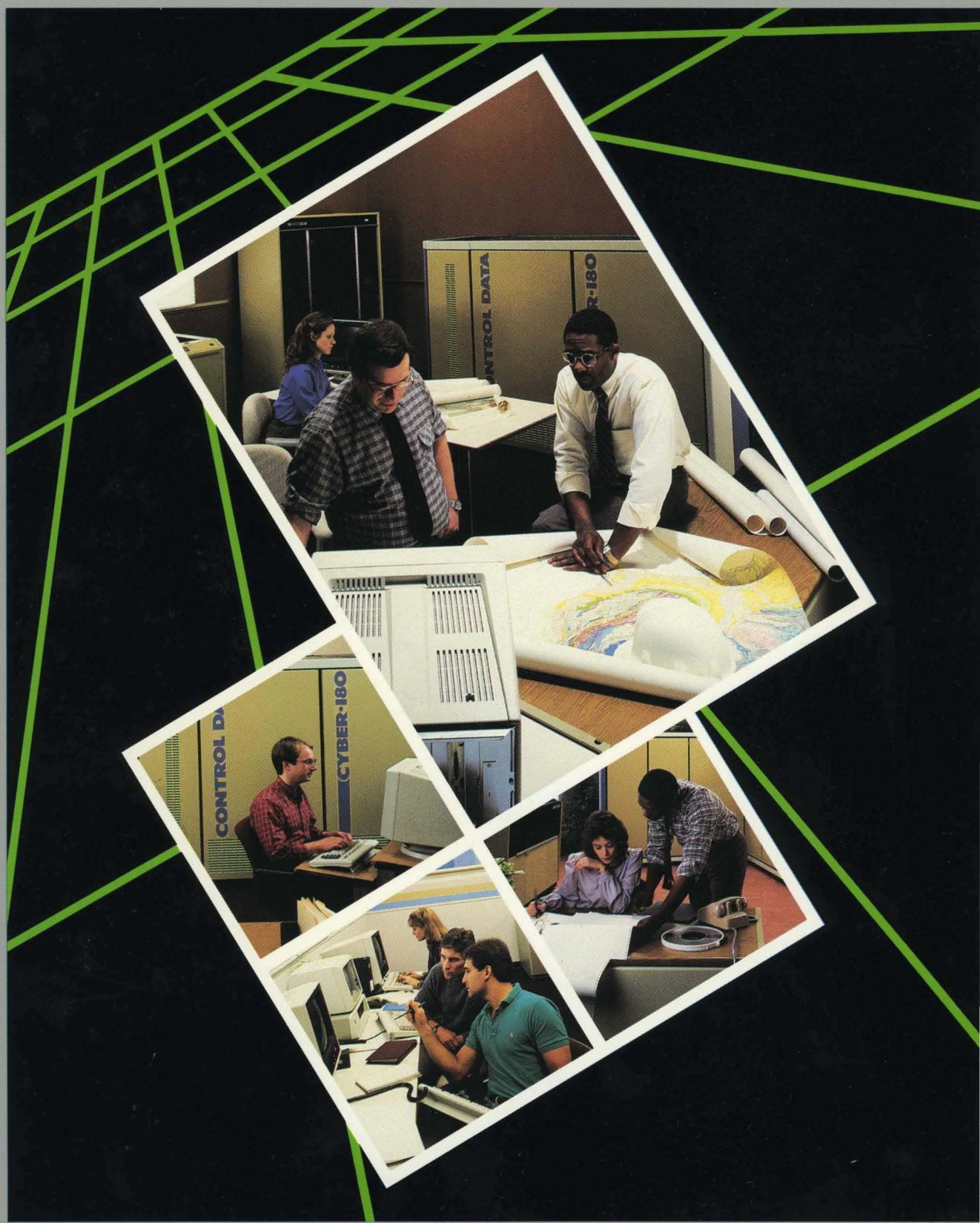
NOS/VE
Security Administration

CONTROL DATA



Usage

60463945

NOS/VE Security Administration

Usage

This product is intended for use only as described in this document. Control Data cannot be responsible for the proper functioning of undescribed features and parameters.

Publication Number 60463945

Manual History

Revision	System Version	PSR Level	Date
A	1.5.1	739	December 1989

Revision A of this manual documents NOS/VE security features for NOS/VE Version 1.5.1 at PSR level 739.

Contents

Tables

About This Manual

This manual describes the major security features of the CONTROL DATA® Network Operating System/Virtual Environment (NOS/VE).

Audience

This manual is primarily intended for site administrative personnel responsible for maintaining computer system security. However, the manual does contain security-related reference information that may be useful to operators or application programmers.

This manual assumes that the reader has a basic knowledge of the NOS/VE operating system and of the System Command Language (SCL) as described in the NOS/VE System Usage manual.

Conventions

The following conventions are used in this manual:

Boldface	In a command or CYBIL interface format description, required parameters are in boldface type.
Italics	In a command or CYBIL interface format description, optional parameters are in italic type.

Submitting Comments

There is a comment sheet at the back of this manual. You can use it to give us your opinion of the manual's usability, to suggest specific improvements, and to report errors. Mail your comments to:

Control Data
Technical Publications ARH219
4201 North Lexington Avenue
St. Paul, Minnesota 55126-6198

Please indicate whether you would like a response.

If you have access to SOLVER, the Control Data online facility for reporting problems, you can use it to submit comments about the manual. When entering your comments, use NV0 (zero) as the product identifier. Include the name and publication number of the manual.

If you have questions about the packaging and/or distribution of a printed manual, write to:

Control Data
Literature and Distribution Services
308 North Dale Street
St. Paul, Minnesota 55103-2495

or call (612) 292-2101. If you are a Control Data employee, call (612) 292-2100.

CYBER Software Support Hotline

Control Data's CYBER Software Support maintains a hotline to assist you if you have trouble using our products. If you need help not provided in the documentation, or find the product does not perform as described, call us at one of the following numbers. A support analyst will work with you.

From the USA and Canada: (800) 345-9903

From other countries: (612) 851-4131

Overview 1

Overview 1

This manual provides an overview of the basic security features of the NOS/VE operating system. Some parts of this manual provide overview information on security features that are documented in other NOS/VE manuals. Other parts of this manual, notably the Audit Utility chapter and the Data Encryption Standard appendix, provide primary reference information that is not described in other manuals.

Chapter 1: Overview

Chapter 1 summarizes the contents of this manual.

Chapter 2: Secure Mode Operations

Chapter 2 describes three security options related to system operations. These options can be selected during deadstart using the SET_SECURITY_OPTION system core command. The options are as follows:

- Disable the use of the System Operator Utility (SOU) from user terminals (thus, the SOU is available only from the system console).
- Disable the use of the system core debugger and other commands and utilities capable of directly reading system memory.
- Enable the use of the Audit Utility, which is described in the Audit Utility chapter of this manual.

Chapter 3: User Validations

Generally, the user validation features of NOS/VE are explained in detail in the NOS/VE User Validation manual. Chapter 3 of this manual provides an overview of the standard user validation controls for NOS/VE.

Chapter 4: Audit Utility

This chapter provides overview and reference information for the use of the ADMINISTER_SECURITY_AUDIT utility, also called the Audit Utility. The Audit Utility allows a security administrator to monitor many types of user activities, such as file attaches, submission of jobs for execution, and changes to user validations.

Appendix A: Related Manuals

This appendix lists the NOS/VE manuals referenced in this manual, along with information about how to order these manuals.

Appendix B: Audit Statistic Formats

The audit information controlled by the Audit Utility is collected by means of statistic interfaces similar to those described in the Statistics Facility chapter of the NOS/VE System Performance and Maintenance manual, Volume 1. Appendix C describes the statistic formats used to report audit information.

Appendix C: Data Encryption Standard

This appendix describes the SCL commands and CYBIL interfaces used to encrypt and decrypt data files. This appendix is intended for use by application programmers who require file encryption capabilities.

Secure Mode Operations 2

Secure Mode Operations 2

This chapter describes three optional security features available for monitoring or restricting the use of various system operation activities. These options can be selected during deadstart using the SET_SECURITY_OPTION system core command, which is described later in this chapter.

