accountable for their actions through login procedures, auditing of security-relevant events, and resource isolation. There is no assurance that a C division system is free of flaws that would allow the subversion or bypassing of the documented security mechanisms through penetration methods.

This page intentionally left blank.

INTRODUCTION

In March 1984, Control Data Corporation (CDC) requested that the NCSC evaluate their commercially available operating system, Network Operating System (NOS) Security Evaluation Package (see page C-1, "Evaluated Software") as run on CYBER 170 mode compatible machines (see page B-1, "Evaluated Hardware"). The objective of this product evaluation was to rate the NOS system against the DoD Trusted Computer System Evaluation Criteria (the Criteria), and to place it on the Evaluated Products List (EPL) with the final rating. This report presents the results of that evaluation.

This report is based on NOS Security Evaluation Package, consisting of NOS version 2.4.1, TMS, and the audit reduction tool, running in secured mode with the following subsystems available:

- Network Access Method (NAM)
- Batch I/O (BIO)
- Interactive Access Facility (IAF)
- Magnet (MAG)
- Remote Batch Facility (RBF)
- Tape Management System (TMS)

Material for this report was gathered by the NCSC NOS evaluation team through documentation review, interaction with system developers, and experience using the NOS 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 DoD Trusted Computer System Evaluation Criteria 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. Product evaluations consist 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 to 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 complete security systems. In addition, the release being evaluated must not undergo any additional 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 Criteria. 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 Criteria requirements in terms of both features and assurances. The final report and EPL entry are made public.

Document Organization

This report consists of five major sections. Section 2 provides an overview of the system hardware and software architecture. Section 3 provides the class C2 requirements, as stated in the Criteria, and describes the functions of the security features that resulted in NOS being assigned a class C2 rating. Section 4 describes NOS features that exceed the C2 requirements. Section 5 presents comments by the evaluation team concerning specific NOS features. Appendix A contains a list of the documents reviewed for this evaluation. Appendix B identifies the specific hardware components to which the evaluation applies. Appendix C identifies the specific software components to which the evaluation applies. Appendix D is a table of acronyms.

SYSTEM OVERVIEW

Subjects and Objects

This report will explain NOS in terms of subjects and objects. A subject is an active entity, which performs operations on passive entities, called objects. An object is an information repository. In NOS, the subjects are jobs; the objects are files and devices.

A job is a stream of commands and data. The commands inform NOS which programs are to be loaded into the job's address space and executed.

There are three kinds of files--temporary, permanent, and queued. Temporary files are files which are local to a job. They cannot be accessed by other jobs, and can be thought of as a logical extension to memory. Permanent files and queued files exist independently of any job. Permanent files provide the means for sharing information between jobs and users. Permanent files are discussed further on page 21, "Discretionary Access Control". Queued files are files which are destined for output devices, or are arriving from input devices.

The devices that are supported by NOS are card readers, line printers, plotters, card punches, disks, and tape drives. Jobs do not have direct access to devices, however. Tape drives are accessed through the trusted program MAG which is automatically invoked by NOS. All other devices are accessed through the trusted program BIO. Jobs communicate with BIO via queued files.

Hardware Architecture

This report covers the CDC Cyber 170 mode compatible machines (see page B-1, "Evaluated Hardware"). The hardware architecture is similar for all the machines. Each is a single unit that contains a Central Processing Unit (CPU), main memory, and some number of peripheral processors (PPs). The interface to extended memory is somewhat different for some of the models, however the same protection is offered in all models. When executing NOS, all the machines use the same instruction set. Additionally, for all 800-series models except the 865 and 875, which are repackaged versions of 170/700 series models, the CPUs have an additional operating mode, Native (CYBER 180) mode, in which they execute a completely different instruction set, primarily intended for use by NOS/Virtual Environment (NOS/VE), the follow-on operating system to NOS.

History of CDC Hardware

The hardware architecture has a long history. It was originally implemented in 1964 starting with the CDC 6400 and 6600 models. These machines provided the same basic instruction set, and had the same division of responsibility between the CPU which performed computation, and the PPs which handled I/O and all operating system tasks.

In 1971, the Cyber 70 series became available. In these models, the Exchange Jump mechanism was a standard feature, allowing some operating system functions to be performed by the CPU, but retaining complete program compatibility with earlier models.

In 1974, CDC released the Cyber 170 series, again implementing essentially the same visible architecture, the major differences being in hardware implementation and extended memory addressing. The original 170 series was supplanted in 1979 by the 170/700 machines, again program-compatible, but faster.

In 1982, the Cyber 170/800 series was released. These machines are program-compatible with all the earlier models, but, when using the additional instruction set, provide a path to upgrade to CDC's new operating system, NOS/VE. The major difference with respect to the 170/700 machines is the microcoded implementation of some, and yet another mechanism for extended memory addressing.

The major trends in the 20-year history are:

A) Larger and more easily accessible main and extended memory,
B) Faster CPUs for user programs, and
C) More operating system tasks performed in the CPU, rather than in the PPs.

The machines all remain program-compatible. A modern Cyber 180 845 can run an object program originally written on a CDC 6600.

Major Hardware Components

A CYBER 170 mode compatible system consists of a central unit connected to a set of peripherals. However, peripherals do not contain any security or protection mechanisms, they are only expected to function correctly (e.g., return only the requested sector), and therefore are not relevant to the report.

The central unit consists of one or two central processors (CPU), a central memory controller (CMC), central memory (CM), extended memory (EM) or in some 800-series machines, unified extended memory (UEM), and peripheral processors (PPs)(also called Peripheral Processing Units (PPUs)). Additionally, a system includes one or more network processor units (NPUs) for communications control. Depending on the model, the CPU may include multiple functional units and various amounts of cache memory. Particularly in the faster models, the CPUs are complex and include considerable lookahead and prefetch logic.

The entire central system is an integrated unit. Some models are field upgradable. In addition, it is possible to selectively disable parts of the central unit to continue operating in a degraded mode. However, there is no concept of assembling a system from component parts.

A Cyber CPU is used primarily for computation. It exercises no direct control over I/O, but relies on PP programs to do this. The CPU communicates with the memory (CMC) through one or two dedicated ports, depending on the model. The CPU architecture and software architecture are described further on page 6, "CPU Architecture" and on page 12, "Software Architecture".

A job executes in a region of central memory (CM) whose size may be up to 128K words. Because this address space is too small for many tasks, a mechanism called extended memory (EM) allows the job access to an additional, potentially much larger region of memory. The job can copy data between its CM region its EM region very rapidly because EM has a much faster access speed than disk. This mechanism is also used by the operating system to transfer data such as I/O buffers and job regions between CM and EM. Executable instructions and addressable data must, however, be kept in the CM region, since only two special data transfer instructions can reference data in EM.

The physical CM in a system can have a maximum size of 256K words, allowing multiple jobs to be resident in CM at the same time. Additional jobs are kept "resident" in EM, and swapped to and from CM to make them executable, again much faster than swapping to disk. The CM is used for operating system data and

CM regions of executable jobs. The EM is used for EM regions of jobs, I/O buffers, and CM regions of jobs which have been swapped to EM.

In the 800-series machines (including the 865 and 875 models if UEM is enabled), both types of memory are actually provided by the same hardware. For this reason, 800-series extended memory is called Unified Extended Memory (UEM). The operating system has some freedom over how it divides the UEM into CM and EM, although it is still limited to 256K for CM. In the 865 and 875 models without UEM, and all earlier (700-series, 70-series, 6000-series) models. there is external EM hardware which being physically different from CM, cannot be treated as part of CM.

The final component of a CYBER 170 mode compatible system is the Input/Output Unit (IOU) consisting of a set of Peripheral Processing units (PPs). All systems have at least 10 PPs and can be expanded to 15 or 20 in the low-end systems, or 14, 17, or 20 in the high-end systems. The PPs are described in more detail on page 7, "PP and I/O Architecture".

CPU Architecture

The CYBER 170 mode compatible CPU has a 60-bit word length; its instruction set is register oriented. The CPU has two states, or modes: program (or user) and monitor. There are no privileged instructions, but some instructions perform different functions in monitor mode than in program mode. There are eight 60-bit X registers used for calculation and moving data to/from memory, eight 18-bit A registers used for memory addresses and some calculations, and eight 18-bit B registers used primarily for floating point operations, indexing, and counting. Additionally, there is an 18-bit program counter (P), a register to support arithmetic exceptions, and a variety of privileged registers, such as base and bounds registers which are not visible to user programs (see page 9, "Hardware Protection Mechanisms").

During program execution, all CM addresses manipulated by the CPU are offset by the value of the base (Reference Address, RA) register, and may not exceed the bound (Field Length, FL) register. The process of "offset"ing an address is performed with each memory reference and is not visible to the executing program. The 18-bit A registers and the P register limit a program's total address space to 256K, and the operating system further restricts it to only 128K.

Data is moved to/from memory by loading an address into an A register, whereupon the data in the corresponding X register is moved to or from memory. Registers X6 and X7 are used for writing to memory, registers X1 through X5 are used for reading, and register X0 can be used only for calculation. All data manipulations are performed on registers. Data can be transferred between registers. In addition, some models support memory to memory character operations via special BDP (Business Data Processing) instructions.

The processor provides fixed-point and floating-point operations, boolean operations, shifts, transfers, character string (6-bit characters) and extended memory operations. Character operations (move, compare) use a descriptor word in memory to supply addresses, lengths, and offsets. Extended memory is accessed with an address from an X register, and can be read or written one word at a time or in blocks. The low-end machines (810, 815, 825, 830) do not implement the block move and string instructions in hardware. Rather, an attempt to use one of those instructions causes a transfer to the native mode of the machine (Cyber 180 Virtual State), where a small program called the Environment Interface (EI) performs the operation in software and then returns the machine to its Cyber 170 state to continue the program.

PP and I/O Architecture

The CYBER 170 mode compatible I/O architecture consists of peripheral devices under the control of the Peripheral Processors (PPs). The peripheral devices implement no protection mechanisms of their own (other than correct operation); all protection is provided by the programs in the PPs.

The PPs are fully independent processors, each running a distinct program to perform I/O and operating system functions. Each group of ten (or five) PPs actually shares the same hardware, multiplexing it to present the appearance of ten (or five) processors running at one-tenth (or one-fifth) the speed of the actual hardware. All PPs have access to all central and extended memory through the memory port dedicated to the IOU.