Security Options

The security options controlled by the SET_SECURITY_OPTION system core command are as follows:

- CONSOLE_OPERATION_ONLY
- SECURE_ANALYSIS
- SECURITY_AUDIT

CONSOLE_OPERATION_ONLY Option

The CONSOLE_OPERATION_ONLY option restricts use of the System Operator Utility (SOU) to the system console and to jobs initiated by the system console using the JOB and JOBEND commands. System operator activities cannot be performed from user terminals used as remote consoles, and it is not possible to log in to any $SYSTEM user on any family from a terminal other than the system console.

SECURE_ANALYSIS Option

The SECURE_ANALYSIS option deactivates the following monitor commands:

DUMPJOB
JDEBUG
SYSDEBUG
TDEBUG

This option also prohibits the system from automatically bringing up the system core debugger. The system displays an error message in the critical display window of the system console whenever one of the above commands is entered or when the system attempts to automatically start the system core debugger.

Any SECURE_ANALYSIS options activated for the system remain in effect across deadstarts, as well as during subsequent deadstarts. This includes the period of time from deadstart initiation to the point at which another SET_SECURITY_OPTION command can be entered to change the security options. However, reloading CIP causes the security option settings to be cleared for that window of time during deadstart.

SECURITY_AUDIT Option

The SECURITY_AUDIT option enables the Audit Utility, which is described in the Audit Utility chapter of this manual. Setting this option enables use of the ADMINISTER_SECURITY_AUDIT command, which starts the Audit Utility, but does not cause the system to begin performing audit functions. To initiate auditing of system activities, the site security administrator must enter the Audit Utility to select and activate the desired audit operations.

Commands and Functions

SET_SECURITY_OPTION System Core Command

Purpose Activates or deactivates a security option.

Format

SET_SECURITY_OPTION or
SETSO
option
value

Parameters

option

Specifies the name of the security option to be activated or deactivated. This parameter is required. Keyword values that can be specified are:

CONSOLE_OPERATION_ONLY

Restricts System Operator Utility activities to the system console.

SECURE_ANALYSIS

Deactivates the following monitor commands: DUMPJOB, JDEBUG, SYSDEBUG, and TDEBUG.

SECURITY_AUDIT

Enables the Audit Utility (ADMINISTER_SECURITY_AUDIT command).

ALL

Selects all security options.

value

Specifies whether or not the security option specified on the option parameter is to be activated or deactivated. This parameter is required. Keyword values that can be specified are:

ON

Activates the specified option(s).

OFF

Deactivates the specified option(s).

Remarks

- Each security option can be set only once during each deadstart of NOS/VE.
- For the SECURE_ANALYSIS option, from the time deadstart is initiated to the time this command is entered to change the SECURE_ANALYSIS option, NOS/VE continues to honor the value set at the previous deadstart.

- To ensure that the security options are set appropriately at every deadstart, you can place the SET_SECURITY_OPTION system core commands on the DCFILE file on the deadstart tape. Because the security options can be set only once per deadstart, placing these commands on the deadstart tape ensures that the options cannot be changed interactively during deadstart.

Examples The following command activates the Audit Utility:

```
set_security_option security_audit on
```

$SECURITY_OPTION SCL Function

Purpose Returns a boolean value indicating whether or not a specific security option is currently active.

Format **$SECURITY_OPTION(name, option)**
name
option

Parameters

name

Specifies the name of the security option. This parameter is required. The following keyword values can be specified:

CONSOLE_OPERATION_ONLY
SECURE_ANALYSIS
SECURITY_AUDIT

option

Specify the following keyword value:

ACTIVE

Returns an indication of whether or not the specified security option is active.

Examples

```
/display_value $security_option(security_audit,active)
TRUE
```

User Validations 3

User Validations 3

This chapter provides a brief overview of some of the more important security features related to NOS/VE user validations. Topics discussed include:

- User name validations
- User login passwords
- System Operator Utility (SOU)

User Name Validations

To log in to a NOS/VE system (that is, to be recognized as a valid NOS/VE user), an individual must have both a valid user name and password. The user name is assigned at the time a user validation is created. All user privileges and capabilities are assigned on the basis of user names. For example, user access to privileged or restricted job classes, to file transfer capabilities, and to remote operator capabilities are all assigned by association with specific user names.

Each user name is assigned to a NOS/VE family. The family is the basic unit of administration for the management of users and file space on NOS/VE. Each family has a family administrator who is responsible for maintaining user validations for the family, and each family has an allocated amount of disk file space available to users in the family.

As a site option, the responsibility for managing NOS/VE family members (users) can also be subdivided into account and project levels of administration. From a security standpoint, this is significant because it allows a finer level of control over user access to system resources. Access to system resources such as files, tape units, and job classes can be restricted on an account or project basis, as well as on an individual user name basis.

This is just a brief explanation of the security-related aspects of user name management on NOS/VE. For a more detailed explanation of user name management and of family, account, and project administration, see the NOS/VE User Validation manual.

User Login Passwords

Using login passwords prevents unauthorized individuals from gaining access to the system. To maintain optimum security against unauthorized access, sites must actively enforce policies and procedures designed to protect passwords.

Certain aspects of password protection must be handled by the enforcement of appropriate policies. For example, users should be discouraged from choosing passwords that are so simple or so obvious that they are easily guessed. Names, telephone numbers, and simple 1- or 2-character strings are not good choices for passwords. Also, users must be discouraged from sharing their passwords with coworkers or from leaving written notations of their passwords on desk tops or taped to the terminal.

To maintain optimum password security, users must also be encouraged to change their passwords at regular intervals. The NOS/VE ADMINISTER_VALIDATIONS utility provides a means of forcing users to change their passwords. For each user in the validation file, the site can specify a mandatory password expiration date or expiration interval. If the user fails to change the password prior to the expiration date, the system invalidates the old password and the user is no longer able to log in.

Users can change their passwords using the CHANGE_LOGIN_PASSWORD command, described in the NOS/VE Commands and Functions manual. The ADMINISTER_VALIDATIONS utility is described in the NOS/VE User Validation manual.

System Operator Utility

Most of the commands used to perform basic system operations tasks must be entered from within the System Operator Utility (SOU). Commands that can be executed within the SOU are further subdivided into groups of related commands. For example, the commands that allow an operator to call system displays constititute one group of commands, while another group consists of the commands used to control removable media devices such as magnetic tape units.

Associated with each group of operator commands within the System Operator Utility is a validation capability (also called an SOU capability). SOU capabilities are assigned using the ADMINISTER_VALIDATIONS utility, described in the NOS/VE User Validation manual. A user who is assigned one or more SOU capabilities can log in to an ordinary user terminal and perform any of the operator functions associated with those capabilities.

Table 3-1 provides a brief description of the SOU capabilities. For more detailed information on the use of the capabilities, see the NOS/VE Operations manual. By default, a user operating from the SYSTEM job on the system console is permitted all of the SOU capabilities listed in table 3-1.

Appropriate use of the System Operator Utility allows sites to limit individual operator capabilities to only those functions required for the job. By restricting access to the system console and assigning SOU capabilities to individual operators on an "as required" basis, the site can exercise much greater control over the use of system operator privileges.

Table 3-1. System Operator Utility Capabilities

Capability	Description
ACCOUNTING_ADMINISTRATION	Provides access to commands that manage the recording of statistics in the accounting log.
CONFIGURATION_ ADMINISTRATION	Provides access to commands that configure certain system options and characteristics, such as: system, network, and default job attributes; statistics recorded to the system statistic log and engineering log; tape validation; and system time and date format.
FAMILY_ADMINISTRATION	Provides access to commands that perform user validation and permanent file maintenance tasks for a family.
REMOVABLE_MEDIA_ OPERATION	Provides access to commands related to the operation of removable media storage devices such as magnetic tapes.
SYSTEM_ADMINISTRATION	Provides access to commands that perform user validation tasks and permanent file maintenance tasks for the entire system.
SYSTEM_DISPLAYS	Provides access to commands that display system and job information.
SYSTEM_OPERATION	Provides access to commands that perform daily system operation tasks such as managing job activity and managing input and output queues.

Audit Utility 4

Audit Utility 4

The NOS/VE Audit Utility provides the site security administrator with a means of monitoring user operations performed at the site. Examples of operations that can be monitored include permanent file attaches, magnetic tape mounts, initiation of program execution, and changes to user validation records.

The Audit Utility allows you to select the specific types of operations to be audited. Users have no way of knowing which operations, if any, are being monitored.

The Audit Utility is initiated by the ADMINISTER_SECURITY_AUDIT (ADMSA) command, described later in this chapter. You must have the SYSTEM_ ADMINISTRATION capability to use the ADMINISTER_SECURITY_AUDIT command.

The Security Log

The Audit Utility collects audit information using statistics interfaces similar to those used by the NOS/VE Statistics Facility. Like the Statistics Facility interfaces, the statistics activated by the Audit Utility are written to a binary log that can be analyzed using the ANALYZE_BINARY_LOG utility. The information in this chapter assumes that you are familiar with the use of the Statistics Facility and the ANALYZE_BINARY_LOG utility. To read more about them, refer to the Statistics Facility chapter of the NOS/VE System Performance and Maintenance manual, Volume 1.

Because information in the security log can be analyzed with the ANALYZE_BINARY_ LOG utility, you can easily extract information pertaining to individual users, individual terminals, invalid login attempts, or any other selection criteria of interest.

The binary log to which audit information is written is a global log called $SECURITY_LOG. Access to $SECURITY_LOG requires the SYSTEM_ ADMINISTRATION capability. To prevent accidental exposure of sensitive information, audit information selected by the Audit Utility cannot be written to any log other than $SECURITY_LOG.

Audit Statistic Format and Contents

Audit information is written to the security log with the same general format used to write other NOS/VE statistics; that is, each statistic may contain a descriptive data field, a series of counters, or both.

The statistic identifier for the audit statistics is SF. For example, the statistic that records permanent file attaches is SF1000.

An audit statistic provides information about the operation being audited. All audit statistics include the following information:

- Date and time the operation occurred
- Name of the job performing the operation
- Identity of the user performing the operation (SF3000 only)[1]
- Interactive terminal name (SF3000 only)[1]
- Whether or not the operation was successful and, if not, the reason for the failure

Most of the audit statistics provide additional information, as appropriate, for the type of operation being audited. Table 4-1 gives a summary of the audit statistics available and the additional information they provide. The Audit Statistic Formats appendix of this manual contains a detailed description of each statistic format.

Table 4-1. Audited Operations

Operation	Additional Statistic Contents
File Operations	
Attach a file	Object type, device class, file path (including cycle), access modes, and ring attributes (if applicable)
Change attribute	Object type, device class, file path (including cycle), name of attribute (cycle number, ring attributes, FAP name, password or logging) and new value (if appropriate)
Change name	Object type, device class, file path, and new path
Create object	Object type, device class, file path (including cycle), and ring attributes (if applicable)
Create permit	Object type, device class, file path (including cycle), and permit information
Delete object	Object type, device class, file path (including cycle)
Delete permit	Object type, device class, file path (including cycle), and permit information
Magnetic tape mount	External VSN, recorded VSN, write ring, drive identification
User FAP load	File path, library path, module name, procedure name, loaded ring

(Continued)

1. The ANALYZE_BINARY_LOG job predecessor selection criteria allow any NOS/VE statistic to be associated with statistic SF3000.

Table 4-1. Audited Operations *(Continued)*

Operation	Additional Statistic Contents
Job Operations	
Command processing	Command name, command type (program description, SCL procedure, and so on), and an indicator specifying whether or not the command came from the job's command file
Job begin (validate job)	Family, user, account, project, and terminal name
Job end	None
Program execution (task execution)	Name of the starting procedure, complete path of the file containing the module that contains the starting procedure, and ring in which the starting procedure is loaded
Validation Operations	
Activate capability	Capability name
Deactivate capability	Capability name
Create validation record	Record type, user, account and/or project
Delete validation record	Record type, user, account and/or project
Change value of validation field	Record type, user, account and/or project names, and field name
Change validation file security password	Family name
Force validation file security password	Family name
User validation (prevalidation)	Target family, user name, account, project, terminal level, field name, and type
Create validation field	Level, field name, and type
Change validation field definition	Level, field name, and names of changed item(s)
Delete validation field	Level and field name

Using the Audit Utility

Activating and Deactivating Audit Statistics

The Audit Utility subcommands used to activate or deactivate audit statistics are listed below. Except for ACTIVATE_STATISTIC and DEACTIVATE_STATISTIC, these subcommands are used to control the audit statistics described in the Audit Statistic Formats appendix of this manual.

- ACTIVATE_FILE_AUDITING
- DEACTIVATE_FILE_AUDITING
- ACTIVATE_JOB_AUDITING
- DEACTIVATE_JOB_AUDITING
- ACTIVATE_VALIDATION_AUDITING
- DEACTIVATE_VALIDATION_AUDITING
- ACTIVATE_STATISTIC
- DEACTIVATE_STATISTIC

The ACTIVATE_STATISTIC and DEACTIVATE_STATISTIC commands provide a means of writing standard NOS/VE statistics to the security log. The standard NOS/VE statistics are described in the Statistics Facility chapter of the NOS/VE System Performance and Maintenance manual, volume 1.

Defining Audit Activity Selection Criteria

Each of the activation subcommands (except ACTIVATE_STATISTIC) allows you to choose the specific operations you want to activate. The activation subcommands also allow you to define a number of selection criteria to further define and limit the amount of information logged.

The selection criteria used on the activation subcommands allow you to:

- Record successful operations, unsuccessful operations, or both.
- Record all commands processed or just those commands issued from the job's command file.
- Record permanent file operations against the owner's catalogs, an alternate user's catalog, and/or $SYSTEM catalogs.
- Record permanent file operations with a specified access mode (for example, read, write, execute).

Locking the Audit Utility

When you activate an audit activity using the Audit Utility, you have the option of locking it in the ON state. Once it is locked, an audit activity cannot be deactivated by anyone until the system is terminated. To lock an activity in the ON state, specify TRUE for the LOCK parameter on the following Audit Utility subcommands:

- ACTIVATE_FILE_AUDITING
- ACTIVATE_JOB_AUDITING
- ACTIVATE_STATISTIC
- ACTIVATE_VALIDATION_AUDITING

You may have certain audit activities that you want to be active at all times. To ensure that they are active, place the appropriate Audit Utility activation commands in your system initiation prolog with the LOCK parameter set to TRUE.

Displaying Audited Activities

Use the DISPLAY_AUDITED_OPERATIONS subcommand to display a list of operations that are currently being audited.

ADMINISTER _SECURITY_AUDIT Command and Subcommands

ADMINISTER _SECURITY_AUDIT Command

Purpose Calls the utility used to control auditing of security related activities.

Format **ADMINISTER _SECURITY_AUDIT** or
ADMSA
SCOPE=keyword
STATUS=status variable

Parameters *SCOPE* or *S*

Specifies whether the utility affects auditing for the current job or for the entire system. The following values can be specified; the default is SYSTEM:

SYSTEM or S

The utility affects the entire system.

JOB or J

The utility affects only the current job.

ACTIVATE _FILE _AUDITING Subcommand

Purpose Initiates recording of specified file system audit information in the security log.

Format

ACTIVATE _FILE _AUDITING or
ACTFA
LOCK = boolean
OPERATION = list of keyword
RESULT = keyword
CATALOG _OWNER = list of keyword
ACCESS _MODE = list of keyword
STATUS = status variable

Parameters

LOCK or **L**

Specifies a boolean value indicating whether or not the audit operations are locked (that is, cannot be deactivated). This parameter is required.

TRUE

Audit operations are locked.

FALSE

Audit operations are not locked.

OPERATION or *OPERATIONS* or *O*

Specifies which file system operations are to be recorded. One or more of the following keyword values can be specified; the default is ALL:

ATTACH _FILE or AF

Records permanent file attaches.

LOAD _FAP or LF

Records loading of FAPs.

MANAGE _OBJECT or MO

Activates auditing of the following operations:

- Creation of permanent files and catalogs
- Deletion of permanent files and catalogs
- Changes to cycle numbers, ring attributes, passwords, FAPs, or logging

MANAGE _PERMIT or MP

Records creation and deletion of permits.

MOUNT_REMOVABLE _MEDIA or MRM

Records mounts of removable media.

ALL

Records all file system audit operations.