Any PP can control any of the I/O channels in the system; there may be 12 or 24 channels. Several I/O channels are dedicated and are present in all IOUs: one for the maintenance interface, one for the display station, one for a real-time clock, and one for the two-port multiplexer. The maintenance interface is used to test and run diagnostics on the CPU and central memory, and also for system initialization. The display station is the standard operator console; in normal operation, it is controlled by a dedicated PP running the console driver. The real-time clock is present only in some models. It is a battery-backed time-of-day clock which provides the system with the correct time(1) at system initialization. If it is not present, operator input is required to set the correct time. The two-port multiplexer provides an asynchronous serial interface to one or two terminals, and can be used for remote maintenance. The other channels are used to connect arbitrary peripherals, or a pair of PPs can use an I/O channel to communicate with each other.

The PPs are 12-bit processors with a 4096-word address space, and each PP always has a full 4096 words. They perform I/O transfers to and from the channels in 12-bit units, assembling these into 60-bit words for transfer to and from central memory. The PPs have three registers: A, R, and P. The A register (18 bits) is used for computation and some central memory addressing (in the low 256K only). The R register (up to 26 bits) is used for central memory addressing, providing the capability to reference all of central and extended memory. The P register is the program counter.

The Network Processor Unit (NPU) is a particular type of peripheral. It is connected to an I/O channel and treated as an I O device, but it is actually a programmable processor loaded with communications control programs under the control of the central system Trusted Computing Base (TCB). The NPU is programmed primarily in Pascal. There are five different NPU models: the 2551-1 through 2551-4, and the 2552-2. They differ only in hardware capabilities. Since the NPU is controlled only through a PP, users are unable to address it. Since users can not address it and it does not implement part of the protection mechanism, the NPU will not be discussed further.

The PPs run only programs provided by the central system TCB; they cannot be programmed by users. When a PP is not performing a task, it is assigned by the monitor to a pool of available PPs. While awaiting assignment, each pool PP continuously executes a small resident program which monitors that PP's task assignment word in CMR. When a request is detected, the PP resident program locates and loads the requested program from the system library

(1) The time of day is maintained by the CPU during system operation, through operating system software that reads the free-running clock register present in all CPUs. This register is initialized from the real-time clock or by the operator.

and initiates execution. On task completion, PP execution control returns to its resident monitor program. The PPs can interrupt the CPU to inform it that an operation has completed, or to request some specific service from the CPU monitor. This is done by forcing the CPU to perform an Exchange Jump (see page 10, "Program and Monitor Modes").

Hardware Protection Mechanisms

There are two main hardware protection mechanisms used by NOS: the "base and bounds" protection implemented by the RA and FL registers, and the distinction between program and monitor mode enforced by the Exchange Jump instruction. These two mechanisms provide protection for CPU programs. However, there are no internal PP hardware protection mechanisms for PP programs; all internal protection mechanisms for PP programs must be implemented in software. PP programs are protected from interference from the CPU and other PP programs by the separate execution domain provided to each PP.

Since users can only execute CPU programs, these mechanisms suffice to protect user jobs from one another. Also, since the user's only interface to the operating system is through CPU programs, these mechanisms protect the operating system from user jobs.

While the separate execution domains of the PPs provide some separation within the operating system, for those parts of the operating system that run on the CPU the hardware protection mechanisms do not provide any effective means for establishing a hierarchy of privilege within the CPU-based operating system itself, or of isolating parts of the CPU-based operating system from other parts. Similarly, the hardware protection mechanisms do not provide any effective means for validating user-supplied parameters in operating system requests. Therefore, all such validation must be performed by software. The problem of parameter validation is made worse by the division of function between the CPU and the PPs and the lack of a consistent or centralized parameter validation method or module.

Base and Bounds

The base and bounds mechanism is a simple memory address mapping and protection mechanism that defines the address space of an executing CPU program. The RA and FL hardware registers control access to central memory; the Reference Address - Extended memory

(RAE) and Field Length - Extended memory (FLE) hardware registers control access to extended memory. These registers are not visible to any executing CPU program.

Memory is allocated to a job in one contiguous block. The absolute address of the beginning of this block is stored in the RA register, and the length of this block is stored in the FL register. All addresses generated by CPU programs are relative; that is, the hardware adds the contents of the RA register to the specified address to obtain the absolute address. Therefore, the job refers to its address space as relative address 0 through relative address FL-1. The RAE and FLE registers are used to constrain a user's access to extended memory in the same manner as the RA and FL registers constrain a user's access to central memory.

The RA, FL, RAE, and FLE registers vary in size depending on the CPU model and the total amount of configurable memory: in the 180 835, for instance, the RA and FL registers are both 21 bits (giving a maximum of 2 million words of central memory); the RAE register is 21 bits, and the FLE register is 24 bits (giving a maximum of 16 million words of extended memory).

Program and Monitor Modes

The CYBER 170 mode compatible CPU has two operating modes: program and monitor. The CPU switches between these two modes using a mechanism called an Exchange Jump. An Exchange Jump always changes the execution mode, from program to monitor, or monitor to program. When invoked, Exchange Jump saves the entire set of CPU registers (both the program-visible registers, such as the A, B, and X registers, and the privileged registers, such as the RA, FL, RAE, and FLE registers) in a 16-word memory region (in system space) called the exchange package and reloads all the CPU registers from the previous contents of the region. In effect, an Exchange Jump swaps the current CPU state with a saved CPU state stored in memory.

Exchange Jumps occur under the following three conditions:

A) The CPU executes an eXchange Jump (XJ) instruction

If the CPU is in program mode, this causes an exchange jump using the exchange package at the location specified by the CPU Monitor Address (MA) register and the address specified in the XJ instruction is ignored. If the CPU is in monitor mode, this causes an exchange jump with the exchange package defined by the address in the instruction. This aspect of the XJ instruction is the only program-visible difference between program and monitor modes. The effect is that a program issuing an XJ always enters the operating system at a known point, whereas the operating system has a choice of which exchange package, and therefore, which job, to start when switching back to program mode.

B) The CPU detects an error (overflow, invalid address, parity)

If the CPU is in program mode, this causes an exchange jump using the exchange package at the location specified by the CPU MA register. The error flag field in the saved exchange package is used to indicate the type of error and to allow this entry into the operating system to be distinguished from one caused by the execution of an XJ instruction. If the error occurs in monitor mode, this is a fatal error; depending on the model, this either causes a trap to native mode, or halts the CPU.

C) A PP executes an EXN, MXN, or MAN instruction

The EXN instruction forces an Exchange Jump, regardless of the current state of the CPU. The MXN and MAN instructions cause an Exchange Jump only if the CPU is in program mode; these instructions have no effect if the CPU is in monitor mode. When the Exchange Jump occurs, The CPU loads the exchange package at the address specified by the MA register (for the MAN instruction) or the PP's A register and R register (for the EXN and MXN instructions). Thus, the PP, by setting the program counter in the exchange package, determines what module of CPUMTR is executed next by the CPU.

Although this is a somewhat unorthodox mechanism, as compared with the traditional "supervisor call" or "monitor call" implemented in many other systems, it is functionally equivalent and uses only a small additional amount of software to provide protection equivalent to other two-state machines.

Software Architecture

Introduction

NOS can be roughly divided into four parts: CPUMTR, MTR, privileged CPU programs, and PP programs. CPUMTR is a CPU program which is the user's only interface to the system. CPUMTR cooperates with MTR, a PP program, in controlling the operation of NOS. There are privileged CPU programs that perform operating system functions, and there are PP programs that perform additional tasks. These parts will be discussed in more detail in later sections.

The low end of central memory is called Central Memory Resident (CMR) and is reserved for system use. In this area are the system tables and communication areas, CPUMTR, and the system program libraries. Central memory above CMR is available for loading jobs. Jobs cannot access CMR directly because of the restrictions enforced by the RA and FL registers.

A job which is loaded in central memory is said to be at a control point. The number of control points in NOS is fixed at deadstart and defines the number of jobs which can reside in CM concurrently. A job which is not loaded in central memory is said to be rolled out; it resides on disk or in extended memory. A job may be rolled out when it is not ready to execute, or when a job with higher priority pre-empts it. A job may also be rolled out while awaiting completion of some event such as time-sharing user input or the mounting of a tape. Rolling out a job completely frees up the control point resources for other uses while awaiting these potentially long events.

A job is defined by its entry in the Executing Job Table (EJT), its Control Point Area (CPA) and Negative Field Length (NFL), and its assigned memory. The EJT entry holds global job information, and the CPA and NFL contain the job's process state. NOS uses the job's ordinal number in the EJT to identify the job; a user uses the Job Sequence Name (JSN), a four-character identifier.

NOS communicates with user jobs through central memory. A job requests an operating system function by placing the request in word 1 of its assigned memory (absolute location RA+1). NOS always signals receipt of the request by clearing RA+1. Completion and/or status of the request may be posted in buffers and parameter blocks in the user's assigned memory.

CPUMTR

Introduction

CPUMTR plays important roles in both operating system control and user-visible functions. CPUMTR handles function requests from MTR, PP programs, the system control point, and user programs, and initiates program requests to PPs and the system control point.

CPUMTR has two execution modes: monitor mode which is not interrupted by PP exchange requests, and program mode which is interruptable. Monitor mode is used for quick functions such as interlocks which must be completed immediately and program mode is used for longer running tasks which can be interrupted and run at low scheduling priorities. In some cases, such operations run at scheduling priorities below user job priority so as to absorb any unused CPU time.

User-Visible Functions

CPUMTR is the user's only interface to NOS. All user function requests are made by placing the request in location RA+1 and performing an XJ instruction. CPUMTR examines the contents of location RA+1 to determine the request. The name of the requested function is checked for validity, and if it is acceptable, the request is passed on. The function may be performed by another module of CPUMTR, or by a PP program. If the function is to be handled entirely by CPUMTR, the request is performed immediately; otherwise the request is queued. The user may ask that I/O be done either asynchronously (i.e., without waiting for it to complete) or synchronously; all other requests are done synchronously. Jobs waiting for a request to complete are said to be in recall. To return to the user program, CPUMTR executes an XJ instruction.

CPUMTR processes requests from the system control point and user sub-control points in a similar way. The system control point runs in user mode, but has an RA of 0 and an FL of CM size, which distinguishes it from user control points. The system control point places its requests in a special location within CPUMTR, rather than in location RA+1. User sub-control points are differentiated from normal user control points by the setting of the "sub-control point active" flag in the user's CPA. If this flag is set, CPUMTR clears it, and invokes the user's control point executive to process the sub-control point's request thereby allowing the user's control point executive to screen and/or respond to sub-control point requests.

Hardware errors while in user mode also force an exchange jump to the MA register's location. CPUMTR distinguishes these from user requests by the fact that the contents of location RA+1 and the P register (from the job's CPA) are zero.