RESULT or *R*

Specifies the result selection criteria for recording file system operations. The following keyword values can be specified; the default is ALL:

SUCCESSFUL or S

Records only successful operations.

UNSUCCESSFUL or U

Records only unsuccessful operations.

ALL

Records both successful and unsuccessful operations.

CATALOG_OWNER or *CO*

Specifies catalog owner selection criteria for recording permanent file operations. One or more of the following keywords can be specified; the default is ALL:

OWNER or O

Records operations that affect any files or catalogs owned by the user performing the operation.

NON_OWNER or NO

Records operations that affect any files or catalogs that are not owned by the user performing the operation. Exception: This option does not audit accesses of files or catalogs under the $SYSTEM user name.

SYSTEM or S

Records operations that affect any files or catalogs that reside under the $SYSTEM user name.

ALL

Records operations that affect any files or catalogs residing in any catalog.

ACCESS_MODE or *ACCESS_MODES* or *AM*

Specifies access mode selection criteria to be used for recording permanent file attaches. Permanent file attaches are only recorded when the access mode specified on the attach request intersects with the set of access modes specified for this parameter. One or more of the following keywords can be specified; the default is ALL:

APPEND or A

Records attaches that include APPEND as one of their access modes.

EXECUTE or E

Records attaches that include EXECUTE as one of their access modes.

MODIFY or M

Records attaches that include MODIFY as one of their access modes.

READ or R
Records attaches that include READ as one of their access modes.

SHORTEN or S
Records attaches that include SHORTEN as one of their access modes.

WRITE or W
Same as specifying APPEND, MODIFY, and SHORTEN.

ALL

Records all file attaches.

ACTIVATE_JOB_AUDITING Subcommand

Purpose Initiates recording of specified job activities in the security log.

Format

ACTIVATE_JOB_AUDITING or
ACTJA
LOCK=boolean
OPERATION=list of keyword
RESULT=keyword
COMMAND_SOURCE=keyword
STATUS=status variable

Parameters

LOCK or L

Specifies a boolean value indicating whether or not audit operations are locked (that is, cannot be deactivated). This parameter is required.

TRUE

Audit operations are locked.

FALSE

Audit operations are not locked.

OPERATION or *OPERATIONS* or *O*

Specifies which operations relating to job activities are to be recorded. One or more of the following keyword values may be specified; the default is ALL:

EXECUTE_PROGRAM or EP

Records program execution (task execution).

PROCESS_COMMAND or PC

Records commands processed by a job.

ALL

Records all job audit operations.

RESULT or *R*

Specifies result selection criteria for recording job operations. Keyword values that can be specified are:

SUCCESSFUL or S

Records only successful operations.

UNSUCCESSFUL or U

Records only unsuccessful operations.

ALL

Records both successful and unsuccessful operations.

COMMAND_SOURCE or *CS*

Specifies whether command processing is audited. for all commands or for only those commands entered from the job's command file. This parameter is valid only if the PROCESS_COMMAND option is specified on the OPERATION parameter. SCL control statements (for example, IF and FOR) are never audited. The following keyword values can be specified; the default is PRIMARY_COMMANDS:

ALL_COMMANDS or AC

Records all commands processed by SCL. This includes commands that are processed as a result of executing an SCL procedure or an INCLUDE_FILE command.

PRIMARY_COMMANDS or PC

Records all commands processed by SCL except those executed from an SCL procedure or an INCLUDE_FILE command.

ACTIVATE_STATISTIC Subcommand

Purpose Initiates recording of specified statistics in the security log. This command can be used to record any NOS/VE statistics in the security log.

Format

ACTIVATE_STATISTICS or
ACTS
LOCK=boolean
STATISTIC=list of statistic codes
STATUS=status variable

Parameters

LOCK or **L**

Specifies a boolean value indicating whether or not the specified statistics are locked (that is, cannot be deactivated). This parameter is required.

TRUE

Audit operations are locked.

FALSE

Audit operations are not locked.

STATISTIC or **STATISTICS** or **S**

Specifies the list of statistic codes to be recorded in the security log. This parameter is required.

ACTIVATE_VALIDATION_AUDITING Subcommand

Purpose Initiates recording of specified validation activities in the security log.

Format

ACTIVATE_VALIDATION_AUDITING or
ACTVA
LOCK=boolean
OPERATION=list of keyword
RESULT=keyword
STATUS=status variable

Parameters

LOCK or **L**

Specifies a boolean value indicating whether or not the specified audit operations are locked (that is, cannot be deactivated). This parameter is required.

TRUE

Audit operations are locked.

FALSE

Audit operations are not locked.

OPERATION or *OPERATIONS* or *O*

Specifies which operations relating to validations are to be recorded. One or more of the following keywords can be specified; the default is ALL:

MANAGE_FIELD or MF

Records operations that create, delete, or modify validation fields.

MANAGE_SECURITY_PASSWORD or MSPW

Records changes to the security password for a validation file.

MANAGE_VALIDATION or MV

Records operations that create, delete, or modify validations at the user, account, account member, project, or project member levels.

USE_CAPABILITY or UC

Records operations that activate or deactivate any of the following capabilities:

ACCOUNTING_ADMINISTRATION
CONFIGURATION_ADMINISTRATION
FAMILY_ADMINISTRATION
REMOVABLE_MEDIA_ADMINISTRATION
REMOVABLE_MEDIA_OPERATION
SYSTEM_ADMINISTRATION
SYSTEM_DISPLAYS
SYSTEM_OPERATION

USER_VALIDATION or UV

Records requests to validate a user name (prevalidation requests).

ALL

Records all validation audit operations.

RESULT or *R*

Specifies result selection criteria for recording validation operations. The following keyword values can be specified; the default is ALL:

SUCCESSFUL or S

Records successful operations only.

UNSUCCESSFUL or U

Records unsuccessful operations only.

ALL

Records both successful and unsuccessful operations.

DEACTIVATE _FILE _AUDITING Subcommand

Purpose Terminates recording of specified file system activities in the security log.

Format

DEACTIVATE _FILE _AUDITING or
DEAFA
OPERATION=list of keyword
STATUS=status variable

Parameters

OPERATION or **OPERATIONS** or **O**

Specifies which operations relating to file accesses should be deactivated. This parameter is required. One or more of the following keywords can be specified:

ATTACH _FILE or AF

Deactivates recording of permanent file attaches.

LOAD _FAP or LF

Deactivates recording of operations that load FAPs.

MANAGE _OBJECT or MO

Deactivates recording of the following operations:

- Creation of permanent files and catalogs
- Deletion of permanent files and catalogs
- Changes to cycle numbers, ring attributes, passwords, FAPs, or statistics logs

MANAGE _PERMIT or MP

Deactivates recording of operations that create or delete file permits.

MOUNT_REMOVABLE _MEDIA or MRM

Deactivates recording of removable media mounts.

ALL

Deactivates all file system audit operations.

DEACTIVATE_JOB_AUDITING Subcommand

Purpose Terminates recording of specified job activities in the security log.

Format

DEACTIVATE_JOB_AUDITING or
DEAJA
OPERATION=list of keyword
STATUS=status variable

Parameters

OPERATION or **OPERATIONS** or **O**

Specifies which operations relating to job activities should be deactivated. This parameter is required. One or more of the following keywords can be specified:

EXECUTE_PROGRAM or EP

Deactivates recording of program execution (task execution).

PROCESS_COMMAND or PC

Deactivates recording of commands processed by a job.

ALL

Deactivates recording of all job audit operations.

DEACTIVATE _STATISTIC Subcommand

Purpose Terminates recording of specified statistics in the security log.

Format

DEACTIVATE _STATISTIC or
DEAS
STATISTIC=list of statistic codes
STATUS=status variable

Parameters **STATISTIC** or **STATISTICS** or **S**

Specifies the list of statistic codes to be deactivated. This parameter is required.

DEACTIVATE _VALIDATION _AUDITING Subcommand

Purpose Terminates recording of specified validation activities in the security log.

Format

DEACTIVATE _VALIDATION _AUDITING or
DEAVA
OPERATION=list of keyword
STATUS=status variable

Parameters

OPERATION or **OPERATIONS** or **O**

Specifies which operations relating to validation activities are to be deactivated. This parameter is required. One or more of the following keywords can be specified:

MANAGE_FIELD or MF

Deactivates recording of operations that create, delete, or modify validation fields.