Operating System Control

CPUMTR executes with the RA register set to 0, and the FL register set to the size of central memory. Likewise, the RAE register is set to 0, and the FLE register is set to the size of extended memory. This enables CPUMTR to access the data structures in CMR and the data structures of user jobs.

CPUMTR handles requests from MTR and from other PP programs. A PP program makes a request by placing it in a CMR buffer dedicated to that PP. The PP program then executes an MXN instruction, which causes the CPU to conditionally perform an exchange jump. The exchange jump occurs only if the CPU is in program mode; if the CPU is in monitor mode, the instruction has no effect. Each PP has a CMR buffer allocated to it for an exchange package. By convention, the exchange package is set up by the PP program so that, after the exchange jump, execution will begin at a designated location in CPUMTR. CPUMTR indicates completion of a request by placing the return value in the PP's CMR buffer.

CPUMTR is used by PP programs to provide exclusive access to shared data in CMR. This is a software convention, since any PP can read and write any part of central memory. The disk allocation tables are handled this way, as are the job allocation tables. CPUMTR also does buffer management for the PPs which handle high-speed disks.

CPUMTR assigns PPs to jobs to handle requests for operating system services. A similar task done by CPUMTR is the one of maintaining the recall queues. Jobs waiting for events (e.g., function completion, terminal I/O, time) are moved from the wait queue to the recall queues, and back again when the events occur.

Privileged CPU Programs

Privileged CPU programs constitute an important part of NOS. They perform functions which would otherwise have to be performed by PP programs or CPUMTR. As NOS evolved, the capabilities of privileged programs were increased to reflect the changing balance between the CPU and the PPs, however these capabilities do not reflect an overall design for program privilege.

Privilege is determined by the contents of a word in the job's CPA. This word is only set by the loader when it does an absolute load from the system library. Each program in the system library has an entry in the Program Status Table (PST), which lists the privileges associated with this program. These privileges are then copied from the PST to the CPA by the loader.

The defined privileges are:

LDR= used by the loader, to read execute-only files.

CLB= used by an interactive editor, which is now obsolete.

SDM= used by programs which do not want their control statement arguments echoed in the dayfiles. (Dayfiles are described in more detail on page 27, "Audit".) For example, the permanent file utilities have this privilege, so that file passwords are not written to the dayfile.

DMP= used to create a rollout file upon loading. This is used by programs that "dump" user memory, and by programs that are loaded into user space when called by a user program. This allows a CPU program to load itself into a user's memory space, execute and manipulate the image of the user's job in the rollout file. After completion of the request, the user's job may be eligible for rollin to a control point which is cleared in the process.

ARG= used by programs which do their own control statement processing, rather than accept the restrictive default processing.

VAL= used by programs which must be able to execute regardless of whether the user has been validated. For example, the programs which actually do the validation (such as CHARGE) require this privilege.

SSM= used by programs which process sensitive information. This privilege prevents dumping, and ensures that memory will be cleared before the next program without this privilege executes.

SSJ= used by programs which require special systems status. Many different capabilities are accorded these programs (for example, the capability to violate the system's security policy). Such a program is essentially omnipotent, since it also has the capability to read or write absolute memory locations.

MTR

MTR is a PP program which shares responsibility for system control with CPUMTR. It is loaded at system boot, or deadstart, time into a dedicated PP and remains active until another deadstart.

The principal task of MTR is the handling of function requests from other PP programs. Most function requests handled by MTR involve data channels. MTR is used by PP programs to solve exclusive access requirements. For instance, the assignment of JSNs to jobs and the assignment of data channels to PP programs both are done by MTR.

Other MTR functions include monitoring the recall queues for jobs that have changed status and checking location RA+1 for a user request. MTR is also responsible for maintaining the system real-time (free-running) and time-of-day clocks. MTR also monitors the state of CPUMTR to detect abnormalities and can detect and report (via DIS) the failure of CPUMTR.

PP Programs

PP programs perform many critical functions within NOS. These functions include physical I/O processing, job scheduling and advancement, swapping, and operator console management. PPs have direct access to all of central memory and to almost all physical devices.

PP programs are written in assembly language, and they are usually highly optimized to reduce their size and/or increase their speed. Since the PP has a limited address space, programs often incorporate techniques such as self-modifying code and partial overlays of other PP programs loaded only from the system library on disk and/or in CMR.

There is a trend towards moving functionality from PP programs to CPUMTR and CPU programs. For example, CPUMTR now performs parameter checking for I/O requests, a function which had been performed by the PP program which handles I/O. However, PP programs are, and will remain, an important element of NOS.

Subsystems

NOS subsystems are system programs that are assigned to a control point in central memory and are thus treated as jobs. They exist to provide centralized control of various tasks. This is especially efficient when a subsystem is used simultaneously by a number of users. Subsystems often perform tasks of a time-critical nature such as communicating with users' terminals, controlling peripheral equipment, and performing real-time process control functions.

Subsystems have several characteristics that distinguish them from user control points. They are started (automatically at system start time or manually by an operator) as system origin jobs(1) and are loaded from the system library. Thus, they have the privileges that go with system origin jobs and central library programs, in particular the SSJ= privilege. In addition, subsystems may be assigned a specific control point; they are not swappable from central memory; their scheduling priority is independent of CPU and CM time slices; and they usually have a high CPU priority.

The System Control Point Facility provides for user control point communication with some executing subsystems through RA+1 calls. Users must be validated to use the system control point facility.

NOS supports up to 20 subsystems. Currently there are 17 available from CDC, a site can run two of its own making, and one is reserved for future use.

The following is a list of the subsystems included in this evaluation:

Network Access Method (NAM): provides a generalized method of using a communication network for switching, queuing, buffering and transmitting data. NAM interfaces the network application

---

(1) "system origin jobs" are those jobs which are initiated via the system console by an authorized user and which acquire certain privileges associated with that job origin type.

programs (e.g., IAF and RBF) with PP program PIP (Peripheral Interface Processor) which in turn services the connected Network Processing Units (NPUs) which are the terminal front-end processors.

Interactive Facility (IAF): acts as interface between a user job and the network communicating through NAM. IAF is used by NOS to process all interactive terminals. It associates a terminal with a job in the Executing Job Table (EJT).

Remote Batch Facility (RBF): interfaces with the network to receive batch jobs, send output, and interact with the operator at a remote terminal with unit record equipment (card readers, card punches and printers). RBF enters jobs in the input queue and monitors the output queue for available files to be copied to the remote terminal.

Batch I O (BIO): used by NOS to manage and drive the local unit record equipment. The control point contains buffers and tables, while PP programs access it, create input queue files, and dispose of print or punch queue files.

Magnetic tape subsystem (MAG): controls all tape operations. MAG maintains information for RESEX (the resource executive), namely the unit descriptor tables on tape drives and their status.

Software Protection Mechanisms

There are some software protection mechanisms which are used by all TCB modules. Other mechanisms are used only for PP programs, since they lack internal hardware protection mechanisms.

Each TCB module that can be invoked by a job must implement the mechanisms which are relevant to its operation rather than using a centralized mechanism. The module is responsible for copying its parameters from the job's address space to CMR, and for validating those parameters. For example, it must ensure that any address given by the job is within the job's assigned memory. The module must also determine whether the job has any privileges, such as the SSJ= privilege, which are pertinent to the function performed by the module. It is difficult to verify that these tasks are performed correctly by each TCB module, since there is no standard mechanism to perform them.

The parameters for an operating system request are placed in location RA+1. One of the parameters may be the address of a block in the user's address space which contains additional parameters. All parameters must be copied, except for RA+1 itself. The contents of RA+1 need not be copied, since the TCB enforces the convention that the relative address of any parameter block or buffer must be greater than 1; the TCB can then assume that the contents of RA+1 have not been modified by an asynchronous operating system request.

Since all the protection mechanisms for PP programs are software-based, all PP programs are part of the TCB, even though the program's intended function may not be security-relevant. In order to ensure that all PP programs invoke the software protection mechanisms, PP programs are only loaded from the system program library. In addition, only certain PP programs may be explicitly invoked by unprivileged jobs. These programs are designed so that they should work correctly under all conditions. These programs may in turn invoke less robust PP programs.

While these protection mechanisms, in conjunction with the hardware protection mechanisms, are sufficient to protect the operating system from user jobs, and to protect user jobs from each other, they cannot protect the operating system from itself. That is, the TCB cannot enforce the principle of least privilege for its modules. All CPU data structures are global, in the sense that they can be accessed by all TCB modules. The most prominent example of the violation of the principle of least privilege is the large set of capabilities which are accorded programs with the SSJ= privilege.

This page intentionally left blank.

EVALUATION AS A C2 SYSTEM

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. The discretionary access control mechanism shall, 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.

Applicable Features

The NOS Permanent File system provides discretionary access control between all users and all permanent files which store user programs and data. Of the three file types, permanent, local, and queued, only permanent files may be shared. Local files are temporary working files specific to a user and are not sharable. Queued files can only be found in the input and output queues. They are accessible only to their creator and NOS system routines.

There are two types of permanent files: direct access files and indirect access files. When accessing an indirect access file, the system creates a temporary copy of the file for use during the job. Changes are made to the copy only; the permanent file remains unchanged until the copy is written back. No temporary copy is made for direct access files; changes are made directly to the file itself.

The NOS file system is a flat file system. Users are grouped into families. A family is a set of logical devices that store permanent files. Logical devices include disk drives which contain user files. Users may be associated with more than one family. The system administrator may provide a default family for the system which will be used by all users that do not specify a family during login. It is also possible for different users with the same username to belong to separate families. This must be avoided as it may result in access compromise in auxiliary devices as well as generating confusion. An auxiliary device is a self-contained file device (e.g., disk drive) which may be defined as public or private (user-owned).

Each user owns a single catalog in each family for which he is authorized and that catalog contains permanent files. Filenames need only to be unique in a catalog. Thus, full identification of a NOS permanent file within a family is: username/filename.

Users in different NOS families may share access to a file by explicit placement of the file on an auxiliary (disk) device. Being outside of a NOS family, files on that named disk may be shared across all families using implicit and explicit access permissions.

The access control mechanism is based on ownership of files. Only the file owner (the user owning the catalog containing the file) may grant or rescind access permissions and the owner may not pass this capability to other users. The mechanism is implemented using implicit file types and explicit access controls (i.e., access control lists) and possibly also a password. If a file is protected with a password then all users, with the exception of the file owner, must supply the password when attempting access. The owner may also place an expiration date on the password to further restrict access.

NOS supports three implicit file types: private, semiprivate, and public. Files designated as private may be accessed only by their owner (creator) and by those granted explicit access permission by the file owner. Files designated as semiprivate may be accessed by their owner as well as any user who knows the file_name, the owner user_name, and the file password (if any) and who has not been explicitly denied permission (via the file's Access Control List (ACL)) to the file. The system records the user name of each user who accessed the file, the number of accesses made, the date and time of last modification, and the date and time of the last access by each user. This information is made available to the file owner. Files designated as public may be accessed by their owner and by any user who knows the file name, the owner user name, and the file password (if any). In contrast to private and semiprivate files, the system records only the number of times the file was accessed and the date and time of the last access for public files.

The access control lists (known as the PERMIT mechanism in NOS) allow the following access types: execute, read, append, modify, write and null. The owner specifies the type and extent of sharing with other users. If the owner takes no action to expand a file's access after creation, a file may be accessed only by its owner. The description of the ATTACH command in the Permanent File Commands Chapter of NOS Version 2 Reference Set, Volume 3 -- System Commands, provides a good definition of shared access modes.

The files of users grouped into a NOS family may be shared within that family by implicit and/or explicit access permissions specified by the file owner. Implicit access to a file for all users in that family may be specified by the owner declaring the file as public or semiprivate and specifying a mode of access such as read (the default), write, etc. Explicit file access permissions on files specify access for a named user to a specific file via defined modes of access (default is read). These explicit file permits provide access controls to the granularity of a single user and may be used to exclude access via the null access mode. The owner may also place an expiration date on the ACL term to further restrict access.

In summary, the system will grant file access to an unprivileged user after going through the following algorithm. In all cases, if the file is password protected, the user must specify the correct password. If the file is not public and the user is found on the file's access control list, only the specified access type(s), including null, is(are) granted. If the file is public or the user is not found on the access control list, then implicit file type is used to determine file access (if any). Both a file password and access via the access control list must be unexpired in order for a user to gain access to a file. The file is always accessible to its owner. Users in different families cannot share files except via the auxiliary devices as described above.

Two classes of NOS users are authorized to access users' files regardless of the discretionary access controls on those files. One group consists of security administrators who can examine any file in the system using the system console. The other group consists of master users whose usernames contain asterisks (all normal users' usernames contain alphanumeric characters only). These users have read-only access permission to files of users whose usernames' characters match the non-asterisk characters of the special asterisk users.

Conclusion

NOS Security Evaluation Package satisfies the C2 Discretionary Access Control requirement.

Object Reuse

Requirement

> When a storage object is initially assigned, allocated, or reallocated to a subject from the TCB's pool of unused storage objects, the TCB shall assure that the object contains no data for which the subject is not authorized.

Applicable Features

Central and extended memory are automatically cleared when allocated to any user job. An installation option can be selected to clear central memory whenever it is to be released from a job, to clear all unassigned central and extended memory during all levels of deadstart, and to clear central memory vacated by certain types of storage moves. However, this option is not necessary to satisfy this requirement.

On mass storage, the TCB maintains accurate allocation and file positioning information and correctly performs all mapping of logical to physical I/O as well as the actual I/O operations. File space is allocated by tracks. End-of-information and end-of-file pointers are kept in the system sector area. For a direct access file, as the user writes into the file, the end-of-information pointer is set to the last physical record unit written. Therefore, the user can only read what the user has written. For indirect access files, when the file is written back onto mass storage by a system program, the end-of-information pointer is set to the last physical record unit being written into the storage track.

If the new indirect access file is the same size as the existing file, the new file is written over the old indirect access file. However, if the new file is smaller than the existing file, the new file is written over the old one and the remaining space becomes a hole. Finally, if the new file is larger than the old file, the old file becomes a hole. A new hole is found, or the new file is placed at the end of the indirect access files. The software ensures that a user cannot read past the end-of-information pointer or access a hole in any way.

Additionally, there is a user-invokable function for mass storage that will write over mass storage with binary zeros or zeros and ones before optionally releasing the files to the system.

Conclusion

NOS Security Evaluation Package satisfies the C2 Object Reuse 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 use a protected mechanism (e.g., passwords) to authenticate the user's identity. 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.

Applicable Features

NOS requires all users to identify themselves to it before they are allowed to access system resources. The user identification must be authenticated by a password. Within a family, each username uniquely identifies a user. The username, SYSTEMX, identifies the operating system.

Users have different passwords for batch and interactive use. Upon login, users may enter their familyname, username and password on separate lines. The password will then be protected by communications echoplex suppression and a screen clear function, when appropriate, after entry of the password. However, if a user chooses to enter all of the identification information on a single line, echoplex suppression will not protect the password, although the screen will still be cleared after the entire line has been entered. If a batch job is entered via the SUBMIT(1) command, the user may protect the password by using the /USER directive in place of the USER command. The directive is reformatted by the system to insert the current user's username and password.

Users must have special authorization to change their password(s). A user must change his or her batch password from a batch job. Likewise, interactive passwords must be changed from interactive jobs. In addition to the authorization required to change passwords, a separate authorization is required to specify an expiration date or expiration term for a password.

Identification and authentication information for each user is stored in a validation file, which is only accessible by special system jobs. There is a separate validation file for each family. The system uses the username as a key to search this file. The password is stored here in an encrypted form. The password entered is encrypted and then compared against the stored password. A one-way encryption algorithm(2) is used to encrypt the passwords.

In NOS all user activity is attributable to a job sequence name. In the system dayfile and the account dayfile (for a description of dayfiles see page 27, "Audit"), the username is recorded along with the job sequence name in the recording of the login transaction. Thereafter, the job sequence name is recorded with all logged transactions. Users are held accountable for all actions by matching the job sequence name by the action in question to the login transaction for that job sequence name.

Conclusion

NOS Security Evaluation Package satisfies the C2 Identification and Authentication requirement.

---

(1) The SUBMIT command is used to enter a batch job from an authenticated and executing job.

(2) Knoble, H. D., Forney, C., and Bader, F. S., An Efficient One Way Encrypting Algorithm, ACM Transactions on Mathematical Software, March 1979, pp. 97 - 107.

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, and actions taken by computer operators and system administrators and/or system security officers. 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. The ADP system administrator shall be able to selectively audit the actions of any one or more users based on individual identity.

Applicable Features

NOS maintains two logs, the system dayfile and the account dayfile, that are used for auditing security-relevant events. The system dayfile keeps a history of all control statements for all jobs processed. The account dayfile keeps a record of all the resources charged to a job. These dayfiles are automatically created at system startup if they do not already exist. The time of the event, the type of event and the job sequence name are recorded in the logs.

Operations on permanent files are audited along with the name of the file affected. The introduction of a permanent file into a user's address space is audited (e.g., GET, ATTACH) as is the deletion of permanent files. Data transfer operations are not audited (e.g., READ, WRITE). The failure to access permanent files that belong to other users is audited.

Valid and invalid login attempts are recorded along with the identification of the terminal of origin. When a file is placed into the input queue or the output queue, the event is recorded with the date and time. The queueing of a file upon system recovery is also recorded.

Two utilities, AFD and DFD, are used to dump the account dayfile and the system dayfile. Normally, when a dayfile gets large, it is terminated and copied to magnetic tape. When a dayfile is terminated, a new mass storage file is created to receive all subsequent messages. A job using these utilities must be a system origin job, or the user must have system origin privileges and the system must be in debug mode. The operator and the security administrator are authorized to examine the dayfiles. Actions taken from the system console are also recorded in system dayfiles and are auditable.

An audit reduction tool is provided with the system. The tool merges system and account dayfiles providing a complete record of events. The tool can be used to select records based on event type, user identification, terminal login port, and file or tape access. Audit Reduction Tool documentation describes the tool's functionality and objectives in more detail.

Conclusion

NOS Security Evaluation Package satisfies the C2 Audit requirement.

System Architecture

Requirement

> The TCB shall maintain a domain for its own execution that protects it from external interference or tampering (e.g., by modification of its code or data structures). Resources controlled by the TCB may be a defined subset of the subjects and objects in the ADP system. The TCB shall isolate the resources to be protected so that they are subject to the access control and auditing requirements.

Applicable Features

The CYBER 170 mode compatible machines provide two states, Monitor mode and program mode. Monitor mode is used only by the TCB and allows the XJ instruction to specify the address of the exchange package, which is not permitted in program mode. It is not possible for a user program to change from program to monitor mode without involving the TCB, since all changes to monitor mode (Exchange Jumps) transfer immediately to a defined location in the TCB, and all exchange packages (which contain the job state as saved and restored by Exchange Jump) are under the TCB's control.

At all times, program addresses are relocated by a "base" (RA) register, and restricted by a "bound" (FL) register, so that a user job cannot reference data outside the region defined by those values. When a program invokes the TCB, switching to monitor mode, the RA and FL are changed to describe the TCB's memory. The TCB programs, its internal data, and its per-user control information, are all stored in memory outside the region defined by those registers for a user job, so the TCB and its data cannot be referenced or modified except by the TCB itself. The TCB programs in the PPs can never be affected directly by user programs.

The TCB validates all accesses to protected objects, and validates all data addresses in user requests, before acting on the requests.

Conclusion

NOS Security Evaluation Package satisfies the C2 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.

Applicable Features

CDC provides a set of diagnostic programs that can be used to verify the correct operation of the central system, peripheral processors, and peripherals. The entire central unit (made up of CPU, memory, and PPs) can be tested using the maintenance channel interface of a PP, which allows direct access to many internal hardware registers. The lowest-level functional tests reside in ROM accessible to the PPs; higher-level tests are read from tape or disk. Most diagnostics can be run remotely through the two-port multiplexer attached to a PP, if such access is allowed. Since most diagnostics require dedication of the entire central system, they cannot be run concurrently with system operation.

Conclusion

NOS Security Evaluation Package satisfies the C2 System Integrity requirement.

Security Testing

Requirement

> The security mechanisms of the ADP system shall be tested and found to work as claimed in the system documentation. Testing shall be done to assure that there are no obvious ways for an unauthorized user to bypass or otherwise defeat the security protection mechanisms of the TCB. Testing shall also include a search for obvious flaws that would allow violation of resource isolation, or that would permit unauthorized access to the audit or authentication data.

Applicable Features

CDC supplied the evaluation team with an extensive set of functional tests that verified the correct operation of the security mechanisms. In addition to running CDC's functional tests, the evaluation team developed specific tests to insure that authentication and auditing data cannot be compromised. The test plan covered the functional areas of access controls, permanent file operations and logins. The evaluation team is satisfied that these tests show that there are no obvious flaws that would allow unauthorized users to bypass or defeat the security mechanisms of the TCB.