MANAGE_SECURITY_PASSWORD or MSPW

Deactivates recording of changes to the security password for a validation file.

MANAGE_VALIDATION or MV

Deactivates recording of operations that create, delete, or modify validations at the user, account, account member, project, and project member levels.

USE_CAPABILITY or UC

Deactivates recording of operations that activate or deactivate any of the following capabilities:

ACCOUNTING_ADMINISTRATION
CONFIGURATION_ADMINISTRATION
FAMILY_ADMINISTRATION
REMOVABLE_MEDIA_ADMINISTRATION
REMOVABLE_MEDIA_OPERATION
SYSTEM_ADMINISTRATION
SYSTEM_DISPLAYS
SYSTEM_OPERATION

USER_VALIDATION or UV

Deactivates recording of requests to validate a user name (prevalidation requests).

ALL

Deactivates recording of all validation audit operations.

DISPLAY_AUDITED_OPERATIONS Subcommand

Purpose Displays the list of operations being recorded in the security log.

Format

DISPLAY_AUDITED_OPERATIONS or
DISAO
OUTPUT=file
STATUS=status variable

Parameters *OUTPUT* or *O*

Specifies the name of the file to receive the display output. The default file name is $OUTPUT.

Related Manuals A

Related Manuals A

Table A-1 lists the titles of all manuals referenced in this manual. The table also includes the titles of any other system, product, or hardware manuals that are directly related to this manual. For a complete list of NOS/VE manuals available, see the Related Manuals appendix of the NOS/VE System Usage manual.

If your site has installed the online manuals, you can find an abstract of each NOS/VE manual in the online System Information manual. To access this manual, enter:

```
/help manual=nos_ve
```

Ordering Printed Manuals

To order a printed Control Data manual, send an order form to:

Control Data
Literature and Distribution Services
308 North Dale Street
St. Paul, Minnesota 55103-2495

To obtain an order form or to get more information about ordering Control Data manuals, write to the above address or call (612) 292-2101. If you are a Control Data employee, call (612) 292-2100.

Accessing Online Manuals

To access an online NOS/VE manual, log in to NOS/VE and enter the HELP command, specifying the online title of the manual. The online titles are listed in table A-1. For example, to see the Site Analyst Examples manual, enter:

```
/help manual=site_analyst_examples
```

Table A-1. Related Manuals

Manual Title	Publication Number	Online Title
NOS/VE Operations Usage	60463914	
NOS/VE System Performance and Maintenance Volume 1: Performance Usage	60463915	
NOS/VE User Validation Usage	60464513	
NOS/VE System Usage Usage	60464014	EXAMPLES
NOS/VE Commands and Functions Usage	60464018	SCL

Audit Statistic Formats B

Audit Statistic Formats B

This appendix describes the formats, descriptive data fields, and counters for the audit statistics controlled by the Audit Utility. To record these statistics you must first turn on the SECURITY_AUDIT security option using the SET_SECURITY_OPTION system core command. Then use the Audit Utility to select the operations you want to audit.

The following table lists the individual statistics described in this appendix.

Audit Statistic	Description
SF1000	Attach file
SF1001	Change selected file attibutes
SF1002	Change the name of a permanent file
SF1003	Create the file or catalog
SF1004	Create permit
SF1005	Delete file or catalog
SF1006	Delete permit
SF1007	Load file access procedure (FAP)
SF1008	Mount magnetic tape
SF2000	Activate capabilities controlled by SOU
SF2001	Change selected information in a validation field description
SF2002	Change validation field name
SF2003	Change validation record
SF2004	Change the security password for a validation file
SF2005	Create validation field
SF2006	Create validation record
SF2007	Deactivate capabilities controlled by SOU
SF2008	Delete validation field
SF2009	Delete validation record
SF2010	Force a new security password on a validation file
SF2012	Request to validate a user
SF3000	User identification
SF3001	Job end
SF3002	Execute program (task)
SF3003	Process command

Statistic Name	Definition

SF1000

Records the attachment of a permanent file. The descriptive data are in the following format:

operation,result,path,objecttype,deviceclass,accessmodes

Field	Description
Operation	Contains the operation abbreviation ATTF.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Path	Contains the complete path name of the file.
Object type	Contains the value CYCLE.
Device class	Indicates the device type on which the file cycle resides; contains one of the following values: MAGNETIC_TAPE or MASS_STORAGE.

Access modes	Indicates the access modes requested on the attach. Each possible access mode is reported in a fixed position within an 8-character field. The field is in the form: (RWESAM). Each letter in the field represents the first letter of one of the access modes (Read, Write, Execute, Shorten, Append, Modify). If a particular access mode is not specified on the attach request, its position in the field is blank. If any of the values Shorten, Append, or Modify is specified on the attach request, Write access will also be reported. For example, an access mode request of (read,execute) produces the field:
	`(R E  )`
	a request of (all) produces:
	`(RWESAM)`
	and a request of (append) produces:
	`( W  A )`

Statistic Name	Definition
SF1001	Records changes to the following file attributes: cycle number, logging, password, ring attributes, and file access procedure name. The descriptive data are in the following format:

operation,result,path,objecttype,deviceclass,attributename,newvalue

Field	Description
Operation	Contains the operation abbreviation CHAFA.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Path	Contains the complete path name of the file.
Object type	Contains one of the following values: FILE or CYCLE.
Device class	Indicates the device type on which a file cycle resides. If the object type is CYCLE, this field contains one of the following values: MAGNETIC_TAPE or MASS_STORAGE. If the object type is not CYCLE, this field is blank.
Attribute name	Identifies the attribute that was changed; contains one of the following values: CYCLE_NUMBER, LOGGING, PASSWORD, RING_ATTRIBUTES, or FAP_NAME.
New value	Contains the new cycle number, logging value (boolean), ring attributes (in the form (r1 r2 r3), or FAP name. The value of a new password is never recorded.

Statistic Name	Definition
SF1002	Records the changing of a file or catalog name. The descriptive data are in the following format: operation,result,path,objecttype,deviceclass,newpath

Field	Description
Operation	Contains the operation abbreviation CHAON.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Path	Contains the old complete path name of the file.
Object type	Contains one of the following values: CATALOG or FILE.
Device class	This field is blank.
New path	Specifies the new complete path name of the file.

Statistic Name	Definition
SF1003·	Records the creation of a file or catalog. The descriptive data are in the following format:

operation,result,path,objecttype,deviceclass

Field	Description
Operation	Contains the operation abbreviation CREO.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Path	Contains the complete path name of the catalog or file.
Object type	Contains one of the following values: CATALOG or CYCLE.
Device class	Indicates the device type on which a file cycle resides. If the object type is CYCLE, this field contains one of the following values: MAGNETIC_TAPE or MASS_STORAGE. If the object type is not CYCLE, this field is blank.

Statistic Name	Definition
SF1004	Records the creation of a permit for a file or catalog. The descriptive data are in the following format: operation,result,path,objecttype,deviceclass,group,family,user, account,project,accessmodes

Field	Description
Operation	Contains the operation abbreviation CREP.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Path	Contains the complete path name of the catalog or file.
Object type	Contains one of the following values: CATALOG or FILE.
Device class	This field is blank.
Group	Indicates the group to which the permit applies; contains one of the following values: PUBLIC, FAMILY, USER, ACCOUNT, ACCOUNT_MEMBER, PROJECT, or PROJECT_MEMBER.
Family	Contains the permitted family name, if applicable.
User	Contains the permitted user name, if applicable.
Account	Contains the permitted account name, if applicable.
Project	Contains the permitted project name, if applicable.

Statistic Name	Definition	
SF1004 (cont'd)	**Field**	**Description**

	Access modes	Indicates the access modes allowed by the permit. Each possible access mode is reported in a fixed position within a 10-character field. The field is in the form: (RWESAMCC). Each letter in the field represents the first letter of one of the access modes (Read, Write, Execute, Shorten, Append, Modify, Cycle, Control).
		
		If a particular access mode is not specified on the permit request, its position in the field is blank. If any of the values Shorten, Append, or Modify is specified on the attach request, Write access will also be reported.
		
		For example, access mode permits of (read,execute) produce a recorded field of:
		
		`(R E     )`
		
		permits of (cycle) produce:
		
		`(       C )`
		
		and permits of (shorten,append) produce:
		
		`( W SA   )`

Statistic Name	Definition
SF1005	Records the deletion of a file or catalog. The descriptive data are in the following format: operation,result,path,objecttype,deviceclass

Field	Description
Operation	Contains the operation abbreviation DELO.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Path	Contains the complete path name of the catalog or file.
Object type	Contains one of the following values: CATALOG or CYCLE.
Device class	Indicates the device type on which a file cycle resides. If the object type is CYCLE, this field contains one of the following values: MAGNETIC_TAPE or MASS_STORAGE. If the object type is not CYCLE, this field is blank.

Statistic Name	Definition
SF1006	Records the deletion of a permit to a file or catalog. The descriptive data are in the following format:

operation,result,path,objecttype,deviceclass,group,family,user,
account,project

Field	Description
Operation	Contains the operation abbreviation DELP.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Path	Contains the complete path name of the catalog or file.
Object type	Contains one of the following values: CATALOG or FILE.
Device class	This field is blank.
Group	Indicates the group to which the permit applies; contains one of the following values: PUBLIC, FAMILY, USER, ACCOUNT, ACCOUNT_MEMBER, PROJECT, or PROJECT_MEMBER.
Family	Contains the permitted family name, if applicable.
User	Contains the permitted user name, if applicable.
Account	Contains the permitted account name, if applicable.
Project	Contains the permitted project name, if applicable.

Statistic Name	Definition
SF1007	Records the loading of a file access procedure. The descriptive data are in the following format: operation,result,path,objecttype,deviceclass,fapname,fapmodule, faplibrary,loadedring

Field	Description
Operation	Contains the operation abbreviation LOAFAP.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Path	Contains the complete path name of the catalog or file.
Object type	Contains the value CYCLE.
Device class	Contains one of the following values: MAGNETIC_TAPE or MASS_STORAGE.
FAP name	Contains the name of the FAP.
FAP module	Contains the name of the module containing the FAP.
FAP library	Contains the path name of the file containing the FAP.
Loaded ring	Indicates the ring in which the FAP executes.

Statistic Name	Definition
SF1008	Records a magnetic tape mount. The descriptive data are in the following format:

operation,result,externalvsn,recordedvsn,writering,elementname

Field	Description
Operation	Contains the operation abbreviation MOUMT.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
External VSN	Contains the external VSN from the tape request.
Recorded VSN	Contains the VSN recorded on the tape.
Write ring	Indicates whether or not a write ring was requested.
Element name	Indicates the tape drive on which the tape was mounted.

Statistic Name	Definition
SF2000	Records activation of a conditional (SOU) capability. The descriptive data are in the following format: operation,result,fieldname

Field	Description
Operation	Contains the operation abbreviation ACTC.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Field name	Contains the name of the capability being activated.

Statistic Name	Definition
SF2001	Records changes to selected information in a validation field description. The descriptive data are in the following format: operation,result,recordtype,validationfile,fieldname,attributename, newvalue

Field	Description
Operation	Contains the operation abbreviation CHAVF.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Record type	Indicates the type of validation record that was changed; contains one of the following values: USER, ACCOUNT, ACCOUNT_MEMBER, PROJECT, or PROJECT_MEMBER.
Validation file	Contains the complete path for the validation file being changed.
Field name	Contains the name of the validation field that was changed.
Attribute name	Indicates the name of the validation field attribute; contains one of the following values: DEFAULT_VALUE, DISPLAY_AUTHORITY, CHANGE_AUTHORITY, or MANAGE_AUTHORITY.
New value	Contains the new value for the display, change, or manage authority. New default values are never recorded.

Statistic Name	Definition
SF2002	Records a validation field name change. The descriptive data are in the following format: operation,result,recordtype,validationfile,oldfieldname, newfieldname

Field	Description
Operation	Contains the operation abbreviation CHAVFN.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Record type	Indicates the type of validation record that was changed; contains one of the following values: USER, ACCOUNT, ACCOUNT_MEMBER, PROJECT, or PROJECT_MEMBER.
Validation file	Contains the complete path for the validation file being changed.
Old field name	Contains the name of the validation field that was changed.
New field name	Contains the new name for the validation field.

Statistic Name	Definition
SF2003	Records changes to validation records. The descriptive data are in the following format: operation,result,recordtype,validationfile,user,account,project,field-name

Field	Description
Operation	Contains the operation abbreviation CHAVR.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Record type	Indicates the type of validation record that was changed; contains one of the following values: USER, ACCOUNT, ACCOUNT_MEMBER, PROJECT, or PROJECT_MEMBER.
Validation file	Contains the complete path for the validation file being changed.
User	Contains the user name, if applicable.
Account	Contains the account name, if applicable.
Project	Contains the project name, if applicable.
Field name	Contains the name of the validation field that was changed.

Statistic Name	Definition
SF2004	Records changes to the security password associated with a validation file. The descriptive data are in the following format: operation,result,validationfile

Field	Description
Operation	Contains the operation abbreviation CHASPW.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Validation file	Contains the complete path for the validation file being changed.

Statistic Name	Definition
SF2005	Records the creation of a validation field. The descriptive data are in the following format: operation,result,recordtype,validationfile,fieldname,fieldtype

Field	Description
Operation	Contains the operation abbreviation CREVF.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Record type	Indicates the type of validation record that was changed; contains one of the following values: USER, ACCOUNT, ACCOUNT_MEMBER, PROJECT, or PROJECT_MEMBER.
Validation file	Contains the complete path for the validation file being changed.
Field name	Contains the name of the validation field that was created.
Field type	Indicates the type of the new validation field; contains one of the following values: ACCUMULATING_LIMIT, CAPABILITY, DATE_TIME, FILE, INTEGER, LIMIT, NAME, REAL, STRING.

Statistic Name	Definition
SF2006	Records the creation of a validation record. The descriptive data are in the following format: operation,result,recordtype,validationfile,user,account,project

Field	Description
Operation	Contains the operation abbreviation CREVR.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Record type	Indicates the type of validation record that was created; contains one of the following values: USER, ACCOUNT, ACCOUNT_MEMBER, PROJECT, or PROJECT_MEMBER.
Validation file	Contains the complete path for the validation file being changed.
User	Contains the user name, if applicable.
Account	Contains the account name, if applicable.
Project	Contains the project name, if applicable.

Statistic Name	Definition
SF2007	Records the deactivation of a conditional (SOU) capability. The descriptive data are in the following format: operation,result,fieldname

Field	Description
Operation	Contains the operation abbreviation DEAC.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Field name	Contains the name of the capability that was deactivated.

Statistic Name	Definition
SF2008	Records the deletion of a validation field. The descriptive data are in the following format: operation,result,recordtype,validationfile,fieldname

Field	Description
Operation	Contains the operation abbreviation DELVF.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Record type	Indicates the type of validation record that was deleted; contains one of the following values: USER, ACCOUNT, ACCOUNT_MEMBER, PROJECT, or PROJECT_MEMBER.
Validation file	Contains the complete path for the validation file being changed.
Field name	Contains the name of the validation field that was deleted.

Statistic Name	Definition
SF2009	Records the deletion of a validation record. The descriptive data are in the following format: operation,result,recordtype,validationfile,user,account,project

Field	Description
Operation	Contains the operation abbreviation DELVR.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Record type	Indicates the type of validation record that was deleted; contains one of the following values: USER, ACCOUNT, ACCOUNT_MEMBER, PROJECT, or PROJECT_MEMBER.
Validation file	Contains the complete path for the validation file being changed.
User	Contains the user name, if applicable.
Account	Contains the account name, if applicable.
Project	Contains the project name, if applicable.

Statistic Name	Definition
SF2010	Records forced changes to the security password associated with a validation file. A "forced change" means that the security password is changed without specifying the old password value. The descriptive data are in the following format: operation,result,validationfile

Field	Description
Operation	Contains the operation abbreviation FORSPW.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Validation file	Contains the complete path for the validation file being changed.

Statistic Name	Definition
SF2012	Records requests to validate a user. The descriptive data are in the following format: operation,result,family,user,account,project,terminal

Field	Description
Operation	Contains the operation abbreviation VALU.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Family	Contains the family name.
User	Contains the user name from the validation request.
Account	Contains the account name from the validation request.
Project	Contains the project name from the validation request.
Terminal	Contains the name of the interactive terminal from which the request was made. If the request was not made from an interactive job, this field is blank.

Statistic Name	Definition
SF3000	Records user identification information. The descriptive data are in the following format: operation,result,family,user,account,project,terminal

Field	Description
Operation	Contains the operation abbreviation USERID.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Family	Contains the family name.
User	Contains the user name.
Account	Contains the account name.
Project	Contains the project name.
Terminal	Contains the name of the interactive terminal at which the user is logged in. If the user's job is not an interactive job, this field is blank.

Statistic Name	Definition
SF3001	Records the end of a job. This statistic has no counters or descriptive data.

Statistic Name	Definition
SF3002	Records execution of a program (task). The descriptive data are in the following format: operation,result,programname,modulename,libraryname,loadedring

Field	Description
Operation	Contains the operation abbreviation EXEP.
Result	Contains the status for the initiation of the task (not the status returned by the task when it completes). This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Program name	Contains the name of the starting procedure.
Module name	Contains the name of the module containing the starting procedure.
Library name	Contains the name of the library containing the starting procedure.
Loaded ring	Contains the ring number of the ring in which the starting procedure was loaded.

Statistic Name	Definition
SF3003	Records the processing of a command. The descriptive data are in the following format: operation,result,commandname,commandsource,callmethod

Field	Description
Operation	Contains the operation abbreviation PROC.
Result	This field is blank if the operation was successful. If unsuccessful, this field contains the status condition identifying the reason for the failure.
Command name	Contains the name of the command that was processed.
Command source	Indicates whether the command came from the job's command file or from an alternate command file (that is, from a procedure or an INCLUDE_FILE file); contains one of the following values: PRIMARY (job command file) or SECONDARY (alternate command file).
Call method	Indicates the type of command call; contains one of the following values: LINKED, UNLINKED, PROCEDURE, or PROGRAM.

Data Encryption Standard C

Data Encryption Standard C

Overview of the DES User Interface

The Data Encryption Standard (DES) is a file encryption standard defined by Federal Information Processing Standard Publication 46, which is published by the National Bureau of Standards. This chapter describes the file encryption user interface available to application programmers on NOS/VE.

The file encryption interface consists of two SCL commands:

- ENCRYPT_FILE
- DECRYPT_FILE

and two CYBIL interfaces:

- EDP$ENCRYPT_DATA_USING_DES
- EDP$DECRYPT_DATA_USING_DES

Data Encryption Standard Modes of Operation

You can select from four modes of operation for encrypting file data. The mode of operation is specified on the DES_MODE parameter of the ENCRYPT_FILE or DECRYPT_FILE command. The four modes are as follows:

- Electronic Codebook (ECB) Mode
- Cipher Block Chaining (CBC) Mode
- Cipher Feedback (CFB) Mode
- Output Feedback (OFB) Mode

Electronic Codebook (ECB) Mode

The electronic codebook (ECB) mode is the basic block cipher method specified by Federal Information Processing Standards Publication 46. The other three modes of operation are extensions of this mode. ECB mode transforms 64 bits of input to 64 bits of output. The same 64 bits of input always produces the same 64 bits of output for a given cryptographic key when the attributes are preserved. Attributes are preserved for an encrypted file when a value of TRUE is specified on the PRESERVE_ATTRIBUTES parameter of the ENCRYPT_FILE command.

Cipher Block Chaining (CBC) Mode

The cipher block chaining (CBC) mode is a block cipher method in which the first plain text, 64-bit data block is combined with an initial vector prior to being processed through the DES basic algorithm. The initial vector is a value defined by the DES_INITIAL_VECTOR parameter of the ENCRYPT_FILE command. The result of that encryption becomes the first encrypted data block and the next initial vector. The process is then repeated using the next 64-bit block of input data.

Cipher Feedback (CFB) Mode

The cipher feedback (CFB) mode is a stream method of encryption that passes the initial vector data block through the DES algorithm and then combines the result with the plain text in order to get the first block of cipher text. The cipher text is then passed back to become the next initial vector.

Output Feedback (OFB) Mode

The output feedback (OFB) mode is an additive stream cipher method that passes the initial vector data block through the DES algorithm. The result of that encryption is passed back to become the next initial vector and is also combined with the plain text data block to become the first block of cipher text.

Integer Formats for Cryptographic Keys

A cryptographic key can be defined in either binary or hexadecimal integer format. For either format, the length of the key is 64 bits.

Binary format:

(B1,B2,...,B7,P1,B8,B9,...,B14,P2,/ ... /,B50,B51,...,B56,P8)

Bits B1 through B56 are independant bits of a DES key. Bits P1 through P8 are reserved for parity bits computed on the preceding seven independent bits. The parity of each octet is set to odd; that is, there will be an odd number of 1-bits in each octet.

Hexadecimal format:

(H1H2H3..H16)

H1 through H16 are any hexadecimal characters (0 through F). Characters can be uppercase or lowercase; embedded blanks are not allowed.

Data Encryption SCL Commands

ENCRYPT_FILE Command

Purpose Encrypts the contents of a file using the National Bureau of Standards Data Encryption Standard algorithm.

Format

ENCRYPT_FILE or
ENCF
INPUT=file
OUTPUT=file
DES_KEY=integer or **string** or **name**
DES_MODE=keyword
DES_INITIAL_VECTOR=integer or *string* or *name*
PRESERVE_ATTRIBUTES=boolean
STATUS=status variable

Parameters

INPUT or **I**

Specifies the file containing the data to be encrypted. The value specified may define how the file is positioned prior to use. Data is encrypted from the open position until end-of-information is reached. This parameter is required.

OUTPUT or **O**

Specifies the file to which the encrypted data is to be written. The value specified may define how the file is positioned prior to use. This parameter is required.

The file attributes of the output file are set to system defaults except for the following attributes:

file_organization:	amc$sequential
file_structure:	amc$data
file_content:	fsc$unknown_contents
file_processor:	fsc$decrypt_file (if file attributes are preserved) or amc$unknown_processor
record_type:	amc$variable
block_type:	amc$system_specified

DES_KEY or **DK**

Defines a cryptographic key to be used by the DES algorithm. The value specified must be an integer, a string value, or a name. This parameter is required.

If an integer value is specified, each byte of the binary representation of the integer must have odd parity. For more information on integer formats, see the section in this appendix called Integer Formats for Cryptographic Keys.

If a string value is used, the string must be exactly 8 characters long, and each character must be a member of the ASCII 128-character set.

If a name value is used, the name must be exactly 8 characters long and must be a valid NOS/VE name.

DES_MODE or *DM*

Specifies the DES encryption mode to be used. The following keywords can be specified; the default value is ELECTRONIC_CODEBOOK:

ELECTRONIC CODEBOOK or ECB

Uses the basic Data Encryption Standard algorithm.

CIPHER_BLOCK_CHAINING or CBC

Uses the cipher block chaining mode of operation.

CIPHER_FEEDBACK or CFB

Uses the cipher feedback mode of operation. The length of the feedback data unit is 64 bits.

OUTPUT_FEEDBACK or OFB

Uses the output feedback mode of operation. The length of the feedback data unit is 64 bits.

DES_INITIAL_VECTOR or *DIV*

Defines a seed value required by the cipher block chaining, cipher feedback, or output feedback mode of operation. If the DES_MODE parameter specifies ELECTRONIC_CODEBOOK, this parameter is ignored; otherwise, this parameter is required. The value specified can be an integer, a string, or a name.

If an integer value is specified, each byte of the binary representation of the integer must have odd parity.

If a string value is used, the string must be exactly 8 characters long, and each character must be a member of the ASCII 128-character set.

If a name value is used, the name must be exactly 8 characters long and must be a valid NOS/VE name.

PRESERVE_ATTRIBUTES or *PA*

Specifies whether or not the NOS/VE file attributes of the input file are to be included (preserved) as data on the output file. (The DECRYPT_FILE command is able to restore the original file's attributes if the attributes are preserved during encryption.) The default value is TRUE. If this parameter specifies a value of FALSE, the length of the input file must be a multiple of 8 bytes.

TRUE or ON or YES

Preserves the input file attributes on the output file.

FALSE or OFF or NO

The original file attributes are not saved.

DECRYPT_FILE Command

Purpose Decrypts the contents of a file that has been encrypted using the ENCRYPT_FILE command.

Format

DECRYPT_FILE or
DECF
INPUT=file
OUTPUT=file
DES_KEY=integer or **string** or **name**
DES_MODE=keyword
DES_INITIAL_VECTOR=integer or *string* or *name*
RESTORE_ATTRIBUTES=boolean
STATUS=status variable

Parameters

INPUT or I

Specifies the file containing the data to be decrypted. The value specified may define how the file is positioned prior to use. Data is decrypted from the open position until end-of-information is reached. This parameter is required.

OUTPUT or O

Specifies the file to which the decrypted data is to be written. The value specified may define how the file is positioned prior to use. This parameter is required.

The file attributes of the output file are set to the original file attributes if RESTORE_ATTRIBUTES=TRUE was specified and the file contains preserved attributes. Otherwise, the attributes are set as follows:

file_organization:	amc$sequential
file_structure:	amc$data
file_content:	fsc$unknown_contents
file_processor:	amc$unknown_processor
record_type:	amc$variable
block_type:	amc$system_specified

DES_KEY or DK

Defines a cryptographic key to be used by the DES algorithm. The value specified must be an integer, a string value, or a name. This parameter is required.

If an integer value is specified, each byte of the binary representation of the integer must have odd parity. For more information on integer formats, see the section in this chapter called Integer Formats for Cryptographic Keys.

If a string value is used, the string must be exactly 8 characters long, and each character must be a member of the ASCII 128-character set.

If a name value is used, the name must be exactly 8 characters long and must be a valid NOS/VE name.

DES_MODE or *DM*

Specifies the DES decryption mode to be used. The following keywords can be specified; the default value is ELECTRONIC_CODEBOOK:

ELECTRONIC CODEBOOK or ECB

Uses the basic Data Encryption Standard algorithm.

CIPHER_BLOCK_CHAINING or CBC

Uses the cipher block chaining mode of operation.

CIPHER_FEEDBACK or CFB

Uses the cipher feedback mode of operation. The length of the feedback data unit is 64 bits.

OUTPUT_FEEDBACK or OFB

Uses the output feedback mode of operation. The length of the feedback data unit is 64 bits.

DES_INITIAL_VECTOR or *DIV*

Defines a seed value required by the cipher block chaining, cipher feedback, or output feedback mode of operation. The value specified must be the same value specified on the DES_INITIAL_VECTOR parameter of the ENCRYPT_FILE command. If the DES_MODE parameter specifies ELECTRONIC_CODEBOOK, this parameter is ignored; otherwise, this parameter is required.

RESTORE_ATTRIBUTES or *RA*

Specifies whether or not the NOS/VE file attributes of the input file are to be restored to the decrypted file. The default value is TRUE. If this parameter specifies a value of FALSE, the length of the input file must be a multiple of 8 bytes. An error will be returned if TRUE is specified for a file whose file attributes were not preserved during encryption.

TRUE or ON or YES

The input file contains preserved attributes which are to be restored to the output file.

FALSE or OFF or NO

Either the original file attributes were not preserved for the input file or the file attributes are to be ignored.

Data Encryption CYBIL Interfaces

EDP$ENCRYPT_DATA_USING_DES

Purpose Converts a cell pointer to an adaptable array pointer and calls the FORTRAN DES subroutines to encrypt the block of data to which the cell points.

Format **EDP$ENCRYPT_DATA_USING_DES (data, length, key, mode, feedback_unit_length, initial_vector, status)**

Parameters

data: VAR { input-output } of ^cell

Specifies the cell pointer to the data block.

length: VAR { input-output } of integer

Specifies the length of the data block in words.

key: integer

Defines a cryptographic key of type integer. Each byte of the integer must consist of seven key bits and one bit of odd parity. For more information on the integer format of a cryptographic key, see the section in this appendix called Integer Formats for Cryptographic Keys.

mode: edt$des_encryption_mode

Specifies the DES encryption mode. Any of the following strings may be specified:

'ELECTRONIC_CODE_BOOK'
'ECB'
'CIPHER_BLOCK_CHAINING'
'CBC'
'CIPHER_FEEDBACK'
'CFB'
'OUTPUT_FEEDBACK'
'OFB'

feedback_unit_length: integer

Specifies the number of bits fed back into the DES algorithm when using CIPHER_FEEDBACK or OUTPUT_FEEDBACK modes. The value specified can be an integer in the set: (0, 1, 7, 8, 56, 64, where 0=64)

initial_vector: VAR { input-output } of integer

Defines a seed value required by all modes except ELECTRONIC_CODE_BOOK mode. The value is processed by the encryption algorithm and a modified value is returned. The returned value may be used on subsequent calls to this procedure.

status: VAR { output } of ost$status

Returns the request status.

EDP$DECRYPT_DATA_USING_DES

Purpose Converts a cell pointer to an adaptable array pointer and calls the FORTRAN DES subroutines to decrypt the block of data to which the cell points.

Format **EDP$DECRYPT_DATA_USING_DES (data, length, key, mode, feedback_unit_length, initial_vector, status)**

Parameters

data: VAR { input-output } of ^cell

Specifies the cell pointer to the data block.

length: VAR { input-output } of integer

Specifies the length of the data block in words.

key: integer

Specifies a cryptographic key of type integer. The value specified must be the same key used to encrypt the file. Each byte of the integer must consist of seven key bits and one bit of odd parity. For more information on the integer format of a cryptographic key, see the section in this appendix called Integer Formats for Cryptographic Keys.

mode: edt$des_encryption_mode

Specifies the DES encryption mode that was used to encrypt the file. Any of the following strings may be specified:

'ELECTRONIC_CODE_BOOK'
'ECB'
'CIPHER_BLOCK_CHAINING'
'CBC'
'CIPHER_FEEDBACK'
'CFB'
'OUTPUT_FEEDBACK'
'OFB'

feedback_unit_length: integer

Specifies the number of bits fed back into the DES algorithm when using CIPHER_FEEDBACK or OUTPUT_FEEDBACK modes. The value specified can be an integer in the set: (0, 1, 7, 8, 56, 64, where 0=64)

initial_vector: VAR { input-output } of integer

Specifies the seed value that was used to encrypt the file. This parameter is required by all modes except ELECTRONIC_CODE_BOOK mode. The value is modified by the decryption algorithm and then returned. The returned value may be used on subsequent calls to this procedure.

status: VAR { output } of ost$status

Returns the request status.

Index

U

Comments (continued from other side)

Please fold on dotted line;
seal edges with tape only.

FOLD

FOLD

FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
First-Class Mail Permit No. 8241 Minneapolis, MN

POSTAGE WILL BE PAID BY ADDRESSEE

CONTROL DATA
Technical Publications
ARH219
4201 N. Lexington Avenue
Arden Hills, MN 55126-9983

NOS/VE Security Administration 60463945 A

We would like your comments on this manual to help us improve it. Please take a few minutes to fill out this form.

Who are you?

- ☐ Manager
- ☐ Systems analyst or programmer
- ☐ Applications programmer
- ☐ Operator
- ☐ Other _____

How do you use this manual?

- ☐ As an overview
- ☐ To learn the product or system
- ☐ For comprehensive reference
- ☐ For quick look-up
- ☐ Other _____

What programming languages do you use? _____

_____

How do you like this manual? Answer the questions that apply.

Yes	Somewhat	No	
☐	☐	☐	Does it tell you what you need to know about the topic?
☐	☐	☐	Is the technical information accurate?
☐	☐	☐	Is it easy to understand?
☐	☐	☐	Is the order of topics logical?
☐	☐	☐	Can you easily find what you want?
☐	☐	☐	Are there enough examples?
☐	☐	☐	Are the examples helpful? (☐ Too simple? ☐ Too complex?)
☐	☐	☐	Do the illustrations help you?
☐	☐	☐	Is the manual easy to read (print size, page layout, and so on)?
☐	☐	☐	Do you use this manual frequently?

Comments? If applicable, note page and paragraph. Use other side if needed.

Check here if you want a reply: ☐

Name _____ Company _____

Address _____ Date _____

_____ Phone _____

Please send program listing and output if applicable to your comment.


CONTROL DATA