CDC runs these tests on each release of NOS to ensure that all functional areas of the system perform correctly. The testing program is an integral part of the design process at CDC. Test plans are designed during the earliest phases of development at CDC.

Conclusion

NOS Security Evaluation Package satisfies the C2 Security Testing 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.

Applicable Features

Section 14 of Volume 2 of the NOS Reference Set(1) provides users guidance on the use of the system protection mechanisms. This section discusses user responsibilities, password protection and maintenance, login procedures, and access control rechanisms. For the most part, the the user is provided an overview and general guidance on the use of the system protection mechanisms. References are provided to other sections of system documentation for further information.

Conclusion

NOS Security Evaluation Package satisfies the C2 Security Features User's Guide requirement.

---

(1) NOS Version 2 Reference Set, Volume 2 - Guide to System Usage, Publication Number 60459670, Revision D.

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.

Applicable Features

The Security Administrator's Handbook(1) describes functions associated with site administration of a NOS system in a secure mode. This handbook consists of four major sections:

Section 1 Gives an overview of the system's protection philosophy and security mechanisms.

Section 2 Describes the responsibilities of the system administrator. Includes warnings and recommendations for controlling system privileges.

Section 3 Provides guidelines for system maintenance and operation. Among other topics, gives guidance on code maintenance, tape management, user authorizations, auditing procedures, audit reduction tools, and password management.

Section 4 Explains installation options for generating a secure system.

The Security Administrator's Handbook also contains several appendices discussing user validation and restricted commands, macros, and utilities. An additional appendix (Appendix F) is provided as part of the NOS Security Evaluation Package. This appendix gives information for the proper use of the NOS audit reduction tool.

---

(1) NOS Version 2 Security Administrator's Handbook, Publication Number 60460410, Revision B.

Further information on audit logs is contained in two additional manuals. Section 2 of the Administration Handbook(1) explains how an operator changes the active audit log and dumps audit log records to a file. Section 5 of the Analysis Handbook(2) describes the structure of audit log records and the types of events recorded by them.

Conclusion

NOS Security Evaluation Package satisfies the C2 Trusted Facility Manual requirement.

Test Documentation

Requirement

> The system developer shall provide to the evaluators a document that describes the test plan and results of the security mechanisms' functional testing.

Applicable Features

The NOS Multi-level Security General Internal Design Document in Appendix D describes the test plan for the security features functional testing. This test plan includes tests of access control. permanent file operations, login, as well as the multi-level security features.

CDC has a well written design plan to test the security features of NOS. Personnel, separate from the developers of NOS, prepared this comprehensive test plan. The testing program is an integral part of the design process at CDC. Test plans are designed during the earliest phases of development. Each test is thoroughly documented and well maintained for future references. Automated testing tools are also available.

---

(1) NOS Version 2 Administration Handbook, Publication Number 600459300, Revision A.

(2) NOS Version 2 Analysis Handbook, Publication Number 60459840, Revision D.

Conclusion

NOS Security Evaluation Package satisfies the C2 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. If the TCB is composed of distinct modules, the interfaces between these modules shall be described.

Applicable Features

An overview of the system's protection philosophy and security mechanisms can be found in section 1 of the Security Administrator's Handbook.(1) NOS Multi-level Security General Internal Design Document explains the security policy and rules to be followed for accessing the secured objects. The NOS discretionary security policy is defined in this document. This is the most comprehensive source of security-relevant design documentation. The interfaces between the TCB sections are defined in the System Programmer's Instant manual.

Conclusion

NOS Security Evaluation Package satisfies the C2 Design Documentation requirement.

---

(1) NOS Version 2 Security Administrator's Handbook. Publication Number 60460410, Revision B.

ADDITIONAL SECURITY FEATURES

Introduction

This section presents an evaluation of NOS Security Evaluation Package security features that exceed the C2 requirements. Some of these additional features do satisfy higher level requirements of the Criteria in terms of the capability provided, however, they do not meet any of the assurance requirements above the C2 level.

NOS Security Evaluation Package does implement some controls required in the handling of classified or non-classified but sensitive information. These controls are not sufficient to fully meet any of the mandatory access control or labeling requirements of the Criteria. Because these controls may be applicable in some environments, the results of their evaluation are included in this section.

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., access control lists) shall allow users to specify and control sharing of those objects. The discretionary access control mechanism shall, either by explicit user action or by default, provide that objects are protected from unauthorized access. These access controls shall be capable of specifying, for each named object, a list of named individuals and a list of groups of named individuals with their respective modes of access to that object. Furthermore, for each such named object, it shall be possible to specify a list of named individuals and a list of groups of named individuals for which no access to the object is to be given. Access permission to an object by users not already possessing access permission shall only be assigned by authorized users.

Applicable Features

The PERMIT mechanism (see page 21, "Discretionary Access Control") allows the user to specify for each object an arbitrary list of users by name and a mode of access, possibly including null access, for each of those users.

Conclusion

NOS Security Evaluation Package does not satisfy the B3 Discretionary Access Control requirement because the PERMIT mechanism does not provide any way to specify a group of users and their access.

Labels

Requirement

> Sensitivity labels associated with each subject and storage object under its control (e.g., process, file, segment, device) 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 security level of the data, and all such actions shall be auditable by the TCB.

Applicable Features

The objects controlled by NOS Security Evaluation Package are devices (tape drives, card readers, and card punches) and files (permanent, local, and queued).

All devices have an entry in the Equipment Status Table (EST) which contains a minimum and a maximum access level defining the access range for that device. No devices have categories associated with them.

All permanent files have an entry in the file owner's catalog. This entry contains the category set and access level of the file. Local files are known only to a single job and are destroyed at job termination. Each job has a local File Name Table (FNT) with entries that contain the local file's access level. Local files do not have categories associated with them.

The Queued File Table (QFT) has an entry for each queued file in the system. This entry includes the file's access level. Queued files do not have categories associated with them.

Each user on the system has an entry in a validation file (there is one validation file for each family) that contains the access levels and categories valid for the user.(1)

In addition, NOS Security Evaluation Package has a Service Class Table (SCT) that contains minimum and maximum access levels valid for a job depending on the job's origin type (system, batch, remote batch, or interactive). The system origin type entry in the SCT provides the access levels valid for the system which, in conjunction with the system category set found in CMR, describe the sensitivity label limits for the entire system.

The subjects controlled by NOS Security Evaluation Package are jobs. Jobs are assigned a sensitivity label (access level and category set) based on the following criteria:

> When a job is created by a user through login, the job is assigned the lowest access level valid for the user that is also valid for the job's origin type. If the job is of interactive origin type, then the assigned access level of the job must also be less than or equal to the access level limit for the terminal communication line. If the job was created by the SUBMIT or ROUTE command, the access level of the job is that of the local file containing the job if the value is valid for both the user and origin type.
>
> If the job has a JOB card, the maximum valid access level of the job is taken from the AL parameter if the value is valid for both the user and the origin type.
>
> Further, a job may have associated with it some subset of the thirty-two categories available on the system (by default, named CAT0 through CAT31). User job categories are

---

(1) Note that the access levels for the user are not a range with a minimum and maximum limit but rather a one-to-one mapping of eight bits to the eight possible access levels (by default, named LVL0 through LVL7). For example, if a user is validated for LVL1 and LVL4, he is valid for the access levels LVL1 and LVL4 and not for the range of access levels LVL1 through LVL4. To have access to the range of access levels LVL1 through LVL4, the user must be validated explicitly for LVL1, LVL2, LVL3, and LVL4.

set to all categories that are valid for both the user and the system. For example: if a user is valid for categories CAT1, CAT2, and CAT4 and the system is valid for categories CAT1, CAT3, CAT4, and CAT5, then the job category set will be set to CAT1 and CAT4 since CAT2 is invalid for the system and CAT3 and CAT5 are invalid for the user. The sensitivity label for a job, once determined, is stored in the job's Control Point Area.

A job's access level may change to a higher access level during the life of the job if that new level is valid for the user and the origin type. This change may only be an increase in value and once changed to a higher access level, a job's access level may be lowered only if the user has the necessary privilege (CLJI) granted in the user validation file. For more information on changing of job access level, see page 45, "Mandatory Access Control".

The sensitivity labels associated with each object and subject on NOS Security Evaluation Package are used to validate all job accesses to storage objects.

Unlabeled data may only be imported to the system through tapes without operator maintained sensitivity labels. Since all users privileged to use tapes may import unlabeled data by using tapes without operator maintained sensitivity labels, procedures must be in place to regulate the use of tapes without operator maintained sensitivity labels. All use of tapes is audited by the TCB.

Conclusion

NOS Security Evaluation Package does not satisfy the B1 Labels requirement. NOS Security Evaluation Package meets some of this requirement. However, categories are not completely implemented and do not meet the B1 requirement. In addition to permanent files, all other storage objects need categories associated with them and need to be used for determining job access to those storage objects.

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.

Applicable Features

Device labels (which contain minimum and maximum access level limits but no categories) are stored in the EST in CMR which can be manipulated only by the TCB. For single-level devices (card readers, card punches, and tape drives), the minimum and maximum access levels must be set equal to insure the proper pairing of physical labels with internal labels associated with the data when the data is imported or exported.

Local FNTs store the access level for local files. The QFT stores the access level for queued files. Both types of tables may only be manipulated by the TCB. No categories are associated with local and queued files.

The Control Point Area, which contains a job's sensitivity label including access level and categories, is stored in either CMR (for jobs resident in memory) or a system rollout file (for jobs that are currently swapped out) both are accessible only through the TCB.

Catalogs, which contain labels including categories and access levels for all permanent files, may only be changed through the TCB. Users may request to change a permanent file's access level with the SETPFAL command provided the new access level is valid for both the job and the device on which the file resides. A user must be privileged to lower a permanent file's access level.

Users may also add (upgrade) or remove (downgrade) categories from permanent files with the SETPFAC command. However, no special privilege is required for a user to remove categories from permanent files. This operation is in violation of the mandatory access control policy (see page 45, "Mandatory Access Control").

The TCB directly controls all exportation of information. The user must request the TCB to perform any I/O operation. Trusted sensitivity labels (access levels and categories) are exported by the TCB for all permanent files. Only trusted labels containing the information's access level are exported for printed output.

Conclusion

NOS Security Evaluation Package does not satisfy the B1 Label Integrity requirement. NOS Security Evaluation Package meets some of this requirement. However, categories need to be associated with devices and queued files so that the sensitivity labels (access levels and categories) of these objects are accurately represented. When these labels are exported from the TCB (for multilevel devices), they must contain both the access level and category set for the data. In addition, the ability for all users to subtract categories from permanent files must either be tightly controlled by the TCB or disallowed.

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 auditable by the TCB. The TCB shall maintain and be able to audit any change in the current security level associated with a single-level communication channel or I/O device.

Applicable Features

Communication lines are assigned a maximum access level limit in the Network Configuration File (NCF). The NCF is used by the system to load the Network Processing Units (NPUs). This access level limit may be changed only by the operator by editing the NCF and reloading the NPUs. The minimum access level limit for communication lines is always the minimum level allowed for the entire system and, in conjunction with the maximum access level limit, it defines the valid limits for that line. However, communication lines only operate at a single level and are therefore single-level devices. Communication lines do not have categories associated with them.

I/O devices (disk drives, tape drives, card readers, card punches, and printers) are all assigned a minimum and maximum access level limit in the EST as described on page 36, "Labels".

Removeable-pack, auxiliary disk drives are multilevel devices on NOS since the information exported to these disks has trusted labels stored on the disk. Printers are also multilevel devices since trusted labels are printed on all output by the subsystem BIO (see page 43, "Labeling Human-Readable Output").

Card readers, card punches and tape drives only operate as single-level devices. The minimum and maximum access levels for these devices must always be the same to insure the proper pairing between sensitivity labels associated with the data internally and physical sensitivity labels attached to card decks and tapes. The access level limits for disk drives and tape drives can only be changed at deadstart time. The system administrator may change the access level limits for card readers, card punches, and printers during system operation with the SECUREQ command. All use of the SECUREQ command is audited in the system dayfile. No I/O devices have category access limits associated with them.

Conclusion

NOS Security Evaluation Package does not satisfy the B1 Exportation of Labeled Information requirement because it does not include categories in the labels kept for communication lines and I/O devices.

Exportation to Multilevel 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.

Applicable Features

The multilevel devices on NOS are removeable-pack, auxiliary disk drives and printers. Permanent files are the only objects exported to auxiliary disk drives. Each auxiliary disk file has a header called the system sector. The system sector contains information about the file that the system requires and can not be accessed by a user. A catalog entry contains the access level and category set for the file and resides in the system sector for each permanent file. Since the system sector resides on disk with the file, the sensitivity label associated with that file does reside on the same physical medium as the file.

All printed output is controlled by the trusted subsystem BIO. The security level of the output is printed on the system banner page. However, the categories of the output are not printed. For a detailed description of labeled output, see page 43, "Labeling Human-Readable Output".

There are no multilevel communication channels on NOS.

Conclusion

NOS Security Evaluation Package does not satisfy the B1 Exportation to Multilevel Devices requirement because it does not include categories in the labels printed on output.

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 can reliably communicate to designate the single security level of information imported or exported via single-level communication channels or I/O devices.

Applicable Features

Card readers, card punches, and tape drives are single-level devices on NOS. The minimum and maximum access level limits for these devices must be set to be the same. This will allow the operator to physically mark any tape or card output. These devices have no categories.

All communication lines on NOS are single-level. When a user job logs in across a communication line, the line will assume the current sensitivity label of the job if the access level portion of the label is valid for the line. Communication lines do not have categories.

Conclusion

NOS Security Evaluation Package does not satisfy the B1 Exportation to Single-Level Devices requirement because it does not include categories in the labels on devices and communication lines.

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, hardcopy 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, hardcopy output (e.g., line printer output) with human-readable sensitivity labels that properly represent the overall sensitivity of the output or that

---

(1) The hierarchical classification component in human-readable sensitivity labels shall be equal to the greatest hierarchical classification of any of the information in the output that the labels refer to; the non-hierarchical category component shall include all of the non-hierarchical categories of the information in the output the labels refer to, but no other non-hierarchical categories.

> 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 sensitivity of the output. Any override of these marking defaults shall be auditable by the TCB.

Applicable Features

NOS Security Evaluation Package has eight access levels and thirty-two categories. By default, the printable names for the access levels and categories are LVL0 through LVL7 and CAT0 through CAT31 respectively. The system administrator may specify different printable name for the access levels and categories by reassembling pertinent TCB modules and generating a new version of the TCB.

The trusted subsystem BIO provides a header banner page which precedes the first page of all user output. On this header page, the file's access level is printed. However, the file's associated categories are not printed. Therefore, the sensitivity label printed on the header page does not properly represent the sensitivity of the output.

NOS Security Evaluation Package does not provide a corresponding trusted trailer banner page or other means to identify the end of a printed file. By using the SECHDR command, a user may individually select to have a trailer banner page printed; however, the TCB does not force this requirement. Therefore, the end of printed output is not properly marked to show the information's sensitivity. Further, since the end of an output file is not distinctly identifiable, there is no way to prevent a user process from generating a false header banner page within the text of the output file. This false banner page could convince the operator that a portion of the output file is at a different security level.

The top and bottom of each page of human-readable output is not labeled by default. The command SECHDR allows the file's access level to be printed on the top and bottom of every page as a user option. The SECHDR command does not print the file's category set.

Conclusion

NOS Security Evaluation Package does not satisfy the B1 Labeling Human-Readable Output requirement. The header banner page must include the file's category set to properly reflect the output's sensitivity. The TCB must also, by default, mark the top and bottom of each page, and the end of all hardcopy output with the information's complete sensitivity label. Any override of these default markings must be audited. Further, the TCB must provide a means by which the beginning and end of each output is identifiable to prevent untrusted processes from generating false banner pages.

Mandatory Access Control

Requirement

> The TCB shall enforce a mandatory access control policy over all subjects and storage objects under its control (e.g., process, file, segment, device). 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. The following requirements shall hold for all accesses between subjects and objects controlled by the TCB: 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.

Applicable Features

NOS Security Evaluation Package provides a mandatory access control policy over subjects (jobs) and objects (files and devices). The basis for this policy is sensitivity labels, a combination of a hierarchical access level and non-hierarchical categories, which are assigned to subjects and objects on the system.

Hierarchical access levels are assigned to all subjects and objects. However, non-hierarchical categories are only assigned to jobs and permanent files. Categories are not associated with local files, queue files, or devices. Thus, complete labels need to be implemented on the remaining system objects in order for the mandatory access control policy to be fully enforced. The mandatory access control policy is applied to read and write operations, change of security level, and assignment of permanent or tape files.

The hierarchical access level for attached direct access permanent files and attached tape files may not change. The access level of local files, indirect access permanent files, and unattached direct access permanent files may be changed automatically by the TCB or via the SETPFAL or SETFAL commands.

A local file is created by obtaining (via the GET or OLD command) an indirect access permanent file or by using a new local filename in a write operation. Local files exist only while the associated job exists unless otherwise destroyed by the user. A user may raise the access level of local files by using the SETFAL command. The new access level must be valid for both the job and the device on which the file resides. The SETFAL command may also be used to lower the access of a local file; however, this operation requires a special privilege.

The TCB may also automatically raise the access level of a local file. When a job attempts to write a local file at a lower access level, the system will raise the file's access level to the job's current access level if the new access level is valid for the device on which the file resides. Otherwise the write request will be denied. The TCB will not automatically lower a local file's access level.

A user may change the access level of indirect access permanent files and unattached direct access permanent file by using the SETPFAL command. The new access level must be valid for both the job and the device on which the file resides. In order to lower a permanent file's access level, the user must have special privilege.

The categories of indirect access permanent files and unattached direct access permanent files may be changed by using the SETPFAC command. The new category set must be a subset of the job's current set. This command can add to or subtract from the file's current category set. However, no special privilege is required to subtract categories from a permanent file. This is in violation of the mandatory security policy. Categories are not associated with the other system objects (local, queue, and tape files and devices).

The hierarchical access level of a job may be changed automatically by the TCB or by using the SETJAL command. The SETJAL command can be used to raise or lower a job's current access level. The new access level must be valid for the job. A user must have special privilege to lower a job's access level. If a job attempts to read a file at a higher access level, the job will automatically advance to the file's access level if the new access level is a valid access level for the job. Otherwise, the read operation will be denied. The TCB will not automatically lower a job's current access level. The categories associated with a job can not change and are always set to all of the categories the user is validated for.

To assign an indirect access permanent file to a local file or to attach a direct access permanent file to a job, the file's access level must be a valid (but not necessarily the current) access level for the job and the file's categories must be a subset of the job's categories. When assigning tape files to a job, the TCB will assign the tape file the access level of the job unless the user explicitly requests a different access level on the tape assignment request. If this access level is not within the tape drive's maximum and minimum authorized range, the assignment will be rejected. However, the TCB does not ensure that the access level a user requests for a tape is consistent with the physically controlled sensitivity label of the tape reel. Categories are not associated with tape files.

See page 36, "Labels" for more information on subject and object labeling.

Conclusion

NOS Security Evaluation Package does not satisfy the B1 Mandatory Access Control requirement. Categories need to be implemented on all objects on the system. The TCB must consider these categories when making all access control decisions. The ability to subtract categories from objects (downgrade) must be controlled in some manner. Also, a method must exist to ensure that the TCB maintained labels for tape files is consistent with tape reels physically controlled sensitivity labels.

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 determine the security level and authorizations of subjects that may be created to act on behalf of the individual 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.

Applicable Features

In addition to satisfying the C2 requirement, NOS also addresses the additional B1 requirement for identification and authentication. The validation file for each family contains information to determine the set of authorized security access levels and categories for each user in that family. This information is used to determine the access levels of the user's job. See page 25, "Identification and Authentication" for a further explanation of C2 features.

Conclusion

NOS Security Evaluation Package satisfies(1) the B1 Identification and Authentication requirement.

Trusted Path

Requirement

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

Applicable Features

NOS provides a mechanism to support a trusted communication between the TCB and the user for initial user login and authentication. This mechanism involves the Network Validation Facility (NVF) and the Communications Control Program (CCP). NVF is a component of the NAM subsystem which validates users logging in. CCP is a program which is loaded into the Network Processing Units (NPUs) by the Network Supervisor, another component of NAM.

The trusted path for login is initiated by entering a sequence of characters. This sequence of characters is dependent on the type of terminal connected to the NPU and the definition of a security character during CCP installation. For example, a secure login for most asynchronous terminals requires pressing the BREAK key (or ATTN key) and entering the security character.

When the initial login trusted path is established, the CCP will terminate any current connection and will reconnect the user to the host computer in order for NVF to initiate a new login.

The security character is, by default, not defined and therefore, the secure login feature is not activated.

---

(1) Although NOS Security Evaluation Package satisfies this requirement at the B1 level, it does not satisfy the assurance requirements above its rated level.

Conclusion

NOS Security Evaluation Package satisfies(1) the B2 Trusted Path requirement.

System Architecture

Requirement

> The TCB shall maintain a domain for its own execution that protects it from external interference or tampering (e.g., by modification of its code or data structures). Resources controlled by the TCB may be a defined subset of the subjects and objects in the ADP system. The TCB shall maintain process isolation through the provision of distinct address spaces under its control. The TCB shall isolate the resources to be protected so that they are subject to the access control and auditing requirements.

Applicable Features

The additional requirement for B1 is that the TCB isolate user jobs in distinct address spaces. Since the TCB provides each user job with a separate and distinct region of memory, and restricts the job with the base and bound registers, it satisfies this requirement.

Conclusion

NOS Security Evaluation Package satisfies(2) the B1 System Architecture requirement.

---

(1) Although NOS Security Evaluation Package satisfies this requirement at the B2 level, it does not satisfy the assurance requirements above its rated level.

(2) Although NOS Security Evaluation Package satisfies this requirement at the B1 level, it does not satisfy the assurance requirements above its rated level.

EVALUATORS' COMMENTS

Mandatory Access Controls

The mandatory controls provided by NOS Security Evaluation Package, while not sufficient to satisfy the Criteria's mandatory access control or labeling requirements, do provide some of the mechanisms needed when handling classified or non-classified but sensitive information. The controls that are present do separate hierarchically ordered sensitivity levels. Thus, NOS Security Evaluation Package would be an adequate environment for the development of applications software which needs to run in an environment requiring mandatory access controls with no compartments.

Denial of Service

Although denial of service is not a stated Criteria requirement, its importance to security, reliability and availability makes it an important topic of consideration. Many denial of service attacks depend on resource exhaustion (i.e., the ability to use a disproportionate amount of some limited resource). NOS Security Evaluation Package, through the user validation file, provides limits on equipment usage (maximum number of tapes, disks, etc.), file usage (ability to create permanent files, maximum number and size of files, etc.), machine usage (maximum CPU time per job step, maximum central and extended memory usage, etc.), system usage (number of jobs in the system, etc.) and application usage (which network applications a user can access, etc.) on a per-user basis. NOS Security Evaluation Package provides sufficient quotas to support very good prevention of denial of service due to resource exhaustion.

Configuration Management

CDC maintains control of changes to NOS's design documentation, implementation documentation, test documentation, source code, and test code. Tools are provided for the generation of new TCB versions from the source code. In addition, tools are provided for comparing a newly generated version with the previous TCB version.

A coding standards document establishes standard procedures to be used by programmers in the development and maintenance of the operating system source code. Included in these standards are details of the program documentation to be embedded within the source code. This documentation may be extracted by the DOCMENT utility.

The source code for different products within the TCB is preserved by the MODIFY and UPDATE utilities. The operating system itself is preserved by MODIFY while some of the subsystems are maintained by the UPDATE utility.

These utilities provide a means for entering source changes into selected modules kept in a compressed form within a library. Whenever changes are entered, a name is associated with each set of modifications; the modification set name may be a new or an existing name. Once each modification set has an associated name, it may be applied selectively. A list of the modification sets for modules within a library may be listed with the CATALOG command.

Testing

Testing is an integral part of CDC's development process. CDC maintains a separate department, Integration and Evaluations (I&E), which is responsible for maintaining and running the necessary tests on every system before release. People from both the development and integration shops work together to initially develop tests for new features. The I&E Department is then responsible for maintaining the tests and insuring they are kept up to date with any changes to the system that might affect the tests. The I&E Department has developed a "test language" to facilitate easy development and maintenance of the test suite. They have also developed the necessary software to automatically evaluate each test run and flag any tests that fail.

REFERENCE MATERIALS

References

CDC Manual	CDC Document Number
NOS Version 2 Reference Set	
Volume 1: Introduction to Interactive Usage	60459660
Volume 2: Guide to System Usage	60459670
Volume 3: System Commands	60459680
Volume 4: Program Interface	60459690
NOS Version 2 System Maintenance Reference Manual	60459300
NOS Version 2 System Programmer's Instant	60459370
NOS Version 2 Application Programmer's Instant	60459360
NOS Version 2 Operator/Analyst Handbook	60459310
NOS Version 2 Security Administrator's Handbook	60460410-02
NOS Version 2 Installation Handbook	60459320
NOS Full Screen Editor	60466420
Xedit Version 3 Reference Manual	60455730
Cyber Loader Version 1 Reference Manual	60429800
Modify Reference Manual	60450100
Cyber Interactive Debug Version 1 Reference Manual	60481400
Cyber Initialization Package (CIP) User's Handbook	60457180
CDC Cyber 170 Computer Systems Models 815 and 825	60469350
CDC Cyber Computer Systems Models 810 and 830	60469420
CDC Cyber 180 Models 810, 830: Virtual State, Volume I	60469680
CDC Cyber 170 Computer Systems Models 835 and 855	60469290
CDC Cyber 170 Model 835, Cyber 180 Model 835: Virtual State, Volume I	60469690
CDC Cyber 170 Computer Systems Models 865 and 875	60458920
CDC Cyber 170 Models 825, 835 and 855 Computer Systems: General Description	60459960
CDC Cyber 170 Computer Systems Model 815, 825, 835 and 855 Executive State: Volume 2, Instruction Descriptions and Programming Information	60458890
CDC Cyber 180 Model 990: Virtual State, Volume I	60462090
CDC Cyber 170 Model 835, 845, 855, Cyber 180 Model 835, 845, 855: Hardware Operator's Guide	60458390
Network Access Method Version 1, Network Definition Language Reference Manual	60480000

NAM Version 2/CCP Version 3 Reference Manual	60499500
Network Processor Unit 2551-1, 2, 3, 4, 2552-2 Reference Manual	60472800
2550-1-1 Emulator (6671/6676) Reference Manual	60474000
MSL151 Maintenance Software Reference Manual	60469400
SYMPL Version 1 Reference Manual	60596400
COMPASS Version 3 Reference Manual	60492600

CDC Internal Document	CDC Document Number

Note: The documents in this section are considered to be proprietary to Control Data Corporation (CDC). CDC reserves the right to refuse requests for this documentation.

Design Requirements (DR) NOS Release 6 [Version 2]	ARH3625
Design Requirements (DR) NOS Version 2.2	ARH5161
General Internal Design (GID) NOS Multilevel Security (MLS) Project	ARH5127
Design Action Paper (DAP) Network Host Products (NHP) Enhancements for NOS MLS	ARH5167
Design Action Paper CCP Support of Trusted Path	S4363
General Standard Programming Project Management Standards	1.01.100
Document Control System (DCS) Procedures 03/01/83	(none)
Design Action Paper (DAP) Trusted Path Identification	ARH5357
General Internal Design (GID) NOS A170 RTA	ARH4692
General Internal Design (GID) Cyber 180 State Interface Software for the A170 System	ARH2949
Advanced 170 Software NOS Overview DAP	ARH263
Preliminary DAP NOS MLS Phase 2	ARH263

CDC Seminars	CDC Seminar Number
Cyber CP COMPASS Student Handout	DA3020
NOS Advanced Compass Student Handout	FH3030
NOS V2 System Analysis Student Handout	FH4210
NOS V2 System Analysis Study Dump	FH4210
Cyber 170 Series 700/800 NOS Differences Student Handout	FH3400
NOS V2 System Maintenance and Installation	FH3400

CDC Functional Tests (MLS Test Suite)

MLS Functional Tests:

FALPL
JALPL/05-83
MLSARC
MLSPL/05-83
PFUSCPL/83-05-12
SECQUPL/05-83
PFSPL
TMSPL

This page intentionally left blank.

EVALUATED HARDWARE

Scope of Hardware Evaluation

The hardware covered by this evaluation is the CDC Cyber product line, including supported hardware present in the field at existing customer sites.

Although much of this hardware was not physically inspected or tested, the evaluation team examined engineering specifications for all hardware components to assure themselves that the components were equivalent, or where not equivalent, made no security-relevant differences in the TCB. Overall, very little code in the TCB deals with different types of hardware, and the differences between different models of hardware components are insignificant.

The primary requirement for hardware evaluation is that the hardware function properly. This was verified by the system integrity tests (see page 29, "System Integrity") and was not given a detailed reevaluation by the team. The integrity assurances provided by the CDC-supplied diagnostic tests are satisfactory.

List of Evaluated Components

This section lists, in several categories, the CDC marketing identification numbers for all hardware covered by this evaluation. This list is equivalent to the set of hardware officially supported by the evaluated release.

Although peripherals were not evaluated as such, the system was evaluated only for running with the supported set of peripherals. Therefore, they are included in this list to allow a determination that a particular configuration contains only devices that are supported by NOS Security Evaluation Package.

In cases where the detailed description of options or devices is not interesting from a security standpoint, the marketing identifiers have simply been listed without descriptions.

To operate in correspondence with the C2 rating, the hardware configuration must contain only components listed in this section.

Central System Combinations

Product Number(s)	Product Name
810A, 830A, 840A, 850A, 860A, 870A, 990E, 995E	Computer
810-43, 810-44, 810-63, 810-64	Hardware Package
830-44, 830-64	Hardware Package

Additional CPUs and CPU Enhancements

This list covers all the CPU models (as purchased individually) and the field-installable performance upgrades for converting one CPU model to another or enhancing the performance of a particular CPU.

Product Number(s)	Product Name
6200, 6400, 6500, 6600, 6700	600 Series Central Computer (Minimum 65K Central Memory)
71-14, 71-16, 71-18, 71-24, 71-26, 71-28	CYBER 70 Model 71 Central Computer (minimum 65K Central Memory)
72-12, 72-13, 72-14, 72-16, 72-18, 72-24, 72-26, 72-28	CYBER 70 Model 72 Central Computer (Minimum 65K Central Memory)
73-12, 73-13, 73-14, 73-16, 73-18, 73-24, 73-26, 73-28	CYBER 70 Model 73 Central Computer (Minimum 65K Central Memory)
74-12, 74-13, 74-14, 74-16, 74-18, 74-24, 74-26, 74-28	CYBER 70 Model 74 Central Computer (minimum 65K Central Memory)

Product Number(s)	Product Name
171-4, 171-6, 171-8, 171-12, 171-16	CYBER 170 Model 171 Central Processor
172-2, 172-3, 172-4, 172-6, 172-8, 172-12, 172-16	CYBER 170 Model 172 Central Processor
173-4, 173-6, 173-8, 173-12, 173-16	CYBER 170 Model 173 Central Processor
174-4, 174-6, 174-8, 174-12, 174-16	CYBER 170 Model 174 Central Processor
175-4, 175-6, 175-8, 175-12, 175-16, 175-108, 175-112, 175-116, 175-208, 175-212, 175-216, 175-308, 175-312, 175-316, 175-608, 175-612, 175-616	CYBER 170 Model 175 Central Processsor
176-8, 176-12, 176-16, 176-21, 176-22, 176-24, 176-31, 176-32, 176-34, 176-41, 176-42, 176-44, 176-408, 176-412, 176-416, 176-421, 176-422, 176-424, 176-431, 176-432, 176-434, 176-441, 176-442, 176-444, 176-501	CYBER 170 Model 176 Central Processor
170-720, 170-730, 170-740, 170-750, 170-760, 170M-875,	CYBER 170-700 Series Central Processor

Evaluated Hardware

Product Number(s)	Product Name
170-815, 170-825, 170-835, 170-845, 170-855, 170-865, 170-875	CYBER 170-800 Series Central Processor
180-810, 180-815, 180-825, 180-830, 180-835, 180-840, 180-845, 180-850, 180-855, 180-860, 180-990, 180-995	CYBER 180-800 Series Computer

Additional Memory and Memory Enhancements

Product Number(s)	Product Name
7030-1, 7030-2, 7030-4, 7030-8, 7030-16, 7030-101, 7030-102, 7030-104, 7030-108, 7030-116	Extended Core Memory
7040-100, 7040-200	Extended Semi-conductor Memory

Disk and Mass Storage Controllers and Options

Product Number(s)	Product Name
7152-1, 7154-1, 7154-2, 7154-3, 7154-4, 7155-1, 7155-11, 7155-12, 7155-13, 7155-14, 7144-401, 7165-21, 7165-22, 7639-21, 7639-22, 7880-1	Mass Storage Controller
7882-1	Mass Storage Transport

Product Number(s)	Product Name
836-221, 836-441	Disk Storage Subsystem
836-110	Disk Storage Option
7255-1	Disk Adapter
834-11, 834-12, 844-2, 844-21, 844-41, 844-44, 885-11, 885-12, 885-42, 895-1, 895-2	Disk Storage Unit
7881-1	Cartridge Storage Unit
7990-11, 7990-21	Storage Controller
7991-11, 7991-12	Storage Module

Tape Controllers and Options

Product Number(s)	Product Name
7021-21, 7021-22, 7021-31, 7021-32, 7021-41, 7021-42, 7152-1	Magnetic Tape Controller
639-1, 667-2, 667-3, 667-4, 669-2, 669-3, 669-4, 677-2, 677-3, 677-4, 679-2, 679-3, 679-4, 679-5, 679-6, 679-7	Magnetic Tape Transport
7221-1	Magnetic Tape Adapter

Communication Processors and Options

Product Number(s)	Product Name
2550-1, 2550-2, 2551-1, 2551-2, 2551-3, 2551-4	Network Processor (10449-1 option)
2554-32	MOS Memory Expansion (255x)
2556-10	Expansion Cabinet
2556-11	Loop Multiplexer Expansion
2558-3	Host Computer Coupler
2560-11, 2560-12, 2560-13, 2560-21, 2560-31, 2561-11, 2561-12, 2561-13, 2563-2, 2563-11, 2563-12, 2563-13	Communications Line Adapter
2580-5	Tape Cassette Transport
2580-6	Upgrade Kit (2551-3 to 2551-4)
2580-7	System Autostart Unit (255x)
2580-8, 2580-9, 2580-10	Communications Expansion Option

Peripheral Equipment (not evaluated)

This section lists all the peripheral devices supported by NOS Security Evaluation Package, and their options. These devices have not been evaluated specifically, since they contain no security-relevant components.

Product Number(s)	Product Name
18001-1, 18001-2, 6681-E, 6681-F, 6681-G, 6681-2, 6681-21	Data Channel Converter
18002-1, 18002-2	Operator Console
3446, 3446-2	Card Punch Controller
415-30	Card Punch
3447, 3447-2	Card Reader Controller
405	Card Reader
536-1, 553-1	Line Printer
580-12, 580-16, 580-20, 580-120, 580-160, 580-200	Train Printer Subsystem
5870-1	Non-Impact Printing System
755-20, 755-21	Impact Printer
533-1, 556-1	Remote Print Station
722	Display Terminal

This page intentionally left blank.

EVALUATED SOFTWARE

Scope of Software Evaluation

This section lists the programs that make up the various major divisions of NOS Security Evaluation Package software. Each subsection describes the division and lists the programs contained therein.

TCB Software

This represents the code that is considered part of the TCB.(1)

---

(1) Inclusion of a product name in this list does not mean that all of the product is necessarily part of the TCB, however since some part of the product contributes to the TCB the product is included in this list.

Product Number(s)		Product Name
p521-nn	p7x0-nn(1) p8xx-nn p990-nn	
	-67	NOS Security Evaluation Evaluation Package (This includes Dxxx-01 NOS 2 Package Version 2.4.1 Level 630, Dxxx-110 Tape Management System (TMS) for NOS 2.4.1, and NOS Audit Reduction Tool.)
-06	-10	Network Access Method (NAM) 1
-08	-11	Interactive Facility (IAF) 1
-26	-12	Remote Batch Facility (RBF) 1
-01 -76 -86	-01	NOS 2 Package Version 2.4.1 Level 630
	D8xxP-110 D7x0P-110	Tape Management System (TMS) for NOS 2.4.1

Non-TCB Software

This represents the software that is not part of the TCB and was therefore not evaluated.

---

(1) p521-nn Generic 6000, CYBER 70, CYBER 170 Series

p7x0-nn CYBER 170-700 Series and 176-xxx
x = 2, 3, 4, 5, 6, 7 (176)

p8xx-nn CYBER 170-800, CYBER 180-800, and 8xxA Series
xx = 10, 15, 25, 30, 35, 40, 45, 50, 55, 60, 65, 75

p990-nn CYBER 180-990, 990E, and 995E Series

p = D for Distributed Systems Software Policy (DSSP)
H for Single Site Software Policy

nn = Specific Software Product which is listed under the appropriate column

Product Number(s)		Product Name
-59	-20	FORTRAN 5
-79		
-39	-21	FORTRAN Extended 4
-77		
-12	22	FORTRAN Extended 4/Interactive Option
-78		
-46	-23	COBOL 5
-17	-24	Interactive BASIC 3
-51	-25	APL 2
-09	-26	PL/I 1
-14	-27	Sort/Merge 4
-24	-28	Xedit 3
-41	-29	CYBER Interactive Debug 1
-11	-30	ALGOL-60 5
-74	-32	FTN 4/5 Conversion Aid 1
-33	-33	Sort/Merge 5
-34	-34	PASCAL 170
-37	-37	Full Screen Editor
n/a	-39	Printer Support Utility
-30	-40	CYBER Database Control System 2
-31	-41	Data Description Language
-42	-42	Query/Update 3
-58	-43	FORTRAN Data Base Facility 1
-84	-48	Information Management Facility (IMF) 1

Evaluated Software

Product Number(s)		Product Name
-53	-53	Data Catalogue 2 Version 2.0
-85	-85	High Speed I/O Package
-150	-150	NOS On-Line Manuals
-905	-905	Conversion Aids Subsystem 3

Software Excluded from the TCB

This subsection lists that software that was excluded from the TCB and therefore it was not evaluated. Since this software does require modifying the TCB for proper installation, this software must not be used in a C2 configuration.

Product Number(s)		Product Name
-02	-02	Maintenance Package
-10	-04	Mass Storage Subsystem (MSS) 1
-87	-05	Tracer 1
-25	-15	CYBER Cross System 1
n/a	-94	Mass Storage Extended (MSE)

ACRONYMS

BIO	Batch I/O subsystem
CCP	Communications Control Program
CM	Central Memory
CMC	Central Memory Controller
CMR	Central Memory Resident
CPA	Control Point Area
CPU	Central Processing Unit
EI	Error Interface
EJT	Executing Job Table
EM	Extended Memory
EST	Equipment Status Table
FL	Field Length
FLE	Field Length - Extended memory
FNT	File Name Table
IAF	Interactive Access Facility
IOU	Input/Output Unit
JSN	Job Sequence Name
MA	Monitor Address (location of exchange package)
MAG	MAGNET
NAM	Network Access Method
NCF	Network Configuration File
NFL	Negative Field Length
NPU	Network Processing Unit
NVF	Network Validation Facility
P	Program (counter)
NOS/VE	NOS/Virtual Environment
PIP	Peripheral Interface Processor
PP	Peripheral Processor
PST	Program Status Table
QFT	Queued File Table
RA	Reference Address
RAE	Reference Address - Extended memory
RBF	Remote Batch Facility
RESEX	Resource Executive
SCT	Service Class Control Table
TCB	Trusted Computing Base
UEM	Unified Extended Memory
XJ	Exchange Jump

This page intentionally left blank.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS NONE
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT DISTRIBUTION UNLIMITED
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) CSC-EPL-86/003	5. MONITORING ORGANIZATION REPORT NUMBER(S) S228,358
6a. NAME OF PERFORMING ORGANIZATION Nat'l Computer Security Ctr — 6b. OFFICE SYMBOL (If applicable) C12	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State, and ZIP Code) 9800 Savage Road Ft. 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 NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
(U) Final Evaluation Report, Control Data Corporation NOS Security Evaluation Package

12. PERSONAL AUTHOR(S)
G. Wagner, J. Rub (Aerospace Corp.), W. Olin Sibert (Oxford Systems, Inc.), et al.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM ______ TO ______	860528	79

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	CDC NOS C2 EPL NCSC Trusted Computer System Evaluation Criteria

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

The security protection provided by Control Data Corporation's (CDC) Network Operating System (NOS) Security Evaluation Package running on CDC CYBER 170 mode compatible machines has been evaluated by the National Computer Security Center (NCSC). The security features of NOS were evaluated against the requirements specified by the Department of Defense Trusted Computer System Evaluation Criteria (the Criteria) dated 15 August 1983. This report presents the findings of the evaluation.

The NCSC evaluation team has determined that the highest class at which NOS satisfies all the specified requirements of the Criteria is class C2, and therefore NOS has been assigned a class C2 rating. Additionally, NOS provides some mandatory access controls that, while not sufficient to satisfy the Criteria's B1 mandatory access control and labeling requirements, do provide some of the mechanisms needed when handling classified or non-classified but sensitive information.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ■ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL LTC Lloyd D. Gary, USA	22b. TELEPHONE (Include Area Code) (301)859-4458	22c. OFFICE SYMBOL C/C12

DD Form 1473, JUN 86 Previous editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED