Safeguard Administrator’s Manual

Abstract

This manual describes Safeguard commands and features reserved for security administrators and privileged users.

Product Version

Safeguard G06.06, H05

Supported Release Version Updates (RVUs)

This publication supports J06.03 and all subsequent J-series RVUs, H06.08 and all subsequent H-series RVUs, and G06.29 and all subsequent G-series RVUs, until otherwise indicated by its replacement publications. Additionally, all considerations for H-series throughout this manual will hold true for J-series also, unless mentioned otherwise.

Part Number	Published
523317-029	August 2013

Document History

Part Number	Product Version	Published
523317-023	Safeguard G06.06, H05	February 2011
523317-026	Safeguard G06.06, H05	February 2012
523317-027	Safeguard G06.06, H05	August 2012
523317-028	Safeguard G06.06, H05	February 2013
523317-029	Safeguard G06.06, H05	August 2013

Legal Notices

Printed in the US

Safeguard Administrator's Manual

Index | Figures | Tables

Legal Notices

What's New in This Manual vii
- Manual Information vii
- New and Changed Information vii

About This Manual xi
- Notation Conventions xii

1. Introduction

Who Can Use the Safeguard Subsystem? 1-1
The Importance of a Security Policy 1-2
Preliminary Security Planning 1-3
- The Corporate Security Officer and Security Policy 1-3
- The Security Administrator 1-3
- Objects That Require Protection 1-3
- Who Can Run SAFECOM? 1-4
- Analyzing Security Needs 1-4

2. Controlling User Access

Introduction 2-1
- USER Commands 2-1
- TERMINAL Commands 2-2
- ALTER SAFEGUARD Command 2-2

Using SAFECOM to Establish a Local User Community 2-4
- Defining Administrative Groups 2-5
- Adding Users to the System 2-6

Using SAFECOM to Manage User Access to Your System 2-16
- Changing the Owner of a User Authentication Record 2-16
- Granting a User Temporary Access to Your System 2-17
- Requiring Users to Change Their Passwords 2-20
- Granting a Grace Period for Changing an Expired Password 2-23
- Forcing Immediate Expiration of a User Password 2-23

Freezing a User's Ability to Access the System 2-24
Specifying Auditing for a User ID 2-25
Deleting Users 2-26
Deleting Administrative Groups 2-26
Using SAFECOM to Establish a Network of Users 2-27
Using Safeguard With Nodes With Standard Security 2-27
Identifying Network Users 2-28
Granting a Network User Access to Objects on Your System 2-29
Establishing a Community of Network Users 2-30
Changes to the PAID During a User Session 2-33
Additional Considerations for Aliases and Groups 2-33
Additional Considerations for ACCESS with Network Specific Subject IDs 2-33
Establishing Default Protection for a User's Disk Files 2-34
Establishing a Default Access Control List 2-35
Establishing Default Ownership 2-36
Specifying Default Audit Attributes 2-36
Eliminating Default Protection for a User 2-37
Specifying a Default Command Interpreter for a User 2-37
Establishing Guardian Defaults 2-38
Setting the File-Security String 2-38
Specifying the Default Volume and Subvolume 2-39
Assigning an Alias to a User 2-40

3. Managing User Groups

Adding User Groups 3-2
Adding and Deleting Group Members 3-3
Using Wild-cards for Managing Group Members 3-4
Transferring Group Ownership 3-5
Deleting Groups 3-6

4. Securing Volumes and Devices

General Procedure for Securing Volumes and Devices 4-2
Considerations for Volumes 4-3
Considerations for Devices and Subdevices 4-4

5. OBJECTTYPE Control

Controlling an Entire Object Type 5-6
Controlling Users as an Object Type 5-7
Controlling Who Can Add an Object Type 5-8
OBJECTTYPE Auditing 5-9

6. Managing Security Groups

Adding Security Groups 6-3
Transferring Security Group Ownership 6-6
Freezing and Thawing Security Groups 6-8
Deleting Security Groups and Group Members 6-10

7. Securing Terminals

Control of the Logon Dialog 7-2
Starting a Command Interpreter 7-2
Adding a Terminal Definition 7-3
Altering a Terminal Definition 7-4
Freezing and Thawing a Terminal 7-5
Deleting a Terminal Definition 7-5

8. Warning Mode

Considerations for Disk Files and Processes 8-2
- Disk-File Security 8-2
- Process Stop Mode Security 8-3

Using Warning Mode 8-4

9. Configuration

Safeguard Attributes 9-1
Configuring User Authentication 9-5
Configuring Password Control 9-6
Configuring Device Control 9-16
Configuring Process Control 9-17
Configuring Disk-File Control 9-18
Configuring Safeguard Auditing 9-21
- Configuring User Authentication Auditing 9-21
- Configuring Device Auditing 9-22
- Configuring Process Auditing 9-23
- Configuring Disk File Auditing 9-24
- Configuring Auditing of All System Objects 9-25
- Configuring Client Auditing 9-26
- Configuring Audit Exclusion of NonStop Client Events 9-27

Configuring a Default Command Interpreter 9-33
Configuring Communication With $CMON 9-34
Configuring Logon Dialog 9-35
Configuring Exclusive Access at Safeguard Terminals 9-36
Configuring Warning Mode 9-36

Configuring Persistence 9-37
Configuring Attributes for Node Specific Subjects in ACLs 9-37
Configuring Dynamic Process Updates 9-37

10. Installation and Management

Safeguard Components 10-1
The Security Manager Process (SMP) 10-1
The Security Monitors (SMONs) 10-2
Safeguard Helper Process (SHP) 10-2
Process Considerations for the SMP and SAFECOM 10-3
Swap Space Migration Considerations 10-3
Safeguard Subsystem Management Commands 10-4
General Installation Procedure 10-4
Installing the Safeguard Software 10-5
Adding the Safeguard Software to the Kernel Subsystem (G-Series RVUs) 10-5
Including the Safeguard Software in the OSIMAGE File (D-Series RVUs) 10-7
Starting the SMP 10-7
Converting to the Safeguard Subsystem 10-9
Updating the Safeguard Software 10-10
Updating a Previous RVU With the Safeguard Software Running 10-10
Updating a Previous RVU With the Safeguard Software Stopped 10-10
Returning to a Previous RVU 10-11
Guidelines for Securing the Safeguard Subsystem 10-12
Monitoring the Safeguard Subsystem 10-14
Safeguard Console Messages 10-14
Managing Safeguard Audit Files 10-14

A. SAFECOM Command Syntax

Common Syntax Elements A-1
SAFECOM Command Syntax A-3

Index

Figures

Figure 2-1. A Community of Network Users 2-32

Tables

Table 2-1. User Security Attributes and Default Attribute Values 2-7
Table 2-2. User Security Commands 2-12
Table 2-3. SAFECOM and Standard Security Programs Used to Manage Network Users 2-28

Table 3-1. Group Command Summary 3-1
Table 4-1. Security Commands for Volumes and Devices 4-1
Table 4-2. Types of Objects and Valid Access Authorities 4-2
Table 5-1. OBJECTTYPE Security Commands 5-5
Table 6-1. Security Groups and Restricted Commands 6-2
Table 6-2. SECURITY-GROUP Command Summary 6-3
Table 7-1. Security Groups and TERMINAL Commands 7-1
Table 7-2. TERMINAL Command Summary 7-2
Table 8-1. Warning Mode Rulings on Object ACLs 8-1
Table 8-2. Warning Mode Rulings on Disk-File ACLs 8-3
Table 8-3. Warning Mode Rulings on Process ACLs 8-4
Table 9-1. Safeguard Attribute Default Values 9-2
Table 9-2. AUDIT-EXCLUDE-FIELDs and their corresponding values 9-28
Table 10-1. Safeguard Subsystem Management Commands 10-4

What's New in This Manual

Manual Information

Abstract

This manual describes Safeguard commands and features reserved for security administrators and privileged users.

Product Version

Safeguard G06.06, H05

Supported Release Version Updates (RVUs)

This publication supports J06.03 and all subsequent J-series RVUs, H06.08 and all subsequent H-series RVUs, and G06.29 and all subsequent G-series RVUs, until otherwise indicated by its replacement publications. Additionally, all considerations for H-series throughout this manual will hold true for J-series also, unless mentioned otherwise.

Part Number	Published
523317-029	August 2013

Document History

Part Number	Product Version	Published
523317-023	Safeguard G06.06, H05	February 2011
523317-026	Safeguard G06.06, H05	February 2012
523317-027	Safeguard G06.06, H05	August 2012
523317-028	Safeguard G06.06, H05	February 2013
523317-029	Safeguard G06.06, H05	August 2013

New and Changed Information

Changes to 523317-029 manual

- Updated the section ALTER SAFEGUARD Command on page 2-4.
- Updated the section Managing Security Groups on page 6-1.
- Updated the section Adding Security Groups on page 6-4.
- Updated the section Transferring Security Group Ownership on page 6-7.
- Updated the section Freezing and Thawing Security Groups on page 6-10.
- Updated the section Deleting Security Groups and Group Members on page 6-13.

- Updated the table Safeguard Attribute Default Values on page 9-2.

Changes to 523317-028 manual

- Updated the chapter Managing Security Groups on page 6-1.
- Added the security group SECURITY-MEDIA-ADMIN in the section Adding Security Groups on page 6-4.
- Added an example for security group SECURITY-MEDIA-ADMIN in the following sections:
  - Transferring Security Group Ownership on page 6-7.
  - Freezing and Thawing Security Groups on page 6-10.
  - Deleting Security Groups and Group Members on page 6-13.
- Added a note on OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME and OBJECTTYPE USER in the following section/pages:
  - Chapter OBJECTTYPE Control on page 5-1.
  - Section Controlling an Entire Object Type on page 5-6.
- Updated a note in the section Configuring Persistence on page 9-37.

Changes to 523317-027 manual

- Updated ALTER SAFEGUARD command with PASSWORD-ERROR-DETAIL in the user access and authentication controls on page 2-4.
- Added PASSWORD-ERROR DETAIL attribute to INFO SAFEGUARD command on page 9-14.
- Added OWNER-LIST group attribute to INFO GROUP and ADD GROUP commands on page A-4.
- Added COMPARE option to INFO SAFEGUARD command syntax on page A-10.

Changes to 523317-026 manual

- Added Note and updated the examples under Controlling Users as an Object Type on page 5-8.
- Added SECURITY-AUDITOR security group under Managing Security Groups on page 6-1.
- Added a Note under PASSWORD-MIN-QUALITY-REQUIRED on page 9-9 .

Changes to 523317-025 manual

- Updated the note on page 10-5.

- Added a new example on page 10-12.

Changes to the H06.22/J06.11 manual

- Updated the Safeguard product version on page -1.
- Added the following password attributes and their descriptions:
  - PASSWORD-MIN-UPPERCASE-REQ on pages 2-3 and 9-10.
  - PASSWORD-MIN-LOWERCASE-REQ on pages 2-3 and 9-11.
  - PASSWORD-MIN-NUMERIC-REQ on pages 2-4 and 9-12.
  - PASSWORD-MIN-SPECIALCHAR-REQ on pages 2-3 and 9-12.
  - PASSWORD-ALPHA-REQUIRED on pages 2-3 and 9-13.
  - PASSWORD-MIN-ALPHA-REQ on pages 2-3 and 9-13.
- Added information about the SECURITY-PRV-ADMINISTRATOR group in the following sections:
  - Adding Security Groups on page 6-5.
  - Transferring Security Group Ownership on page 6-8.
  - Freezing and Thawing Security Groups on page 6-11.
  - Deleting Security Groups and Group Members on page 6-14.
- Added a note specifying that the following password attributes support the DES and HMAC256 password algorithms:
  - PASSWORD-UPPERCASE-REQ on page 9-8.
  - PASSWORD-LOWERCASE-REQ on page 9-8.
  - PASSWORD-NUMERIC-REQ on page 9-8.
  - PASSWORD-SPECIALCHAR-REQ on page 9-9.
- Updated the description of the PASSWORD-MIN-QUALITY-REQUIRED attribute on page 9-9.

Changes to the H06.21/J06.10 Manual

- Updated the list of users who can set the PROGID attribute to protect the program code, on page 2-36.
- Added the AUDIT-TACL-LOGOFF attribute in Table 9-1, Safeguard Attribute Default Values, on page 9-5.
- Added the DYNAMIC-PROC-UPDATE attribute in Table 9-1, Safeguard Attribute Default Values, on page 9-5 and its description on page 9-27.

- Updated the description of PASSWORD-UPPERCASE-REQUIRED {ON / OFF} on page 9-7.
- Updated the description of PASSWORD-LOWERCASE-REQUIRED {ON / OFF} on page 9-8.
- Updated the description of PASSWORD-NUMERIC-REQUIRED {ON / OFF} on page 9-8.
- Updated the description of PASSWORD-SPECIALCHAR-REQUIRED {ON / OFF} on page 9-9.
- Added the Configuring Dynamic Process Updates section on page 9-37.
- Added a section on the Safeguard Helper process on page 10-2.
- Added a note mentioning that the $SYSTEM.SAFE.SPTGUARD file is created when the SAVED-DISKFILE-PATTERN protection records are created, on page 10-9.

Changes to the H06.20/J06.09 Manual

- Updated the description of PASSWORD-MIN-QUALITY-REQUIRED on page 9-9.

Changes to the 523317-020 Manual

- Updated notes in the following sections to include support for G-series RVUs:
  - User Security Attributes and Default Attribute Value Table on page 2-7.
  - OBJECT-TEXT-DESCRIPTION attribute on page 5-4.
  - INFO SECURITY-GROUP command display on page 6-5.
  - Safeguard Attribute Default Values Table on page 9-2.
  - AUDIT-TACL-LOGOFF attribute on page 9-27.
  - AUDIT-PRIV-LOGON attribute on page A-15.
- Added a note to the Using Wild-cards for Managing Group Members section to include support for J-series and H-series RVUs on page 3-4.
- Updated the Process Considerations for the SMP and SAFECOM section on page 10-3.

Changes to the H06.19/J06.08 Manual

- Updated the description of AUDIT-USER-ACTION-PASS attribute in Table 2-1, User Security Attributes and Default Attribute Values, on page 2-8.
- Updated the description of AUDIT-USER-ACTION-FAIL attribute in Table 2-1, User Security Attributes and Default Attribute Values, on page 2-9.

- Added the following tokens to the display of INFO USER command:
  - CREATION-TIME on pages 2-14, 2-15, 2-19, 2-20, 2-21, 2-22, 2-23, 2-24, 2-39, 2-40, 2-42, 3-2, 3-3, 3-3, and 3-4.
  - CREATOR-USER-NAME on pages 2-14, 2-15, 2-19, 2-42, 3-2, 3-3, 3-3, and 3-4.
  - CREATOR-USER-TYPE on pages 2-14, 2-15, 2-19, 2-42, 3-2, 3-3, 3-3, and 3-4.
  - CREATOR-NODENUMBER on pages 2-14, 2-15, 2-19, 2-42, 3-2, 3-3, 3-3, and 3-4.
- Updated the syntax of ALTER GROUP command on page 3-3.
- Added Using Wild-cards for Managing Group Members on page 3-4.
- Updated the display of INFO SAFEGUARD command on page 8-5.
- Changed the default value of PASSWORD-ENCRYPT in the footnote of Table 9-1, Safeguard Attribute Default Values, on page 9-2.
- Added AUDIT-OSS-FILTER attribute in Table 9-1, Safeguard Attribute Default Values, on page 9-5 and its description on page 9-26.
- Updated the description of PASSWORD-ENCRYPT on page 9-14.
- Added PASSWORD-ALGORITHM on page 9-15.
- Updated the note on CLEARONPURGE-DISKFILE on page 9-19.
- Added MID option on page 9-20.
- Added AUDIT-TACL-LOGOFF attribute on page 9-26.
- Updated the Process Considerations for the SMP and SAFECOM section on page 10-3.

About This Manual

This manual describes features of the Safeguard software that are reserved for security administrators and privileged users. The first section of this manual introduces the Safeguard software and presents general guidelines and recommendations for establishing system security.

The remainder of the manual covers:

- Controlling user access to the system; setting up local and network communities to use SAFECOM (that is, providing Safeguard protection for users)
- Assigning aliases to users
- Establishing file-sharing groups of users
- Securing disk volumes and nondisk device
- Using OBJECTTYPE authorization to designate who has control of specific types of system resources
- Defining terminals so that the Safeguard software can start specific command interpreters at those terminals
- Using warning mode to test the effectiveness of your access control lists (ACLs)
- Configuring the Safeguard software to match the needs of your security policy
- Managing the Safeguard subsystem—including instructions for installing and monitoring the Safeguard software

Appendix A, SAFECOM Command Syntax, summarizes the syntax of all SAFECOM commands, including those available to the general user.

This manual uses examples to introduce the SAFECOM commands and illustrate typical usage. For more information about the syntax and all the SAFECOM commands, see the *Safeguard Reference Manual*.

The reader of this manual is assumed to know how to run SAFECOM, the Safeguard command interpreter, and how to use the Safeguard software to secure disk files, subvolumes, and processes. These topics are covered in the *Safeguard User's Guide*.

In addition, before reading this manual, you should be familiar with the *Introduction to Tandem NonStop Systems* (D-series RVUs), *NonStop S-Series Servers Introduction* (G-series RVUs), and the *Guardian User's Guide.*

You might find the following reference manuals helpful:

- *Safeguard Reference Manual*
- *Security Management Guide*
- *Safeguard Management Programming Manual*

Notation Conventions

Hypertext Links

Blue underline is used to indicate a hypertext link within text. By clicking a passage of text with a blue underline, you are taken to the location described. For example:

> This requirement is described under Backup DAM Volumes and Physical Disk Drives on page 3-2.

General Syntax Notation

The following list summarizes the notation conventions for syntax presentation in this manual.

UPPERCASE LETTERS. Uppercase letters indicate keywords and reserved words; enter these items exactly as shown. Items not enclosed in brackets are required. For example:

```
MAXATTACH
```

lowercase italic letters. Lowercase italic letters indicate variable items that you supply. Items not enclosed in brackets are required. For example:

file-name

computer type. `Computer type` letters within text indicate C and Open System Services (OSS) keywords and reserved words; enter these items exactly as shown. Items not enclosed in brackets are required. For example:

```
myfile.c
```

italic computer type. *`Italic computer type`* letters within text indicate C and Open System Services (OSS) variable items that you supply. Items not enclosed in brackets are required. For example:

pathname

[ ] Brackets. Brackets enclose optional syntax items. For example:

```
TERM [\system-name.]$terminal-name
```

```
INT[ERRUPTS]
```

A group of items enclosed in brackets is a list from which you can choose one item or none. The items in the list can be arranged either vertically, with aligned brackets on

each side of the list, or horizontally, enclosed in a pair of brackets and separated by vertical lines. For example:

```
FC [ num  ]
   [ -num ]
   [ text ]

K [ X | D ] address
```

{ } Braces. A group of items enclosed in braces is a list from which you are required to choose one item. The items in the list can be arranged either vertically, with aligned braces on each side of the list, or horizontally, enclosed in a pair of braces and separated by vertical lines. For example:

```
LISTOPENS PROCESS { $appl-mgr-name }
                  { $process-name  }

ALLOWSU { ON | OFF }
```

| Vertical Line. A vertical line separates alternatives in a horizontal list that is enclosed in brackets or braces. For example:

```
INSPECT { OFF | ON | SAVEABEND }
```

… Ellipsis. An ellipsis immediately following a pair of brackets or braces indicates that you can repeat the enclosed sequence of syntax items any number of times. For example:

```
M address [ , new-value ]...

[ - ] {0|1|2|3|4|5|6|7|8|9}...
```

An ellipsis immediately following a single syntax item indicates that you can repeat that syntax item any number of times. For example:

```
"s-char..."
```

Punctuation. Parentheses, commas, semicolons, and other symbols not previously described must be entered as shown. For example:

```
error := NEXTFILENAME ( file-name ) ;

LISTOPENS SU $process-name.#su-name
```

Quotation marks around a symbol such as a bracket or brace indicate the symbol is a required character that you must enter as shown. For example:

```
"[" repetition-constant-list "]"
```

Item Spacing. Spaces shown between items are required unless one of the items is a punctuation symbol such as a parenthesis or a comma. For example:

```
CALL STEPMOM ( process-id ) ;
```

If there is no space between two items, spaces are not permitted. In the following example, there are no spaces permitted between the period and any other items:

```
$process-name.#su-name
```

Line Spacing. If the syntax of a command is too long to fit on a single line, each continuation line is indented three spaces and is separated from the preceding line by a blank line. This spacing distinguishes items in a continuation line from items in a vertical list of selections. For example:

```
ALTER [ / OUT file-spec / ] LINE

   [ , attribute-spec ]...
```

!i and !o. In procedure calls, the !i notation follows an input parameter (one that passes data to the called procedure); the !o notation follows an output parameter (one that returns data to the calling program). For example:

```
CALL CHECKRESIZESEGMENT (  segment-id                    !i
                         , error        ) ;              !o
```

!i,o. In procedure calls, the !i,o notation follows an input/output parameter (one that both passes data to the called procedure and returns data to the calling program). For example:

```
error := COMPRESSEDIT ( filenum ) ;                      !i,o
```

!i:i. In procedure calls, the !i:i notation follows an input string parameter that has a corresponding parameter specifying the length of the string in bytes. For example:

```
error := FILENAME_COMPARE_ (  filename1:length           !i:i
                            , filename2:length ) ;       !i:i
```

!o:i. In procedure calls, the !o:i notation follows an output buffer parameter that has a corresponding input parameter specifying the maximum length of the output buffer in bytes. For example:

```
error := FILE_GETINFO_ (  filenum                        !i
                        , [ filename:maxlen ] ) ;        !o:i
```

Notation for Messages

The following list summarizes the notation conventions for the presentation of displayed messages in this manual.

Bold Text. Bold text in an example indicates user input entered at the terminal. For example:

```
ENTER RUN CODE

?123

CODE RECEIVED:      123.00
```

The user must press the Return key after typing the input.

Nonitalic text. Nonitalic letters, numbers, and punctuation indicate text that is displayed or returned exactly as shown. For example:

```
Backup Up.
```

lowercase italic letters. Lowercase italic letters indicate variable items whose values are displayed or returned. For example:

```
p-register
```

```
process-name
```

[ ] Brackets. Brackets enclose items that are sometimes, but not always, displayed. For example:

```
Event number = number [ Subject = first-subject-value ]
```

A group of items enclosed in brackets is a list of all possible items that can be displayed, of which one or none might actually be displayed. The items in the list might be arranged either vertically, with aligned brackets on each side of the list, or horizontally, enclosed in a pair of brackets and separated by vertical lines. For example:

```
proc-name trapped [ in SQL | in SQL file system ]
```

{ } Braces. A group of items enclosed in braces is a list of all possible items that can be displayed, of which one is actually displayed. The items in the list might be arranged either vertically, with aligned braces on each side of the list, or horizontally, enclosed in a pair of braces and separated by vertical lines. For example:

```
obj-type obj-name state changed to state, caused by
{ Object | Operator | Service }
```

```
process-name State changed from old-objstate to objstate
{ Operator Request. }
{ Unknown.          }
```

| Vertical Line. A vertical line separates alternatives in a horizontal list that is enclosed in brackets or braces. For example:

```
Transfer status: { OK | Failed }
```

% Percent Sign. A percent sign precedes a number that is not in decimal notation. The % notation precedes an octal number. The %B notation precedes a binary number. The %H notation precedes a hexadecimal number. For example:

```
%005400
```

```
%B101111
```

```
%H2F
```

```
P=%p-register E=%e-register
```

Notation for Management Programming Interfaces

The following list summarizes the notation conventions used in the boxed descriptions of programmatic commands, event messages, and error lists in this manual.

UPPERCASE LETTERS. Uppercase letters indicate names from definition files; enter these names exactly as shown. For example:

```
ZCOM-TKN-SUBJ-SERV
```

lowercase letters. Words in lowercase letters are words that are part of the notation, including Data Definition Language (DDL) keywords. For example:

```
token-type
```

!r. The !r notation following a token or field name indicates that the token or field is required. For example:

```
ZCOM-TKN-OBJNAME       token-type ZSPI-TYP-STRING.          !r
```

!o. The !o notation following a token or field name indicates that the token or field is optional. For example:

```
ZSPI-TKN-MANAGER       token-type ZSPI-TYP-FNAME32.         !o
```

Change Bar Notation

Change bars are used to indicate substantive differences between this edition of the manual and the preceding edition. Change bars are vertical rules placed in the right margin of changed portions of text, figures, tables, examples, and so on. Change bars highlight new or revised information. For example:

> The message types specified in the REPORT clause are different in the COBOL85 environment and the Common Run-Time Environment (CRE).
>
> The CRE has many new message types and some new message type codes for old message types. In the CRE, the message type SYSTEM includes all messages except LOGICAL-CLOSE and LOGICAL-OPEN.

HP Encourages Your Comments

HP encourages your comments concerning this document. We are committed to providing documentation that meets your needs. Send any errors found, suggestions for improvement, or compliments to docsfeedback@hp.com.

Include the document title, part number, and any comment, error found, or suggestion for improvement you have concerning this document.

1 Introduction

As a security administrator or privileged user, you have access to Safeguard features that are not usually available to general users. This manual describes those features and the additional responsibilities you have as a member of the system security team. Those duties and responsibilities include:

- Installing, configuring, and managing the Safeguard subsystem
- Adding users to the Safeguard database, managing their user authentication records, and assigning aliases to users
- Establishing groups of users for file-sharing purposes
- Securing disk volumes and nondisk devices
- Controlling who can create authorization records for objects of a given type
- Establishing security groups of users who can execute restricted commands
- Adding terminal definitions so that the Safeguard software can provide exclusive access and automatic starting of a specific command interpreter at the terminal
- Using warning mode to test the effectiveness of your security policy

In addition to these specific duties, you are probably involved in formulating an overall security policy for your installation and in planning the most appropriate ways to use the Safeguard software.

Note. In earlier product versions, extended features for logon dialog, such as warning of a pending password expiration, were available only at a Safeguard terminal. Effective with the D30 product version, the TACL command interpreter also provides these logon features when the Safeguard software is running on the system.

Who Can Use the Safeguard Subsystem?

To use the Safeguard command interpreter, an individual must have EXECUTE authority for the SAFECOM program. As a security administrator, you can limit this authority to certain users by creating an access control list for the SAFECOM program file.

Initially, SAFECOM limits what certain classes of users can do. For example:

- By default, general users can add their own disk files, subvolumes, processes, and subprocesses to the Safeguard database. For more information on functions, see the *Safeguard User's Guide*.
- By default, only local super-group members (user ID 255,*n*) can add volumes, devices, and subdevices to the Safeguard database.
- By default, the group manager (user ID *n*,255) can add and delete users, thereby controlling all the user authentication records in the group.

- By default, the local super ID (user ID 255,255) can execute any SAFECOM command for any user or object.
- By default, only local super-group members (user ID 255,*n*) can initially add users to file-sharing groups and special security groups, execute audit service commands, add terminal definition records, and control the Safeguard configuration.

You can limit or change these authorities to suit your company's security policy. One way to do this is to specify OBJECTTYPE authorization, as explained in Section 5, OBJECTTYPE Control. For example, you might want to put only two people in charge of securing all disk volumes. To perform this task, a local super-group member could give these users exclusive control of all volumes by creating an OBJECTTYPE VOLUME protection record and giving these create authority on that access control list.

Another way to restrict the use of certain commands is to define security groups, as explained in Section 6, Managing Security Groups.

Even the use of the super ID can be controlled. Ownership of the authentication record for the super ID can be given to the security administrator, who can then suspend the super ID with the FREEZE command and restore it, when needed, with the THAW command.

The Importance of a Security Policy

Before you use the Safeguard software, it is important that your installation has an established security policy that is supported by management. This policy should address all aspects of security, including physical security (such as control of tape libraries) and security of the HP NonStop™ system itself (through software controls).

The security policy must be understood by everyone involved. You should have a user education program aimed at both new and experienced employees. This program should also keep everyone informed of any changes to the policy.

The next subsection in this manual summarizes the key issues to consider in a formulating a security policy. The *Security Management Guide* provides more detailed information to help you develop a comprehensive security policy. The Safeguard subsystem provides software tools to help you implement the policy.

Preliminary Security Planning

Advance planning is required before you install the Safeguard software. To plan the security for your installation, you must understand the applications used on your system, and you must know which users should be allowed to use system resources.

The Corporate Security Officer and Security Policy

Effective security requires that you have a security officer with executive status and with authority to establish and enforce security policy. In turn, the security officer must also be responsible for handling breaches of security.

Before installing the Safeguard software, the security officer establishes a security policy for the system or network. The protection mechanisms offered by the Safeguard software can help you implement many different security schemes, such as:

1. Restrictive security, in which a security administrator controls access to system resources, and most users can access only a few resources
2. Permissive security, in which many users control access to different system resources, and most resources are available to all users
3. Centralized security for a network, in which the security administrator is a network user who either owns most of the network resources or serves as the network group manager for local security administrators
4. Decentralized security for systems in a network, in which local users restrict access to the resources on their node

The Security Administrator

Each installation must have a designated security administrator to set up and maintain security control. HP recommends that the individual with the super ID not be assigned the role of security administrator. Instead, a user familiar with computer operations or a user from a security-related group such as the auditing department might be a better choice.

A single part-time security administrator might be adequate for a centralized security scheme that controls only a few system objects. A more comprehensive centralized security scheme might require one or several full-time security administrators. For a decentralized security scheme, in which each node controls the security of at least some local objects, a local security administrator is probably needed at each node.

Objects That Require Protection

A security administrator should set only needed restrictions for all system objects that require protection. To do this, consider the applications that run in the system and the security requirements for each. Access to disk files, devices, subdevices, volumes, subvolumes, processes, and subprocesses must be provided. You must determine exactly which users need to have access to each object or type of object.

Who Can Run SAFECOM?

Decide who will use SAFECOM. Once a user runs SAFECOM, only the Safeguard internal restrictions limit the user's capabilities. The Safeguard software imposes internal restrictions on commands such as ADD USER, ALTER USER, and ADD DEVICE, and most SAFECOM operations on existing protection records are restricted to the record owner, the owner's group manager, and the local super ID. However many SAFECOM commands are unrestricted.

If you want to limit the number of users who can use SAFECOM, you can use either the standard security system or the Safeguard subsystem to secure the SAFECOM program object file. Section 10, Installation and Management, describes how to use the Safeguard subsystem to secure the SAFECOM program object file and the Safeguard Subsystem Programmatic Interface (SPI).

Analyzing Security Needs

When you first start the SMP (Security Manager Process), all users listed in the USERID file are automatically put under Safeguard control. The Safeguard software offers control over additional user attributes, such as the password change requirement and user expiration date. Decide in advance which users require additional controls and determine what those controls will be. Section 2, Controlling User Access, describes these additional user controls.

2 Controlling User Access

This section describes how to use the SAFECOM user security commands to establish a local user community and to manage user access to a system protected by the Safeguard software. It also describes how to identify network users, how to set up network access for users, and how to establish default protection for users' disk files.

Introduction

User security controls are established with USER security commands when you add or alter a user authentication record. They are supplemented by other user access controls offered by the TERMINAL security commands and by the ALTER SAFEGUARD command.

USER Commands

With USER security commands, you can specify the following user access and authentication controls:

- User expiration date, to terminate a user's ability to log on (USER-EXPIRES attribute)
- Future password expiration date for a user (PASSWORD-MUST-CHANGE attribute)
- Immediate or future password expiration for a user (PASSWORD-EXPIRES attribute)
- Grace period during which an expired password can be changed at a terminal controlled by the Safeguard software (PASSWORD-EXPIRY-GRACE attribute)
- Temporary suspension of a user's ability to log on (FREEZE and THAW)
- Default protection for a user's disk files (DEFAULT-PROTECTION attribute)
- Initial password for a user (PASSWORD attribute)
- Remote password for a user (REMOTEPASSWORD attribute)
- Guardian default security string (GUARDIAN DEFAULT SECURITY attribute)
- Guardian default volume and subvolume (GUARDIAN DEFAULT VOLUME attribute)
- Auditing of a user's logon attempts (AUDIT-AUTHENTICATE attributes)
- Auditing of attempts to change a user authentication record (AUDIT-MANAGE attributes)
- Auditing of a user's attempts to access objects and manage protection records (AUDIT-USER-ACTION attributes)
- Primary group for a user (PRIMARY-GROUP attribute)

- Initial directory, initial program, and initial program type for the user in an HP NonStop Open Systems Services (OSS) environment (INITIAL-DIRECTORY, INITIAL-PROGRAM, and INITIAL-PROGTYPE attributes)
- Automatic starting of a command interpreter for a user after logon at a terminal controlled by the Safeguard software (CI-PROG attribute)

TERMINAL Commands

The TERMINAL commands, which are described in Section 7, Securing Terminals, allow you to add terminal definitions to the Safeguard database. The TERMINAL commands control this aspect of user authentication:

- Automatic starting of a command interpreter for a user after logon (PROG attribute)

ALTER SAFEGUARD Command

You use the ALTER SAFEGUARD command, described in Section 9, Configuration, to set the Safeguard configuration attributes. Many configuration attributes provide security controls for users on a systemwide basis. For example, you can set the minimum password length or limit the number of failed logon attempts. Other configuration attributes become the default value for attributes not specified in a particular user authentication record. For example, you can configure the Safeguard software so that all attempts to log on are audited even if auditing is not specified in individual user authentication records.

With the Safeguard configuration attributes, you specify these user access and authentication controls:

- Password algorithm (PASSWORD-ALGORITHM) (only on systems running G06.29 and later G-series RVUs and H06.06 and later H-series RVUs)
- Minimum password length for all users (PASSWORD-MINIMUM-LENGTH)
- Maximum password length for all users (PASSWORD-MAXIMUM-LENGTH) (only on systems running G06.31 and later G-series RVUs and H06.08 and later H-series RVUs)
- Password compatibility mode specifies that only first eight characters of the password will be considered during password change. (PASSWORD-COMPATIBILITY-MODE) (only on systems running G06.31 and later G-series RVUs and H06.08 and later H-series RVUs)
- Password history, to prevent reuse of passwords (PASSWORD-HISTORY)
- Password encryption (PASSWORD-ENCRYPT)
- Time period prior to expiration during which a password can be changed (PASSWORD-MAY-CHANGE)
- Password uppercase required, specifies whether a user password will be enforced to have at least one uppercase character

(PASSWORD-UPPERCASE-REQUIRED) (only on systems running G06.31 and later G-series RVUs and H06.09 and later H-series RVUs)

- Password lowercase required, specifies whether a user password will be enforced to have at least one lowercase character (PASSWORD-LOWERCASE-REQUIRED) (only on systems running G06.31 and later G-series RVUs and H06.09 and later H-series RVUs)
- Password numeric required, specifies whether a user password will be enforced to have at least one numeric character (PASSWORD-NUMERIC-REQUIRED) (only on systems running G06.31 and later G-series RVUs and H06.09 and later H-series RVUs)
- Password special character required, specifies whether a user password will be enforced to have at least one special character (non-alphanumeric character except comma, semicolon, and double quote) (PASSWORD-SPECIALCHAR-REQUIRED) (only on systems running G06.31 and later G-series RVUs and H06.09 and later H-series RVUs)
- Password spaces allowed, specifies whether a user password will be allowed to have embedded spaces (PASSWORD-SPACES-ALLOWED) (only on systems running G06.31 and later G-series RVUs and H06.09 and later H-series RVUs)
- Password minimum quality required, specifies the minimum number of quality criteria that have to be met when a password is set or changed (PASSWORD-MIN-QUALITY-REQUIRED) (only on systems running G06.31 and later G-series RVUs and H06.09 and later H-series RVUs)
- Password alphabets required, specifies whether a user password will be enforced to have at least one alphabetic character (PASSWORD-ALPHA-REQUIRED) (only on systems running J06.11 or later J-series RVUs and H06.22 or later H-series RVUs)
- Password minimum alphabets required, indicates the minimum number of alphabetic characters that must be included in a user password (PASSWORD-MIN-ALPHA-REQ) (only on systems running J06.11 or later J-series RVUs and H06.22 or later H-series RVUs)
- Password minimum uppercase characters required, indicates the minimum number of uppercase characters that must be included in a user password (PASSWORD-MIN-UPPERCASE-REQ) (only on systems running J06.11 or later J-series RVUs and H06.22 or later H-series RVUs)
- Password minimum lowercase characters required, indicates the minimum number of lowercase characters that must be included in a user password (PASSWORD-MIN-LOWERCASE-REQ) (only on systems running J06.11 or later J-series RVUs and H06.22 or later H-series RVUs)
- Password minimum special characters required, indicates the minimum number of special characters that must be included in a user password (PASSWORD-MIN-SPECIALCHAR-REQ) (only on systems running J06.11 or later J-series RVUs and H06.22 or later H-series RVUs)

- Password minimum numeric characters required, indicates the minimum number of numeric characters that must be included in a user password (PASSWORD-MIN-NUMERIC-REQ) (only on systems running J06.11 or later J-series RVUs and H06.22 or later H-series RVUs)
- Password error detail, indicates the detailed error message displayed when password quality criteria is not met (PASSWORD-ERROR-DETAIL) (only on systems running J06.14 or later J-series RVUs and H06.25 or later H-series RVUs).
- Maximum number of consecutive failed logon attempts allowed before the Safeguard software freezes the user ID or causes a timeout (AUTHENTICATE-MAXIMUM-ATTEMPTS)
- Timeout period to occur after a user exceeds the maximum number of failed logon attempts (AUTHENTICATE-FAIL-TIMEOUT)
- Freezing the user ID after a user exceeds the maximum number of failed logon attempts (AUTHENTICATE-FAIL-FREEZE)
- Blind (nondisplayable) passwords during logon (BLINDLOGON)
- Mandatory use of user names instead of numeric user IDs during logon (NAMELOGON)
- Mandatory use of password by privileged users (such as the super ID) when they are logging on as another user (PASSWORD-REQUIRED)
- Auditing of all logon attempts (AUDIT-AUTHENTICATE)
- Auditing of all attempts to manage user authentication records (AUDIT-SUBJECT-MANAGE)
- Grace period during which an expired password can be changed (PASSWORD-EXPIRY-GRACE)
- Exclusive terminal access for a user logged on at a Safeguard terminal (TERMINAL-EXCLUSIVE-ACCESS)
- Automatic starting of a command interpreter for a user after logon at a Safeguard terminal (CI-PROG)
- Prompting for user confirmation before stopping Safeguard (PROMPT-BEFORE-STOP).

Note. This attribute is supported only on systems running J06.16 and later J-series RVUs, and H06.27 and later H-series RVUs.

Using SAFECOM to Establish a Local User Community

Before a new user can log on to a system, a group manager or the local super ID must use SAFECOM commands to create a user authentication record in the Safeguard subject database. This user authentication record contains the user ID and user name, password, and other security attributes defined for the user. The Safeguard software uses these security attributes to control access to the system. This subsection describes the user security attributes and the SAFECOM user security commands, and gives examples of adding and deleting users in a system.

Note. When the Safeguard software is installed on a system with an existing user community, it takes over the USERID file as its subject database. When a user logs on, that user's record in the USERID file is expanded to include Safeguard security attributes. You do not have to add existing users individually.

For these users, the Safeguard software retains the existing security attributes that are common to both Safeguard security and the standard Guardian security system. In addition, the Safeguard software assigns values for user security attributes that are unique to Safeguard security (described in Table 2-1 on page 2-7).

Users added through the Safeguard software are recognized by the operating system if the Safeguard subsystem is shut down. However, the extra capabilities that the Safeguard software provides are no longer active.

Defining Administrative Groups

The first step to perform in establishing a local user community is to define group names and group numbers for the administrative groups you will use for managing user authentication records. The second step is to add users to those administrative groups.

Each administrative group has a name and number. An administrative group name is from one to eight alphanumeric characters. The first character must be alphabetic. An administrative group number is a number from 0 through 255.

A particular user’s user name and user ID are derived from the group name and group number of the administrative group to which the user was added with the ADD USER command. This group is known as the user administrative group.

A user can be made a member of other administrative groups with the ADD and ALTER GROUP commands. This form of group membership is used for file-sharing purposes, not administrative purposes. For more information, see Section 3, Managing User Groups.

An administrative group is defined implicitly when the first member of that group is added to the system. By default, only the local super ID can define a new administrative group with the ADD USER command. If your installation has group managers (with member number 255), you might want to add that user as the first group member. The group manager can then add other new members to the group.

(Alternatively, you can use the OBJECTTYPE USER command to define a special set of users who have the authority to add other users to the Safeguard database. For more information, see Section 5, OBJECTTYPE Control.)

To create a new administrative group, the local super ID assigns a unique group name to a previously unused group number. From that point on, the association between the group number and the group name is fixed. Any other users added to the group must be assigned the group name defined for that group number. Examples in this section show how to define administrative groups.

It is also possible to explicitly create a group that can serve as an administrative group before adding users to it. Use the ADD GROUP command, as described in Section 3, Managing User Groups.

Adding Users to the System

When a new user is added to a system with the Safeguard software, whoever adds the user can define several security attributes for the user. The Safeguard subsystem uses these attributes to control user access to the system.

Table 2-1 on page 2-7 describes the user security attributes and gives the predefined default value for each attribute. The predefined attribute values are in effect when you start a SAFECOM process.

Table 2-1. User Security Attributes and Default Attribute Values (page 1 of 5)

Attribute	Description	Default Value
OWNER	Identifies the primary owner of this user authentication record. The primary owner can: ● Change any of the user's security attributes. ● Suspend and restore the user's ability to log on to the system. ● Transfer ownership of the user authentication record to another user. ● Delete the user authentication record.	The default value is the user ID of the person who adds the new user. However, when the Safeguard software is installed on an existing system, the owner of an existing user authentication record is the group manager for that user.
OWNER-LIST	Identifies the secondary owners of this user authentication record. The secondary owners have the same privileges as the primary owners. However, unlike the primary owner, the group managers of the secondary owners do not have access to the user authentication record.	The default value is no secondary owners.
PASSWORD	Is the logon password that the user must enter with the user name to gain access to the system.	The default value is no password. However, when the Safeguard software is installed on an existing system, users retain their current logon passwords.
USER-EXPIRES	Establishes a date on which the ability to log on to the system is suspended.	The default value is no expiration date.
PASSWORD-EXPIRES	Establishes a date on which the user password expires (expiration can be immediate).	The default value is no expiration date.
PASSWORD-MUST-CHANGE EVERY <*num*> DAYS	Specifies the maximum number of days that the user can use the same password.	The default value is no required password change.

[1] The STATIC-FAILED-LOGON-RESET-TIME attribute is supported only on systems running H06.10 and later H-series RVUs and G06.32 and later G-series RVUs.

Table 2-1. User Security Attributes and Default Attribute Values (page 2 of 5)

Attribute	Description	Default Value
PASSWORD-EXPIRY-GRACE	Specifies the number of days after a password expires that the user can change his or her password during logon.	The default value is no grace period.
AUDIT-AUTHENTICATE-PASS	Specifies the conditions under which the Safeguard software creates an audit record of successful attempts to log on with this user name.	The default value is no auditing.
AUDIT-AUTHENTICATE-FAIL	Specifies the conditions under which the Safeguard software creates an audit record of unsuccessful attempts to log on with this user name.	The default value is no auditing.
AUDIT-MANAGE-PASS	Specifies the conditions under which the Safeguard software creates an audit record of successful attempts to change or display this user authentication record.	The default value is no auditing.
AUDIT-MANAGE-FAIL	Specifies the conditions under which the Safeguard software creates an audit record of unsuccessful attempts to change or display this user authentication record.	The default value is no auditing.
AUDIT-USER-ACTION-PASS	Specifies the conditions under which the Safeguard software creates an audit record of successful events performed by this user. When the Safeguard global configuration attributes AUDIT-CLIENT-OSS and AUDIT-OSS-FILTER are enabled, the AUDIT-USER-ACTION-PASS attribute takes effect for OSS auditing.	The default value is no auditing.

[1] The STATIC-FAILED-LOGON-RESET-TIME attribute is supported only on systems running H06.10 and later H-series RVUs and G06.32 and later G-series RVUs.

Table 2-1. User Security Attributes and Default Attribute Values (page 3 of 5)

Attribute	Description	Default Value
AUDIT-USER-ACTION-FAIL	Specifies the conditions under which the Safeguard software creates an audit record of unsuccessful events attempted by this user. When the Safeguard global configuration attributes AUDIT-CLIENT-OSS and AUDIT-OSS-FILTER are enabled, the AUDIT-USER-ACTION-FAIL attribute takes effect for OSS auditing.	The default value is no auditing.
TEXT-DESCRIPTION	Specifies a string of descriptive text to be associated with the user authentication record.	The default value is no descriptive text.
BINARY-DESCRIPTION-LENGTH	Specifies the length in bytes of the binary description to be associated with the user authentication record.	The default value is 0.
REMOTEPASSWORD	Sets a remote password for a node in a network of NonStop systems.	The default value is no remote passwords. However, when the Safeguard software is installed on an existing system, users keep their established remote passwords.
DEFAULT-PROTECTION	Establishes default protection attributes for a user's disk files. The attributes are ACCESS, OWNER, and the auditing attributes. DEFAULT-PROTECTION applies to any new files the user creates.	The default value is no default protection. However, if default protection is specified for some, but not all, of the DEFAULT-PROTECTION attributes, the unspecified attributes have the following default values: an empty access control list; no auditing; OWNER is the user for which DEFAULT-PROTECTION is being specified.

[1] The STATIC-FAILED-LOGON-RESET-TIME attribute is supported only on systems running H06.10 and later H-series RVUs and G06.32 and later G-series RVUs.

Table 2-1. User Security Attributes and Default Attribute Values (page 4 of 5)

Attribute	Description	Default Value
GUARDIAN DEFAULT SECURITY	Sets the Guardian default security string for a user. This security string is given to any of the user disk files that are not added to Safeguard.	The default value is OOOO. However, when the Safeguard software is installed on an existing system, users keep their established Guardian default security.
GUARDIAN DEFAULT VOLUME	Sets the Guardian default volume and subvolume for a user.	The default value is $SYSTEM.NOSUBVOL. However, when the Safeguard software is installed on an existing system, users keep their established default volume and subvolume.
CI-PROG	Specifies the command interpreter to be started automatically after the user logs on at a terminal controlled by the Safeguard software.	The default value is no command interpreter. However, the CI-PROG global configuration attribute is set to $SYSTEM.SYSTEM.TACL. This value is used if no command interpreter is specified for the user or the Safeguard terminal.
CI-LIB	Specifies the library to be used with the command interpreter started after the user logs on at a terminal controlled by the Safeguard software.	The default value is no library file.
CI-NAME	Specifies the process name to be given to the command interpreter started after the user logs on at a terminal controlled by the Safeguard software.	The default value is NONE. The command interpreter is be given a random name of the form $Z*nnn*.
CI-CPU	Specifies the processor in which the command interpreter is to be started.	The default value is any CPU.

[1] The STATIC-FAILED-LOGON-RESET-TIME attribute is supported only on systems running H06.10 and later H-series RVUs and G06.32 and later G-series RVUs.

Table 2-1. User Security Attributes and Default Attribute Values (page 5 of 5)

Attribute	Description	Default Value
CI-SWAP	Specifies the swap volume to be used when the command interpreter is started after the user logs on at a terminal controlled by the Safeguard software.	The default value is null.
CI-PRI	Specifies the priority at which the command interpreter is run when it is started at a terminal controlled by the Safeguard software.	The default value is null. However, the CI-PRI global configuration attribute has a value of 149. This value is used if no priority is specified for the user.
CI-PARAM-TEXT	Specifies the parameter text to be used when the command interpreter is started after the user logs on at a terminal controlled by the Safeguard software.	The default value is no text.
PRIMARY-GROUP	Specifies the primary group for the user.	The default value is the user administrative group.
INITIAL-DIRECTORY	Specifies the pathname for the initial working directory for the user in an OSS environment.	The default is no pathname.
INITIAL-PROGRAM	Specifies the pathname for the initial program for the user in an OSS environment.	The default is no pathname.
INITIAL-PROGTYPE	Specifies the initial program type for the user in an OSS environment.	The default is PROGRAM.
STATIC-FAILED-LOGON-RESET [1]	Specifies the last time when the value of the attribute, STATIC FAILED LOGON COUNT, was reset.	The default is NONE.

[1] The STATIC-FAILED-LOGON-RESET-TIME attribute is supported only on systems running H06.10 and later H-series RVUs and G06.32 and later G-series RVUs.

Table 2-2 lists and describes the SAFECOM user security commands. The commands are listed in the order in which they are normally used, rather than in alphabetic order.

Table 2-2. User Security Commands

Command	Description
SET USER	Establishes default values for the user security attributes. When a user is added to the system, the default values are used for any attributes not specified with ADD USER.
SHOW USER	Displays the current values of the default user security attributes.
ADD USER	Adds a user authentication record to the Safeguard subject database. Assigns the user a unique user name and user ID, and defines security attributes for the new user.
RESET USER	Resets the value of one or more default user security attributes to predefined values.
INFO USER	Displays the current values of the security attributes defined for a user.
ALTER USER	Changes security attribute values for a user.
FREEZE USER	Suspends a user's ability to log on to the system.
THAW USER	Restores a user's ability to log on.
DELETE USER	Deletes a user from the system (by deleting the user authentication record for that user).

To add users to the system, you normally use the SET USER command, the SHOW USER command, the ADD USER command, and the INFO USER command.

For example, consider these series of commands, which assigns the group name ADMIN to group number 1 and adds the user ADMIN.MANAGER to the group.

Use the SET USER command to set default USER attributes:

```
=SET USER PASSWORD-MUST-CHANGE EVERY 30 DAYS
```

Then use the SHOW USER command to check the default settings:

```
=SHOW USER
```

```
TYPE        OWNER    WARNING-MODE
 USER      255,255        OFF

  PASSWORD =
  USER-EXPIRES               =    * NONE *
  PASSWORD-EXPIRES           =    * NONE *
  PASSWORD-MUST-CHANGE EVERY =    30 DAYS
  PASSWORD-EXPIRY-GRACE      =    * NONE *
  GUARDIAN DEFAULT SECURITY  = OOOO
  GUARDIAN DEFAULT VOLUME    = $SYSTEM.NOSUBVOL

  AUDIT-AUTHENTICATE-PASS  = NONE         AUDIT-MANAGE-PASS  = NONE
  AUDIT-AUTHENTICATE-FAIL  = NONE         AUDIT-MANAGE-FAIL  = NONE
  AUDIT-USER-ACTION-PASS   = NONE
  AUDIT-USER-ACTION-FAIL   = NONE

  TEXT-DESCRIPTION =

  CI-PROG = * NONE *
  CI-LIB  = * NONE *
  CI-NAME = * NONE *
  CI-SWAP = * NONE *
  CI-CPU  = ANY
  CI-PRI  = * NONE *
  CI-PARAM-TEXT =

  INITIAL-PROGTYPE      = PROGRAM
  INITIAL-PROGRAM       =
  INITIAL-DIRECTORY     =

 SUBJECT DEFAULT-PROTECTION SECTION UNDEFINED!

 SUBJECT OWNER-LIST SECTION UNDEFINED!
```

Add the group manager and specify a password:

```
=ADD USER admin.manager, 1,255, PASSWORD mx8z75
```

Verify the attributes of the user authentication record, using the DETAIL option of the INFO USER command:

```
=INFO USER 1,255, DETAIL
```

```
GROUP.USER    USER-ID  OWNER LAST-MODIFIED  LAST-LOGON STATUS WARNING-MODE
ADMIN.MANAGER 1,255  255,255 15JUN05,  8:11 * NONE *   THAWED      OFF

   UID                          =        511
   USER-EXPIRES                 =    * NONE *
   PASSWORD-EXPIRES             = 15JUL05,  0:00
   PASSWORD-MAY-CHANGE          =    * NONE *
   PASSWORD-MUST-CHANGE EVERY   =    30 DAYS
   PASSWORD-EXPIRY-GRACE        =    * NONE *
   LAST-LOGON                   =    * NONE *
   LAST-UNSUCESSFUL-ATTEMPT     =    * NONE *
   LAST-MODIFIED                = 15JUN05,  8:11
   CREATION-TIME                = 15JUN05,  2:03
   FROZEN/THAWED                = THAWED
   STATIC FAILED LOGON COUNT    =          0
   STATIC-FAILED-LOGON-RESET    =    * NONE *
   GUARDIAN DEFAULT SECURITY    = OOOO
   GUARDIAN DEFAULT VOLUME      = $SYSTEM.NOSUBVOL

   CREATOR-USER-NAME            = SUPER.SUPER
   CREATOR-USER-TYPE            = USER  (255,255)
   CREATOR-NODENUMBER           = 86

   AUDIT-AUTHENTICATE-PASS  = NONE        AUDIT-MANAGE-PASS  = NONE
   AUDIT-AUTHENTICATE-FAIL  = NONE        AUDIT-MANAGE-FAIL  = NONE
   AUDIT-USER-ACTION-PASS   = NONE
   AUDIT-USER-ACTION-FAIL   = NONE

   TEXT-DESCRIPTION  =

   BINARY-DESCRIPTION-LENGTH =   0

   CI-PROG  = * NONE *
   CI-LIB   = * NONE *
   CI-NAME  = * NONE *
   CI-SWAP  = * NONE *
   CI-CPU  = ANY
   CI-PRI   = * NONE *
   CI-PARAM-TEXT =

   INITIAL-PROGTYPE      = PROGRAM
   INITIAL-PROGRAM       =
   INITIAL-DIRECTORY     =

   PRIMARY-GROUP  = ADMIN
   GROUP          = ADMIN

  SUBJECT DEFAULT-PROTECTION SECTION UNDEFINED!

  SUBJECT OWNER-LIST SECTION UNDEFINED!
```

In this sequence of commands, the owner of this record is by default the person who is adding the new user. In this case, the local super ID is the default owner.

Because ADMIN.MANAGER is the first user added to group number 1, this sequence of commands creates a new administrative group, ADMIN. You can define all the administrative groups on the system by adding the first user to each group.

After you add a group manager to the ADMIN group, the group manager can then add other new users to the ADMIN group.

For example, ADMIN.MANAGER starts a SAFECOM session and uses the following command to add a new user, ADMIN.BOB, to group 1 and to assign a password to ADMIN.BOB:

```
=ADD USER admin.bob, 1,0, PASSWORD q5s4
```

ADMIN.MANAGER uses the INFO USER command to verify the settings of the user authentication record for ADMIN.BOB:

```
=INFO USER 1,0, DETAIL
```

The display shows:

```
 GROUP.USER  USER-ID  OWNER   LAST-MODIFIED  LAST-LOGON   STATUS WARNING-MODE
 ADMIN.BOB    1,0     1,255   17JUN05, 10:11  * NONE *    THAWED       OFF

   UID                          =        256
   USER-EXPIRES                 =    * NONE *
   PASSWORD-EXPIRES             =    * NONE *
   PASSWORD-MAY-CHANGE          =    * NONE *
   PASSWORD-MUST-CHANGE EVERY   =    * NONE *
   PASSWORD-EXPIRY-GRACE        =    * NONE *
   LAST-LOGON                   =    * NONE *
   LAST-UNSUCESSFUL-ATTEMPT     =    * NONE *
   LAST-MODIFIED                = 17JUN05, 10:11
   CREATION-TIME                = 15JUN05, 02:03
   FROZEN/THAWED                = THAWED
   STATIC FAILED LOGON COUNT    =          0
   STATIC-FAILED-LOGON-RESET    =    * NONE *
   GUARDIAN DEFAULT SECURITY    = OOOO
   GUARDIAN DEFAULT VOLUME      = $SYSTEM.NOSUBVOL

   CREATOR-USER-NAME            = SUPER.SUPER
   CREATOR-USER-TYPE            = USER  (255,255)
   CREATOR-NODENUMBER           = 86

   AUDIT-AUTHENTICATE-PASS  = NONE        AUDIT-MANAGE-PASS  = NONE
   AUDIT-AUTHENTICATE-FAIL  = NONE        AUDIT-MANAGE-FAIL  = NONE
   AUDIT-USER-ACTION-PASS   = NONE
   AUDIT-USER-ACTION-FAIL   = NONE

   TEXT-DESCRIPTION = (Is supported on systems running H06.07 and later H-
series RVUs)

   BINARY-DESCRIPTION-LENGTH =   0 (Is supported on systems running H06.07 and
later H-series RVUs)

   CI-PROG  = * NONE *
   CI-LIB   = * NONE *
   CI-NAME  = * NONE *
   CI-SWAP  = * NONE *
   CI-CPU   = * NONE *
   CI-PRI   = * NONE *
   CI-PARAM-TEXT =

   INITIAL-PROGTYPE      = PROGRAM
   INITIAL-PROGRAM       =
   INITIAL-DIRECTORY     =

   PRIMARY-GROUP  = ADMIN
   GROUP          = ADMIN

  SUBJECT DEFAULT-PROTECTION SECTION UNDEFINED!

  SUBJECT OWNER-LIST SECTION UNDEFINED!
```

ADMIN.BOB should change his password immediately to insure its security. Therefore, on June 17, ADMIN.BOB uses the command interpreter PASSWORD program to change his password:

```
1> PASSWORD BigChill
```

ADMIN.MANAGER could have used the PASSWORD-EXPIRES attribute to force ADMIN.BOB to change his password immediately. For an example of the use of this attribute, see Forcing Immediate Expiration of a User Password on page 2-23.

Using SAFECOM to Manage User Access to Your System

The owner of a user authentication record can use SAFECOM to control these aspects of the user's ability to access the system:

- Ownership of the record can be transferred to another user.
- The user can be granted temporary access to the system.
- The user can be required to change his or her password periodically.
- The user can be granted a grace period during which his or her expired password can be changed.
- The user's ability to access the system can be frozen (temporarily suspended).
- Users or administrative groups can be deleted from the system.

The next subsections describe how to establish these controls.

Changing the Owner of a User Authentication Record

Many of the security attributes stored in a user authentication record can be changed with the ALTER USER command. However, only the primary and secondary owners of the authentication record, the primary owner's group manager, or the local super ID can change these attributes.

Because security is controlled by record owners, not by users themselves, each system or network protected by the Safeguard software can assign one or more user IDs to security administrators. If ownership of user authentication records is transferred to a security administrator, the security administrator then has complete control of the system-access controls that the Safeguard software enforces for those users.

For example, ADMIN.MANAGER could give the user authentication record for ADMIN.BOB to a security administrator (SECURITY.SUSAN) with the following

sequence of SAFECOM commands. ADMIN.MANAGER begins by displaying the current user attributes defined for ADMIN.BOB:

```
=INFO USER admin.bob
```

```
GROUP.USER USER-ID  OWNER  LAST-MODIFIED  LAST-LOGON   STATUS   WARNING-MODE
ADMIN.BOB     1,0   1,255  17JUN05, 11:22  * NONE *    THAWED       OFF
```

This INFO display shows that the user authentication record for ADMIN.BOB is currently owned by 1,255 (ADMIN.MANAGER). ADMIN.MANAGER now gives the user authentication record for ADMIN.BOB to SECURITY.SUSAN with this command:

```
=ALTER USER admin.bob, OWNER security.susan
```

Then SECURITY.SUSAN checks the INFO display:

```
=INFO USER admin.bob
```

```
GROUP.USER   USER-ID  OWNER  LAST-MODIFIED   LAST-LOGON  STATUS   WARNING-MODE
ADMIN.BOB      1,0    200,1  20JUN05, 11:25    * NONE *  THAWED       OFF
```

Now the display shows that 200,1 (SECURITY.SUSAN) owns the user authentication record for ADMIN.BOB. ADMIN.MANAGER has thus limited the ability to change the user authentication record for ADMIN.BOB to only three users: SECURITY.SUSAN, her group manager, and the super ID. Now that she owns this user authentication record, SECURITY.SUSAN can use the ALTER USER command to control the ability of ADMIN.BOB to access the system.

Granting a User Temporary Access to Your System

Occasionally, an installation needs to limit the period that a user has access to the system. For example, if you hire a contract programmer for only a few weeks, the security administrator might want to limit that programmer's ability to access the system to the term of employment.

To control the length of time that a user can access the system, specify a USER-EXPIRES date for the user. The USER-EXPIRES attribute is contained in every user authentication record. Its default value is no expiration date.

For example, assume you are the manager of the SOFTWARE group (user ID 4,255). The following sequence of SAFECOM commands adds a user whose ability to access the system expires on December 19, 2005.

Reset the default user attributes to predefined values:

```
=RESET USER
```

Set default values for the USER-EXPIRES and PASSWORD attributes:

```
=SET USER USER-EXPIRES Dec 19 2005
```

```
=SET USER PASSWORD b9v7
```

Next, enter a SHOW USER command to check the default attribute values:

```
=SHOW USER
```

```
TYPE        OWNER       WARNING-MODE
USER         4,255           OFF

  PASSWORD = b9v7
  USER-EXPIRES               = 19DEC05,  0:00
  PASSWORD-EXPIRES           =    * NONE *
  PASSWORD-MUST-CHANGE EVERY =    * NONE *
  PASSWORD-EXPIRY-GRACE      =    * NONE *
  GUARDIAN DEFAULT SECURITY  = OOOO
  GUARDIAN DEFAULT VOLUME    = $SYSTEM.NOSUBVOL

  AUDIT-AUTHENTICATE-PASS  = NONE        AUDIT-MANAGE-PASS  = NONE
  AUDIT-AUTHENTICATE-FAIL  = NONE        AUDIT-MANAGE-FAIL  = NONE
  AUDIT-USER-ACTION-PASS   = NONE
  AUDIT-USER-ACTION-FAIL   = NONE

  TEXT-DESCRIPTION =

  CI-PROG  = * NONE *
  CI-LIB   = * NONE *
  CI-NAME  = * NONE *
  CI-SWAP  = * NONE *
  CI-CPU   = * NONE *
  CI-PRI   = * NONE *
  CI-PARAM-TEXT =

  INITIAL-PROGTYPE      = PROGRAM
  INITIAL-PROGRAM       =
  INITIAL-DIRECTORY     =

 SUBJECT DEFAULT-PROTECTION SECTION UNDEFINED!

 SUBJECT OWNER-LIST SECTION UNDEFINED!
```

Add the temporary user, SOFTWARE.GEORGE (user ID 4,21):

```
=ADD USER software.george, 4,21
```

Check the status of the SOFTWARE.GEORGE user authentication record with an INFO USER command:

```
=INFO USER software.george, DETAIL
```

```
GROUP.USER      USER-ID OWNER  LAST-MODIFIED  LAST-LOGON  STATUS   WARNING-MODE
SOFTWARE.GEORGE  4,21   4,255  27JUN05, 14:37  * NONE *   THAWED       OFF

   ID                          =       1045
   USER-EXPIRES                = 19DEC05,  0:00
   PASSWORD-EXPIRES            =    * NONE *
   PASSWORD-MAY-CHANGE         =    * NONE *
   PASSWORD-MUST-CHANGE EVERY  =    * NONE *
   PASSWORD-EXPIRY-GRACE       =    * NONE *
   LAST-LOGON                  =    * NONE *
   LAST-UNSUCESSFUL-ATTEMPT    =    * NONE *
   LAST-MODIFIED               = 27JUN05, 14:37
   CREATION-TIME               = 15JUN05, 02:03
   FROZEN/THAWED               = THAWED
   STATIC FAILED LOGON COUNT   =          0
   STATIC-FAILED-LOGON-RESET   =    * NONE *
   GUARDIAN DEFAULT SECURITY   = OOOO
   GUARDIAN DEFAULT VOLUME     = $SYSTEM.NOSUBVOL

   CREATOR-USER-NAME           = SUPER.SUPER
   CREATOR-USER-TYPE           = USER  (255,255)
   CREATOR-NODENUMBER          = 86


   AUDIT-AUTHENTICATE-PASS  = NONE        AUDIT-MANAGE-PASS  = NONE
   AUDIT-AUTHENTICATE-FAIL  = NONE        AUDIT-MANAGE-FAIL  = NONE
   AUDIT-USER-ACTION-PASS   = NONE
   AUDIT-USER-ACTION-FAIL   = NONE

   TEXT-DESCRIPTION =

   BINARY-DESCRIPTION-LENGTH =   0

   CI-PROG  = * NONE *
   CI-LIB   = * NONE *
   CI-NAME  = * NONE *
   CI-SWAP  = * NONE *
   CI-CPU   = * NONE *
   CI-PRI   = * NONE *
   CI-PARAM-TEXT =

   INITIAL-PROGTYPE      = PROGRAM
   INITIAL-PROGRAM       =
   INITIAL-DIRECTORY     =

   PRIMARY-GROUP  = SOFTWARE
   GROUP          = SOFTWARE

  SUBJECT DEFAULT-PROTECTION SECTION UNDEFINED!

  SUBJECT OWNER-LIST SECTION UNDEFINED!
```

The general INFO USER report shows that the user's ability to access the system expires at midnight on December 18, 2005.

The ALTER USER command can be used to change a USER-EXPIRES date. For example, if the contract for SOFTWARE.GEORGE is extended one month, the

manager of the SOFTWARE group can change that user's USER-EXPIRES date with this command:

```
=ALTER USER software.george, USER-EXPIRES Jan 19 2006
```

The ALTER user command can also be used to remove an expiration date. For example, if SOFTWARE.GEORGE is hired as a permanent employee, the manager of the SOFTWARE group removes his USER-EXPIRES date with this command:

```
=ALTER USER software.george, USER-EXPIRES
```

Specifying a USER-EXPIRES attribute without a date has the effect of removing any existing USER-EXPIRES date.

Requiring Users to Change Their Passwords

The Safeguard password control mechanism protects a system against unauthorized access. Passwords are more effective when users change them periodically. You can use the PASSWORD-MUST-CHANGE attribute to require a user to change the password within a specified period.

As with most other attributes of a user authentication record, the value of PASSWORD-MUST-CHANGE can be set when a user is added to the system with ADD USER, and the owner of the user authentication record can later change the value with ALTER USER.

For example, SECURITY.SUSAN establishes a PASSWORD-MUST-CHANGE period for ADMIN.BOB with the ALTER USER command:

```
=ALTER USER admin.bob, PASSWORD-MUST-CHANGE EVERY 30 DAYS
```

Then she checks the user record with the GENERAL option of the INFO USER command:

```
=INFO USER admin.bob, GENERAL
```

```
GROUP.USER  USER-ID  OWNER  LAST-MODIFIED    LAST-LOGON    STATUS  WARNING-MODE
ADMIN.BOB    1,0     200,1  28JUN05, 14:09 28JUN05,  7:48  THAWED      OFF

   UID                          =        256
   USER-EXPIRES                 =    * NONE *
   PASSWORD-EXPIRES             = 28JUL05,  0:00
   PASSWORD-MAY-CHANGE          =    * NONE *
   PASSWORD-MUST-CHANGE EVERY   =    30 DAYS
   PASSWORD-EXPIRY-GRACE        =    * NONE *
   LAST-LOGON                   = 28JUN05,  7:48
   LAST-UNSUCESSFUL-ATTEMPT     =    * NONE *
   LAST-MODIFIED                = 28JUN05, 14:09
   CREATION-TIME                = 15JUN05, 02:03
   FROZEN/THAWED                = THAWED
   STATIC FAILED LOGON COUNT    =          0
   STATIC-FAILED-LOGON-RESET    =    * NONE *
   GUARDIAN DEFAULT SECURITY    = OOOO
   GUARDIAN DEFAULT VOLUME      = $SYSTEM.NOSUBVOL
```

The INFO USER report now shows that ADMIN.BOB must change his password at least once every 30 days or his password expires. The LAST-MODIFIED field shows

that SECURITY.SUSAN changed Bob's authentication record on June 28, 2005. The Safeguard software calculated a PASSWORD-EXPIRES date by adding the PASSWORD-MUST-CHANGE period to the current date (June 28). At this point, ADMIN.BOB has until July 28, 2005, to change his password.

Any user can use the command interpreter PASSWORD program to change his or her own password. You can also change your password during logon without using the PASSWORD program. This method of changing your password is described in the *Safeguard User's Guide*.

To continue the example, suppose the date is July 27, 2005, and ADMIN.BOB changes his password with the PASSWORD command:

```
8> PASSWORD x87d9
```

After changing his password, ADMIN.BOB uses the INFO USER command to determine the next date by which he must change his password:

```
9> SAFECOM INFO USER admin.bob, GENERAL
```

```
GROUP.USER   USER-ID OWNER LAST-MODIFIED  LAST-LOGON     STATUS   WARNING-MODE
ADMIN.BOB     1,0    200,1 27JUL05, 14:09 27JUL05,  8:02 THAWED         OFF

   UID                          =        256
   USER-EXPIRES                 =    * NONE *
   PASSWORD-EXPIRES             = 26AUG05,  0:00
   PASSWORD-MAY-CHANGE          =    * NONE *
   PASSWORD-MUST-CHANGE EVERY   =    30 DAYS
   PASSWORD-EXPIRY-GRACE        =    * NONE *
   LAST-LOGON                   = 27JUL05,  8:02
   LAST-UNSUCESSFUL-ATTEMPT     =    * NONE *
   LAST-MODIFIED                = 27JUL05, 14:09
   CREATION-TIME                = 15JUN05, 02:03
   FROZEN/THAWED                = THAWED
   STATIC FAILED LOGON COUNT    =          0
   STATIC-FAILED-LOGON-RESET    =    * NONE *
   GUARDIAN DEFAULT SECURITY    = OOOO
   GUARDIAN DEFAULT VOLUME      = $SYSTEM.NOSUBVOL
```

The Safeguard software has recalculated the PASSWORD-EXPIRES date. Because ADMIN.BOB changed his password on July 27 and his PASSWORD-MUST-CHANGE period is 30 days, his new PASSWORD-EXPIRES date is August 26.

This example illustrates the basic password expiration mechanism. When a PASSWORD-MUST-CHANGE period is first established or changed for a user, the Safeguard software calculates a PASSWORD-EXPIRES date by adding the PASSWORD-MUST-CHANGE period to the current date. Also, whenever the user changes the password, the Safeguard software calculates a new PASSWORD-EXPIRES date by adding the PASSWORD-MUST-CHANGE period to the current date.

The Safeguard configuration attribute PASSWORD-MAY-CHANGE can affect password expiration and the user's ability to change the password. For a description of the PASSWORD-MAY-CHANGE attribute, see Section 9, Configuration.

If July 28 passes and ADMIN.BOB forgets to change his password, he cannot log on to the system after that date because he has no expired password grace period. Unable

to log on, ADMIN.BOB calls his security administrator, SECURITY.SUSAN on July 29, and asks her to find out why he cannot log on. Susan runs SAFECOM and enters the following INFO USER command:

```
=INFO USER admin.bob
```

```
GROUP.USER  USER-ID OWNER  LAST-MODIFIED   LAST-LOGON     STATUS  WARNING-MODE
ADMIN.BOB     1,0   200,1  28JUN05, 14:09  27JUL05, 08:02 PSWD-EXP    OFF
```

The STATUS field in the short INFO USER report shows SECURITY.SUSAN that ADMIN.BOB's password has expired.

To restore ADMIN.BOB's ability to log on to the system, SECURITY.SUSAN uses the ALTER USER command to change his PASSWORD-EXPIRES date and the INFO USER command to check the results:

```
=ALTER USER admin.bob, PASSWORD-EXPIRES 29 jul 05, 18:00

=INFO USER admin.bob, GENERAL
```

```
GROUP.USER  USER-ID  OWNER  LAST-MODIFIED   LAST-LOGON     STATUS  WARNING-MODE
ADMIN.BOB    1,0     200,1  29JUL05,  8:38  27JUL05,  8:02 THAWED      OFF

   UID                          =        256
   USER-EXPIRES                 =    * NONE *
   PASSWORD-EXPIRES             = 29JUL05, 18:00
   PASSWORD-MAY-CHANGE          =    * NONE *
   PASSWORD-MUST-CHANGE EVERY   =    30 DAYS
   PASSWORD-EXPIRY-GRACE        =    * NONE *
   LAST-LOGON                   = 27JUL05,  8:02
   LAST-UNSUCESSFUL-ATTEMPT     =    * NONE *
   LAST-MODIFIED                = 29JUL05,  8:38
   CREATION-TIME                = 15JUN05, 02:03
   FROZEN/THAWED                = THAWED
   STATIC FAILED LOGON COUNT    =          0
   STATIC-FAILED-LOGON-RESET    =    * NONE *
   GUARDIAN DEFAULT SECURITY    = OOOO
   GUARDIAN DEFAULT VOLUME      = $SYSTEM.NOSUBVOL
```

The INFO USER report shows that ADMIN.BOB's status is now THAWED (that is, he can log on to the system). Bob has until 6:00 p.m. on July 29 to change his password. When ADMIN.BOB changes his password, the Safeguard software recalculates the PASSWORD-EXPIRES date by adding the PASSWORD-MUST-CHANGE period to the current date.

Note. If you set the user PASSWORD-EXPIRES attribute after setting the PASSWORD-MUST-CHANGE attribute, the explicit setting of the PASSWORD-EXPIRES attribute overrides the PASSWORD-EXPIRES date previously calculated as a result of setting PASSWORD-MUST-CHANGE.

If you set the user PASSWORD-MUST-CHANGE attribute after setting the PASSWORD-EXPIRES attribute, the PASSWORD-EXPIRES date calculated as a result of setting PASSWORD-MUST-CHANGE overrides the explicit setting of the PASSWORD-EXPIRES attribute. That is, whichever attribute is set last takes precedence over the other setting.

Granting a Grace Period for Changing an Expired Password

You can use the PASSWORD-EXPIRY-GRACE attribute to specify a grace period during which a user can change his or her expired password. The PASSWORD-EXPIRY-GRACE attribute can be specified either in the user authentication record for an individual user or in the Safeguard configuration record for all users. If the grace period is specified in both records, the value in the user authentication record takes precedence.

For example, assume that SECURITY.SUSAN wants to grant ADMIN.BOB a grace period of 10 days during which he can change his password if he allows it to expire. She enters this ALTER USER command:

```
ALTER USER admin.bob, PASSWORD-EXPIRY-GRACE 10 DAYS
```

She then displays the user record to verify the results of the command:

```
INFO USER admin.bob, GENERAL
```

```
GROUP.USER  USER-ID  OWNER   LAST-MODIFIED  LAST-LOGON      STATUS WARNING-MODE
ADMIN.BOB     1,0    200,1  29JUL05,  8:56  27JUL05,  8:02  THAWED     OFF

   UID                          =        256
   USER-EXPIRES                 =    * NONE *
   PASSWORD-EXPIRES             = 28AUG05,  0:00
   PASSWORD-MAY-CHANGE          =    * NONE *
   PASSWORD-MUST-CHANGE EVERY   =    30 DAYS
   PASSWORD-EXPIRY-GRACE        =    10 DAYS
   LAST-LOGON                   = 27JUL05,  8:02
   LAST-UNSUCESSFUL-ATTEMPT     =    * NONE *
   LAST-MODIFIED                = 29JUL05,  8:56
   CREATION-TIME                = 15JUN05, 02:03
   FROZEN/THAWED                = THAWED
   STATIC FAILED LOGON COUNT    =          0
   STATIC-FAILED-LOGON-RESET    =    * NONE *
   GUARDIAN DEFAULT SECURITY    = OOOO
   GUARDIAN DEFAULT VOLUME      = $SYSTEM.NOSUBVOL
```

The general INFO USER report shows that ADMIN.BOB now has a grace period of 10 days in which to change an expired password. If ADMIN.BOB allows his password to expire, he can change it during the grace period. To change his expired password, ADMIN.BOB must log on during the grace period. He cannot use the PASSWORD program during this period because he cannot log on until the expired password is changed. For more information on logon dialog, see the *Safeguard User's Guide*.

Forcing Immediate Expiration of a User Password

You can use the PASSWORD-EXPIRES attribute to cause the immediate expiration of a user password. This feature can be particularly useful when you want a new user to change his or her password during their first logon attempt. To accomplish this, add the user with an expired password and grant a grace period during which the user can change the password.

For example, assume that the current time is 10:14 on July 29, 2005. To add the new user ADMIN.ALICE with an expired password and a password expiry grace period of five days, ADMIN.MANAGER enters this command:

```
=ADD USER admin.alice, 1,6, LIKE admin.bob, PASSWORD abc,&
=PASSWORD-EXPIRES 29 jul 2005, 10:00,&
=PASSWORD-EXPIRY-GRACE 5 DAYS
```

The PASSWORD-EXPIRES attribute specifies a time that has already passed. Therefore the user password is expired.

ADMIN.MANAGER then displays the user record to verify the results of the command:

```
INFO USER admin.alice, GENERAL
```

```
GROUP.USER    USER-ID  OWNER   LAST-MODIFIED  LAST-LOGON   STATUS  WARNING-MODE
ADMIN.ALICE   1,6      200,1   29JUL05, 10:14   * NONE *  PSWD-EXP      OFF

   UID                          =        262
   USER-EXPIRES                 =    * NONE *
   PASSWORD-EXPIRES             = 29JUL05, 10:00
   PASSWORD-MAY-CHANGE          =    * NONE *
   PASSWORD-MUST-CHANGE EVERY   =    30 DAYS
   PASSWORD-EXPIRY-GRACE        =     5 DAYS
   LAST-LOGON                   =    * NONE *
   LAST-UNSUCESSFUL-ATTEMPT     =    * NONE *
   LAST-MODIFIED                = 29JUL05,  10:14
   CREATION-TIME                = 15JUN05, 02:03
   FROZEN/THAWED                = THAWED
   STATIC FAILED LOGON COUNT    =          0
   STATIC-FAILED-LOGON-RESET    =    * NONE *
   GUARDIAN DEFAULT SECURITY    = OOOO
   GUARDIAN DEFAULT VOLUME      = $SYSTEM.NOSUBVOL
```

The display shows that Alice has five days in which to log on and change her password.

The PASSWORD-EXPIRES attribute can also be set to a future date. However, this date is altered if you subsequently set the user PASSWORD-MUST-CHANGE attribute or if the user changes the password before expiration.

Freezing a User's Ability to Access the System

Security administrators occasionally need to suspend a user's ability to log on to the system. For example, when a user goes on vacation, a security administrator might want to ensure that nobody else uses that user's identity to gain access to the system while the user is away. A security administrator can use the FREEZE USER command to freeze a user ID and its associated user name. While a user ID is frozen, nobody can use the user ID or its associated user name to gain access to the system.

However, freezing a user authentication record has no effect on user aliases associated with the user ID. The user can still log on using an alias.

For example, suppose ADMIN.BOB goes on vacation. SECURITY.SUSAN freezes the ADMIN.BOB user name with the FREEZE USER command and displays the record with the INFO USER command:

```
=FREEZE USER admin.bob

=INFO USER admin.bob
```

```
GROUP.USER  USER-ID  OWNER  LAST-MODIFIED   LAST-LOGON    STATUS   WARNING-MODE
ADMIN.BOB     1,0    200,1  5AUG05, 16:45  5AUG05,  8:07  FROZEN       OFF
```

The STATUS field in the short INFO USER report shows the status of ADMIN.BOB is now frozen.

When ADMIN.BOB returns from vacation, SECURITY.SUSAN restores his ability to log on to the system with the THAW USER command and displays the record with the INFO USER command:

```
=THAW USER admin.bob

=INFO USER admin.bob
```

```
GROUP.USER  USER-ID  OWNER  LAST-MODIFIED  LAST-LOGON     STATUS  WARNING-MODE
ADMIN.BOB     1,0    200,1  15AUG05,  8:15 5AUG05,  8:07  THAWED      OFF
```

Now the INFO USER display shows that record for ADMIN.BOB is thawed, and he can once again log on to the system.

Specifying Auditing for a User ID

The primary and secondary owners of a user authentication record can specify auditing for three types of events. These attributes are used to specify auditing for user authentication, such as attempts to log on to the system:

```
AUDIT-AUTHENTICATE-PASS
AUDIT-AUTHENTICATE-FAIL
```

These attributes are used to specify auditing for attempts to manage the user authentication record:

```
AUDIT-MANAGE-PASS
AUDIT-MANAGE-FAIL
```

These attributes are used to specify auditing for attempts by the user to perform an event:

```
AUDIT-USER-ACTION-PASS
AUDIT-USER-ACTION-FAIL
```

For more information about specifying these types of auditing, see the *Safeguard Audit Service Manual*.

Deleting Users

The primary and secondary owners of a user authentication record can delete that user with the DELETE USER command. For example, SECURITY.SUSAN can delete ADMIN.BOB with this command:

```
=DELETE USER admin.bob
```

Note. After deleting a user, the security administrator should notify users to remove the deleted user ID from access control lists for objects they own. Also, objects that the deleted user ID owns should be transferred to other owners or deleted. Until all references to a deleted user ID are removed, the user ID cannot be safely reused.

After these precautions are taken, the deleted user ID can be reassigned to a new user.

To remove a deleted user from an access control list, you must designate that user by user ID, not by user name. For example, the following commands remove ADMIN.BOB (user ID 1,0) from the access control lists for all disk files on the system:

```
=ALTER DISKFILE $*.*.*, ACCESS 1,0 - *

=ALTER DISKFILE $*.*.*, ACCESS 1,0 - DENY *
```

Use the same set of commands specifying the other object types to be sure that ADMIN.BOB is removed from all access control lists.

To remove a user from a diskfile-pattern protection record:

```
=ALTER DISKFILE-PATTERN $*.*.*, ALL, ACCESS 1,0 - *, ACCESS 1,0 – DENY *
```

Deleting Administrative Groups

An administrative group that was created with the ADD USER command is deleted automatically when the last member of the group is deleted. Just as the user name and user ID of a deleted user can be reassigned to a new user, the group name and number of a deleted group can be reassigned to a new group. The local super ID (or anyone authorized to add users) can reassign the deleted group name and number to a new administrative group by adding the first member of the new group with that group name or group number.

Automatic deletion does apply to an administrative group created with the ADD GROUP command. For more information, see Section 3, Managing User Groups.

Using SAFECOM to Establish a Network of Users

Users can be granted access to nodes other than their own and can have access authority for remote objects. A user who can access objects on one or more remote nodes is called a network user.

Being a system user on one node of a network of HP NonStop systems does not make you a network user. Before you can access objects on a remote node, you must be defined as a network user on your local node and on the remote node.

This requirement is an important feature of both the Safeguard subsystem and the standard security system. Defining a network user requires that the user be given the same user name, user ID, and remote password at both nodes. A user alias can also be defined as a network user by giving the same alias the same user ID and remote passwords at both nodes. Once a network user has been given the ability to access a remote node, that ability can be revoked at either the user's local node or at the remote node.

Using Safeguard With Nodes With Standard Security

The Safeguard subsystem is fully integrated with the standard security system. Thus, some nodes can be protected by Safeguard software, while other nodes are protected by the standard security system. On the nodes with only the standard security system, system managers must use the TACL command interpreter to add users and define remote passwords. On the nodes protected by Safeguard software, system managers can use either SAFECOM or TACL to add users and define remote passwords for users. However, the Safeguard software gives system managers more control over user access to their system than they would have with the standard security system.

On a system protected only by the standard security system, any local user can define his or her own remote passwords (provided the user has execute access to the RPASSWRD program object file). On a system protected by Safeguard software, the RPASSWRD program can be removed. Then only the owner of a user authentication record, the owner's group manager, and the local super ID can define a remote password for a user.

User aliases are supported only between nodes on which D30 or later Safeguard is running. In addition, standard security programs cannot be used to manage user aliases.

Table 2-3 shows the relationship between the SAFECOM and standard security programs that manage user access to a network of HP NonStop systems.

Table 2-3. SAFECOM and Standard Security Programs Used to Manage Network Users

Function	SAFECOM Command	Standard Security Program
Add a local user	ADD USER	ADDUSER program
Define a remote password for a local user	ADD USER REMOTEPASSWORD ALTER USER REMOTEPASSWORD SET USER REMOTEPASSWORD	RPASSWRD program
Change or delete a remote password for a local user	ALTER USER REMOTEPASSWORD	RPASSWRD program
Delete a local user	DELETE USER	DELUSER program
Add a user alias	ADD ALIAS	Not supported
Define a remote password for a user alias	ADD ALIAS REMOTEPASSWORD ALTER ALIAS REMOTEPASSWORD SET ALIAS REMOTEPASSWORD	RPASSWRD program
Change or delete a remote password for a user alias	ALTER ALIAS REMOTEPASSWORD	RPASSWRD program
Delete a user alias	DELETE ALIAS	Not supported

Identifying Network Users

You identify a user as a network user on an access control list by preceding the user ID or user name with *. (backslash-star-period).

The network forms for user IDs and user names are:

Network user ID: *node*.*group number* , *member number*

Network user name: *node*.*group name*.*member name*

For example, if ADMIN.BOB (user ID 1,0) is a network user, either of these commands gives him READ and WRITE access to the disk file QUARTER1:

```
=ALTER DISKFILE quarter1, ACCESS \test.admin.bob (r,w)
```

or

```
=ALTER DISKFILE quarter1, ACCESS \test.1,0 (r,w)
```

A user cannot use the network form of a user name to log on to the system. The network form is used only for user identification within the Safeguard subsystem.

Granting a Network User Access to Objects on Your System

This subsection gives instructions for using SAFECOM to set up remote passwords for a network user. The SAFECOM ADD USER and ALTER USER commands in this procedure can normally be executed only by the local super ID or the local group manager.

Before a user on a remote system can access objects on your system, take these steps:

1. Add the remote user as a local user on your system. The user name and user ID you add to your system must be the same user name and user ID defined on the remote system.
2. On your system, give the remote user a remote password for your system.
3. On the remote system, the remote user must be given a matching remote password for your system.

As an example of establishing a network user, suppose your system is \NY and you want a user on system \LA to be able to access a disk file on your system. If the remote user's local user name is ADMIN.BOB, and his local user ID is 1,0, the following steps give him access to your local disk file:

On the local system, \NY

Add the remote user as a local user on your system, \NY:

```
=ADD USER ADMIN.BOB , 1,0
```

Give the remote user a remote password for your system, \NY:

```
=ALTER USER 1,0, REMOTEPASSWORD \NY xyz
```

On the remote system, \LA

The remote user is given a remote password for \NY:

```
=ALTER USER 1,0, REMOTEPASSWORD \NY xyz
```

ADMIN.BOB now has one-way network access between his home node (\LA) and your node (\NY). ADMIN.BOB at \LA can be granted the authority to access objects on your system (\NY).

However, if ADMIN.BOB logs on to \NY, he cannot access his own objects on \LA from \NY. Although ADMIN.BOB has matching remote passwords for \NY on both \NY and \LA, he does not have matching remote passwords for \LA on both \LA and \NY. Before a network user can access objects on two systems from either system, he must have matching remote passwords for both systems set up on both systems.

Continuing with the example, the network user ADMIN.BOB can be granted two-way access between \LA and \NY if the system managers at both systems take these additional steps:

On the remote system, \LA

ADMIN.BOB is given a remote password for his system, \LA:

```
=ALTER USER 1,0, REMOTEPASSWORD \LA abc
```

On the local system, \NY

Give ADMIN.BOB a remote password for his system, \LA:

```
=ALTER USER 1,0, REMOTEPASSWORD \LA abc
```

Now the network user ADMIN.BOB has two-way access between \NY and \LA. The difference between one-way access and two-way access for a network user can be diagrammed as:

ONE-WAY ACCESS (from \LA to \NY):

```
System \NY                          System \LA

User Name      : ADMIN.BOB          User Name      : ADMIN.BOB
User ID        : 1,0                User ID        : 1,0
Remote Password: \NY xyz            Remote Password: \NY xyz
```

With these remote passwords, ADMIN.BOB can access objects on \NY from \LA.

ONE-WAY ACCESS (from \NY to \LA):

```
System \NY                          System \LA

User Name      : ADMIN.BOB          User Name      : ADMIN.BOB
User ID        : 1,0                User ID        : 1,0
Remote password: \LA abc            Remote password: \LA abc
```

With these remote passwords, ADMIN.BOB can access objects on \LA from \NY.

TWO-WAY ACCESS (between \NY and \LA):

```
System \NY                          System \LA

User Name      : ADMIN.BOB          User Name      : ADMIN.BOB
User ID        : 1,0                User ID        : 1,0
Remote password: \NY xyz            Remote password: \NY xyz
Remote password: \LA abc            Remote password: \LA abc
```

With these remote passwords, ADMIN.BOB can access objects on \NY from \LA, and he can access objects on \LA from \NY.

Establishing a Community of Network Users

Because network users must have the same user name and user ID on every node to which they have access, the assignment of group names to group numbers must be coordinated at each node. For example, a user at the node \LA named ADMIN.BOB with a user ID of 1,0 cannot be granted access to any system where the group number 1 is assigned to a group name other than ADMIN. Also, ADMIN.BOB cannot

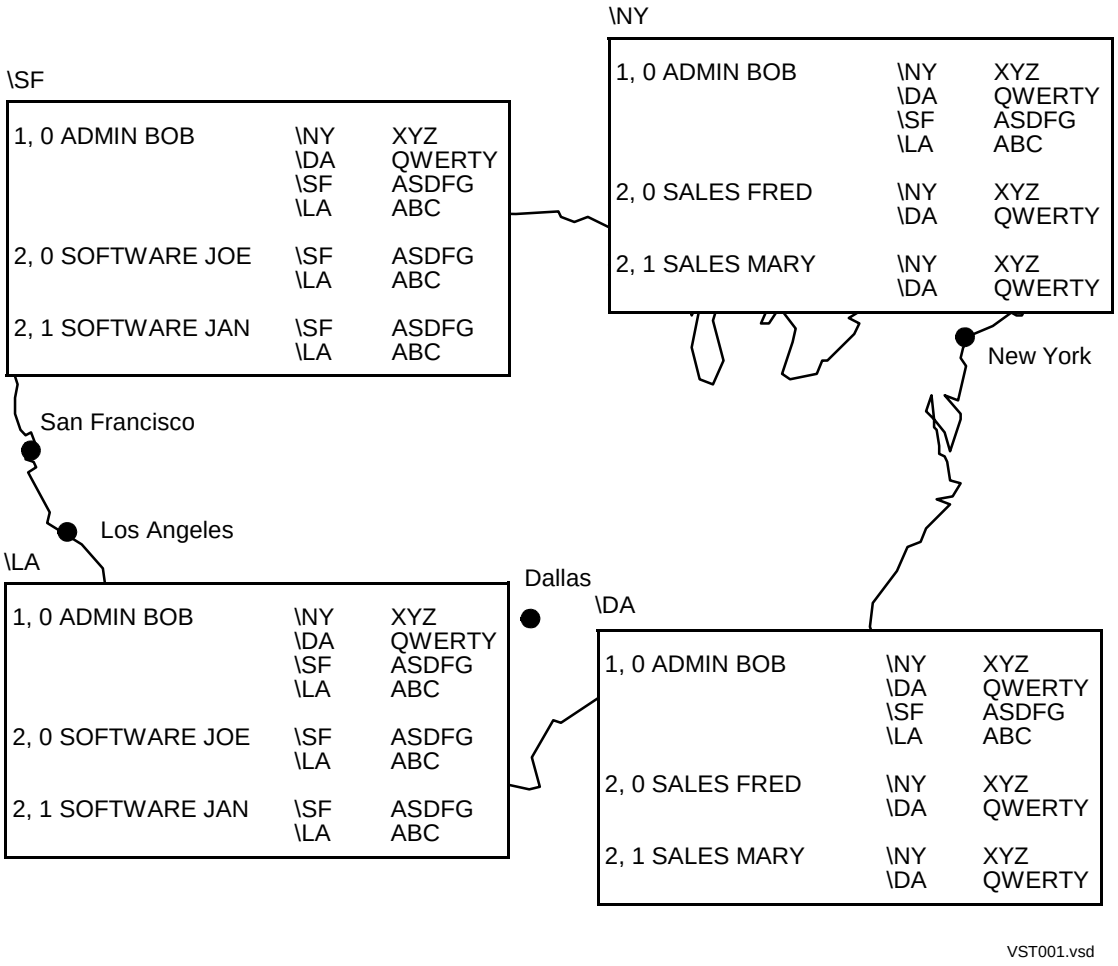
be granted access to any system on which the user ID 1,0 is assigned to another user name, such as, ADMIN.CAROL. (The use of user aliases as network users can alter this behavior, as described at the end of this subsection.)

Coordination of group names and numbers across a network also means that an administrative group can be defined as a network group or as a local group. A local group is unique to one node. For example, Figure 2-1 shows a network on which the group number 1, ADMIN, is defined on every node of the network, while group number 2 is assigned to the SOFTWARE group on \SF and \LA and to the SALES group on \NY and \DA.

In the network user community shown in Figure 2-1, ADMIN.BOB is defined on every node on the network and thus has access to every node on the network. But SOFTWARE.JOE has access only to \SF and \LA, and SALES.FRED has access only to \NY and \DA. In fact, network access is effectively segmented between the SOFTWARE and SALES groups. No member of the SOFTWARE group can access objects on \NY or \DA, and no member of the SALES group can access objects on \SF or \LA.

For each node illustrated in Figure 2-1, a partial list of the system users on each node is shown, along with the user ID, user name, and all remote passwords defined for the user.

Figure 2-1. A Community of Network Users



The concept of network groups and local groups does not apply if you define user aliases as network users. Administrative group names need not be coordinated for user aliases functioning as network users. Group names are not checked during the remote validation of a user alias. However, group numbers are checked.

For instance, assume that in the previous example, SALES.FRED at the node \NY has the user alias Freddie defined as follows. If an alias authentication record for Freddie is defined with a matching user ID and matching remote passwords at \SF, SALES.FRED can access objects on \SF from \NY regardless of the group name associated with the alias Freddie at \SF. Similarly, the user with the alias Freddie at \SF can access objects on \NY.

System \NY

```
Alias           : Freddie
User Name       : SALES.FRED
User ID         : 2,0
Remote password: \NY xyz
Remote password: \SF asdfg
```

System \SF

```
Alias           : Freddie
User Name       : SOFTWARE.JOE
User ID         : 2,0
Remote password: \NY xyz
Remote password: \SF asdfg
```

With these remote passwords, SALES.FRED can access objects on \SF when he is logged on with the alias Freddie at \NY. SOFTWARE.JOE can access objects on \NY when he is logged on with alias Freddie at \SF. However, Safeguard access control decisions are based on the underlying user ID of the alias at the remote node. In effect, SALES.BOB has access to objects to which SOFTWARE.JOE is normally granted access at \SF, and vice versa.

Changes to the PAID During a User Session

Prior to D30 Safeguard, remote validation is always based on the PAID of the process running on behalf of the requesting user. In most instances, the PAID is the same as the user ID of the user who initially logged on at the start of the session. Under certain circumstances, such as when the user executes a PROGID program, the PAID is changed so that it no longer matches the original user ID. For remote validation to be successful in this instance, matching remote passwords must exist for the user ID identified by the PAID.

Remote validation involving systems running D30 Safeguard functions similarly, with one distinct exception. If the user originally logged on as an alias, and the PAID of the process running on behalf of the alias remains unchanged, the alias name rather than the PAID is used for remote validation. (For more information, see User Aliases.) If the PAID has changed during the session, the user ID identified by PAID is used for remote validation.

Additional Considerations for Aliases and Groups

D30 and later Safeguard offers features that include the support of user aliases and file-sharing groups. When you define users for remote access, you should be aware of certain additional considerations regarding these features.

Additional Considerations for ACCESS with Network Specific Subject IDs

The global configuration attribute ALLOW-NODE-ID-ACL defines whether ACL entries containing explicit node identifiers for subjects are consulted to determine access. The initial ALLOW-NODE-ID-ACL value is off, ignoring ACL entries containing explicit node identifiers.

User Aliases

If a user who logs on as an alias is to have remote access to another node, alias authentication records with matching underlying user IDs and remote passwords must be defined on both the local and remote nodes. When an alias attempts remote access, the alias authentication record on the remote node is checked for a matching user ID and remote password. The underlying user name of the alias requesting access is not verified on the remote node. The same alias can have a different underlying user names at different nodes.

However, the underlying user ID defined for the alias at the remote node is still used in access decisions based on Safeguard access control lists at that node.

If the remote node is running a product version prior to D30 and does not support user aliases, the user ID identified by the PAID requesting the access is verified, and access decisions are based on that user ID.

If the local node is running a product version prior to D30 and does not support user aliases, the request cannot originate from an alias.

Group Lists

Effective with D30 Safeguard, users and aliases can belong to multiple groups. Each user and alias has a group list that specifies group membership. Any user or alias can belong to any group, no matter what kind of a group it is. When a subject is validated at a remote node, the group list used is the one associated with that user or alias authentication record at the remote node.

D30 and later Safeguard systems can determine the user ID or alias under which a user originally logged on at the start of a session even when the PAID is changed during the session. This original name, known as the login name, is used to determine the subject’s group list at the remote node. Although the user or alias was validated under a different PAID at the remote node, the group list is taken from the remote node’s authentication record associated with the requestor’s login name.

If no group list is specified at the remote node, the group list is empty, and process' group membership is indicated by its effective group ID. (The effective group ID is the same as the user's primary group ID unless it has been changed during the session.)

If either the local node or the remote node is running a product version prior to D30 and does not support group lists, the administrative group of the user ID associated with the requesting PAID is the only group to which the user belongs.

Establishing Default Protection for a User's Disk Files

The owner of a user authentication record can specify default protection for a user's disk files. With default protection, when a user creates a disk file, it is automatically added to the Safeguard database with the specified settings. If no default protection is specified, new files created by the user are protected by Guardian security unless the user explicitly adds them to the Safeguard database. You control default protection with the DEFAULT-PROTECTION attribute of the user authentication record.

You can specify the following DEFAULT-PROTECTION attributes for disk files that a user creates:

- A default access control list
- A default owner of the disk-file authorization record

Default auditing specifications

Note. Before using DEFAULT-PROTECTION, you might need to convert the USERID file. For more information, see Section 10, Installation and Management.

Note. Specifying DEFAULT-PROTECTION when CHECK-DISKFILE-PATTERN is set to ONLY results in the creation of extraneous normal protection records, which will not be examined because ONLY looks at only pattern protection records.

If default protection is specified for a user's disk files, the Safeguard software automatically creates protection records for that user's files even if the user has not been granted authority to add disk files to the Safeguard database. A user can be granted or denied such authority with the OBJECTTYPE DISKFILE command, which is described in Section 5, OBJECTTYPE Control.

Establishing a Default Access Control List

Assume that SECURITY.SUSAN owns the authentication record for ADMIN.JEFF, user ID 1,12. She can establish a default access control list that applies to any disk files that ADMIN.JEFF creates. Because the authentication record for ADMIN.JEFF already exists, SECURITY.SUSAN uses the ALTER USER command to change the record to include DEFAULT-PROTECTION:

```
=ALT USER 1,12, DEFAULT-PROTECTION ACCESS \200.1,12 *; 1,* (R,E)
```

This command gives all authorities to ADMIN.JEFF and gives READ and EXECUTE authority to any other member of group 1, the ADMIN group. These authorities apply only to disk files that ADMIN.JEFF creates after DEFAULT-PROTECTION is specified.

To check the DEFAULT-PROTECTION settings, SECURITY.SUSAN issues the INFO USER command with the DEFAULT-PROTECTION option:

```
=INFO USER 1,12, DEFAULT-PROTECTION
```

The display shows:

```
GROUP.USER  USER-ID  OWNER  LAST-MODIFIED   LAST-LOGON     STATUS  WARNING-MODE
ADMIN.JEFF  1,12     200,1  15AUG05,  8:55  12AUG05, 16:02 THAWED       OFF

  SUBJECT DEFAULT-PROTECTION SECTION

  OWNER=       1,12
  AUDIT-ACCESS-PASS = NONE      AUDIT-MANAGE-PASS = NONE
  AUDIT-ACCESS-FAIL = NONE      AUDIT-MANAGE-FAIL = NONE

                         001,012      R,W,E,P, O
                         001, *       R,  E
                    \200.001,012      R,W,E,P, O
```

Because ADMIN.JEFF was given all authorities, including OWNER authority, he can change the protection on his individual files if he wants to. He cannot change the DEFAULT-PROTECTION, however, because he does not own the user authentication record for ADMIN.JEFF.

Establishing Default Ownership

You can specify two types of ownership with DEFAULT-PROTECTION. You specify one with the OWNER attribute of the user record, and you specify the other with OWNER authority in an access control list. Both types of owners can modify the disk file authorization record after it is created. However, only the primary owner, secondary owners, or a super ID can set the PROGID attribute to protect program code. Also, if the disk file is removed from the Safeguard database, the primary owner becomes the Guardian owner.

If no primary owner is specified, as in the previous example, the primary owner is the user for which DEFAULT-PROTECTION is being specified: in this case, ADMIN.JEFF. However, if SECURITY.SUSAN needs to be the primary owner of all new files that ADMIN.JEFF creates, she can specify her user ID for the OWNER attribute. For example:

```
=ALTER USER 1,12, DEFAULT-PROTECTION OWNER 200,1
```

SECURITY.SUSAN displays the authentication record for ADMIN.JEFF to verify that she is the default owner of any files he creates:

```
=INFO USER 1,12, DEFAULT-PROTECTION
```

The display shows:

```
GROUP.USER  USER-ID  OWNER  LAST-MODIFIED  LAST-LOGON      STATUS  WARNING-MODE
ADMIN.JEFF  1,12     200,1  15AUG05,  9:05 12AUG05, 16:02  THAWED      OFF

  SUBJECT DEFAULT-PROTECTION SECTION

  OWNER=     200,1
  AUDIT-ACCESS-PASS = NONE      AUDIT-MANAGE-PASS = NONE
  AUDIT-ACCESS-FAIL = NONE      AUDIT-MANAGE-FAIL = NONE

                          001,012      R,W,E,P,  O
                          001, *       R,  E
```

Specifying Default Audit Attributes

DEFAULT PROTECTION also includes these four audit attributes for disk files:

- AUDIT-ACCESS-PASS
- AUDIT-ACCESS-FAIL
- AUDIT-MANAGE-PASS
- AUDIT-MANAGE-FAIL

These audit attributes are explained in detail in the *Safeguard Audit Service Manual.*

Assume that SECURITY.SUSAN wants to specify default auditing for all files that ADMIN.JEFF creates. To do so, she sets DEFAULT-PROTECTION to include auditing of all successful attempts to access Jeff's disk files:

```
=ALTER USER 1,12, DEFAULT-PROTECTION &
=AUDIT-ACCESS-PASS ALL
```

Then she displays the record to verify the DEFAULT-PROTECTION audit settings:

```
=INFO USER 1,12, DEFAULT-PROTECTION
```

```
GROUP.USER  USER-ID  OWNER  LAST-MODIFIED  LAST-LOGON     STATUS   WARNING-MODE
ADMIN.JEFF  1,12     200,1  15AUG05,  9:35 12AUG05, 16:02 THAWED        OFF

  SUBJECT DEFAULT-PROTECTION SECTION

  OWNER=     200,1
  AUDIT-ACCESS-PASS = ALL       AUDIT-MANAGE-PASS = NONE
  AUDIT-ACCESS-FAIL = NONE      AUDIT-MANAGE-FAIL = NONE

                          001,012      R,W,E,P,  O
                          001, *       R,  E
```

Eliminating Default Protection for a User

If you decide to eliminate default protection for a user, specify the DEFAULT-PROTECTION attribute with no default attributes following it.

For example, SECURITY.SUSAN issues this command to eliminate default protection for ADMIN.JEFF:

```
=ALTER USER 1,12, DEFAULT-PROTECTION
```

Now any new files created by ADMIN.JEFF are protected by Guardian security unless they are explicitly added to the Safeguard database.

Specifying a Default Command Interpreter for a User

When a user logs on at a Safeguard terminal, a specific command interpreter (process) is started automatically after the user has been authenticated. You can designate this command interpreter in the user authentication record for each user.

For example, assume that ADMIN.JEFF uses SAFECOM for most of his daily activities and that his terminal is controlled by the Safeguard software. SECURITY.SUSAN can specify SAFECOM as the default command interpreter in Jeff's user authentication record.

To do so, she uses the ALTER USER command to specify the name of the SAFECOM object file on Jeff's system:

```
ALTER USER admin.jeff, CI-PROG $system.system.safecom
```

Then she issues the INFO USER command with the CI option to check the results:

```
INFO USER admin.jeff, CI
```

The display shows:

```
GROUP.USER  USER-ID  OWNER   LAST-MODIFIED  LAST-LOGON     STATUS  WARNING-MODE
ADMIN.JEFF   1,12    200,1  15AUG05, 11:54  12AUG05, 16:02 THAWED       OFF

   CI-PROG  = $SYSTEM.SYSTEM.SAFECOM
   CI-LIB   = * NONE *
   CI-NAME  = * NONE *
   CI-SWAP  = * NONE *
   CI-CPU   = * NONE *
   CI-PRI   = * NONE *
   CI-PARAM-TEXT =
```

Now SAFECOM is started automatically whenever ADMIN.JEFF logs on at a Safeguard terminal. For more information, see Section 7, Securing Terminals.

The INFO USER display shows that you can specify other optional attributes relating to the default command interpreter. For more information about these attributes, see the *Safeguard Reference Manual.*

Establishing Guardian Defaults

When you add a user authentication record to the Safeguard database, you can specify the Guardian default file-security string and the saved default volume and subvolume for that user. The Guardian default file-security string is given to any of the user disk files that are not under Safeguard protection. The user Guardian-saved default volume and subvolume are established each time the user logs on to the system or enters a VOLUME command without any parameters.

The GUARDIAN DEFAULT SECURITY attribute controls the Guardian default file-security string. When you set this attribute in the Safeguard user authentication record, it accomplishes the same function as using the DEFAULT program to set the security string. For more information about the security string and the DEFAULT program, see the *Safeguard User's Guide*.

The GUARDIAN DEFAULT VOLUME attribute controls the Guardian-saved default volume and subvolume. When you set this attribute in the Safeguard user authentication record, it accomplishes the same function as using the DEFAULT program to set the user saved default volume and subvolume.

Setting the File-Security String

If you do not specify a value for the GUARDIAN DEFAULT SECURITY attribute when you add a Safeguard user authentication record, that user is given a Guardian default security string of OOOO. This string indicates that when Guardian default protection is applied, only the local file owner, the owner's group manager, and the super ID have READ, WRITE, EXECUTE, and PURGE authority.

Assume that SECURITY.SUSAN wants to change the Guardian default security string for ADMIN.JEFF to NUNU. To do so, SECURITY.SUSAN uses this SAFECOM command:

```
=ALTER USER admin.jeff, GUARDIAN SECURITY 'NUNU'
```

The word DEFAULT in the GUARDIAN DEFAULT SECURITY attribute is optional when you enter the command. You can include it for readability, but it is not required. Similarly, quotes around the security string specifier are also optional.

To verify the results of the command, SECURITY.SUSAN issues this INFO USER command:

```
=INFO USER admin.jeff, GENERAL
```

```
GROUP.USER   USER-ID OWNER  LAST-MODIFIED  LAST-LOGON      STATUS WARNING-MODE
ADMIN.JEFF   1,12    200,1  15AUG05, 11:54 12AUG05, 16:02  THAWED      OFF

   UID                          =        268
   USER-EXPIRES                 =    * NONE *
   PASSWORD-EXPIRES             = 28AUG05,  0:00
   PASSWORD-MAY-CHANGE          =    * NONE *
   PASSWORD-MUST-CHANGE EVERY   =    30 DAYS
   PASSWORD-EXPIRY-GRACE        =    * NONE *
   LAST-LOGON                   = 12AUG05,  16:02
   LAST-UNSUCESSFUL-ATTEMPT     =    * NONE *
   LAST-MODIFIED                = 15AUG05,  11:54
   CREATION-TIME                = 15JUN05, 02:03
   FROZEN/THAWED                = THAWED
   STATIC FAILED LOGON COUNT    =          0
   STATIC-FAILED-LOGON-RESET    =    * NONE *
   GUARDIAN DEFAULT SECURITY    = NUNU
   GUARDIAN DEFAULT VOLUME      = $SYSTEM.NOSUBVOL
```

Specifying the Default Volume and Subvolume

If you do not specify a value for the GUARDIAN DEFAULT VOLUME attribute when you add a Safeguard user authentication record, that user is assigned the saved default volume and subvolume of $SYSTEM.NOSUBVOL on the local system.

Assume that SECURITY.SUSAN wants to change the Guardian saved default volume and subvolume for ADMIN.JEFF to $SECURE.JEFF. To do so, SECURITY.SUSAN uses this SAFECOM command:

```
=ALTER USER admin.jeff, GUARDIAN DEFAULT VOLUME $secure.jeff
```

The word DEFAULT in the GUARDIAN DEFAULT VOLUME attribute is optional, but it is included here for readability.

To verify the results of the command, SECURITY.SUSAN issues this INFO USER command:

```
=INFO USER admin.jeff, GENERAL
```

```
GROUP.USER   USER-ID OWNER  LAST-MODIFIED  LAST-LOGON      STATUS WARNING-MODE
ADMIN.JEFF   1,12    200,1  15AUG05, 11:54 12AUG05, 16:02  THAWED      OFF

   UID                          =        268
   USER-EXPIRES                 =    * NONE *
   PASSWORD-EXPIRES             = 28AUG05,  0:00
   PASSWORD-MAY-CHANGE          =    * NONE *
   PASSWORD-MUST-CHANGE EVERY   =    30 DAYS
   PASSWORD-EXPIRY-GRACE        =    * NONE *
   LAST-LOGON                   = 12AUG05,  16:02
   LAST-UNSUCESSFUL-ATTEMPT     =    * NONE *
   LAST-MODIFIED                = 15AUG05,  11:54
   CREATION-TIME                = 15JUN05, 02:03
   FROZEN/THAWED                = THAWED
   STATIC FAILED LOGON COUNT    =          0
   STATIC-FAILED-LOGON-RESET    =    * NONE *
   GUARDIAN DEFAULT SECURITY    = NUNU
   GUARDIAN DEFAULT VOLUME      = $SECURE.JEFF
```

Assigning an Alias to a User

A user can be given alternate names, called aliases. Each alias assigned to a user can have its own set of unique user attributes. For example, a single user can have several aliases, and each alias can have a different command interpreter specified. This allows the user to choose among several command interpreters by logging on with the appropriate alias to start a session.

An alias can be used to log on to the system, but an alias name cannot appear on an access control list. Because aliases cannot be specified on access control lists, all Safeguard access mediation for an alias is based on the underlying user ID.

Special restrictions apply to the use of the ADD ALIAS command to assign an alias to a user. To add an alias for a particular user, a person must have the authority to both add and alter that user authentication record. The specific restrictions regarding use of the ADD ALIAS command are:

- If an OBJECTTYPE USER record exists, the person executing the ADD ALIAS command must meet these two qualifications:
  - Have CREATE (C) authority on the OBJECTTYPE USER access control list
  - Be the owner of the underlying user ID or be the group manager of the owner of the underlying user ID
- If an OBJECTTYPE USER record does not exist, the person executing the ADD ALIAS command must meet these two qualifications:
  - Be the group manager of the underlying user ID
  - Be the owner of the underlying user ID or be the group manager of the owner of the underlying user ID

- In addition, the local super ID can add an alias for any user regardless of the existence of an OBJECTTYPE USER record (unless OBJECTTYPE USER specifically denies the super ID).

Each alias must be unique within the local system. An alias is a case-sensitive text string that can be up to 32 alphanumeric characters in length. In addition to alphabetic and numeric characters, the characters period (.), hyphen (-), and underscore (_) are permitted within the text string. The first character of an alias name must be alphabetic or numeric.

An alias name cannot match an existing user name when it is converted to uppercase letters. For example, the alias Dev12.PatM is not allowed if a user name DEV12.PATM already exists.

These names are examples of valid aliases:

```
BILL-J
my-new-name
juj_smails
Admin5
36checker
BBonds.660
YodaJones
```

This example shows how to assign an alias to a user. In the example, the group manager for the user SOFTWARE.RALPH (user ID 4,32) assigns the alias RalphW to the user:

```
=ADD ALIAS RalphW, software.ralph, PASSWORD BluSky
```

SOFTWARE.RALPH can now log on with alias RalphW. When he logs on using this alias, he must enter the password BluSky.

The alias RalphW can also be assigned its own unique set of user attributes that can differ from those assigned to the user name SOFTWARE.RALPH. For example, the group manager issues the following command to specify auditing for every access attempt performed by SOFTWARE.RALPH when he is logged on with the alias RalphW:

```
=ALTER ALIAS RalphW, AUDIT-USER-ACTION-PASS ALL, &
=AUDIT-USER-ACTION-FAIL ALL
```

The group manager then uses this command to verify the alias authentication record for RalphW:

```
=INFO ALIAS RalphW, DETAIL
```

The display shows:

```
NAME                  USER-ID    OWNER         STATUS     WARNING-MODE
RalphW                4,32       4,255         THAWED         OFF

  UID                          =       1056
  USER-EXPIRES                 =    * NONE *
  PASSWORD-EXPIRES             =    * NONE *
  PASSWORD-MAY-CHANGE          =    * NONE *
  PASSWORD-MUST-CHANGE EVERY   =    * NONE *
  PASSWORD-EXPIRY-GRACE        =    * NONE *
  LAST-LOGON                   =    * NONE *
  LAST-UNSUCESSFUL-ATTEMPT     =    * NONE *
  LAST-MODIFIED                = 12SEP05, 10:43
  CREATION-TIME                = 15JUN05, 02:03
  FROZEN/THAWED                = THAWED
  STATIC FAILED LOGON COUNT    =          0
  STATIC-FAILED-LOGON-RESET    =    * NONE *
  GUARDIAN DEFAULT SECURITY    = OOOO
  GUARDIAN DEFAULT VOLUME      = $SYSTEM.NOSUBVOL

  CREATOR-USER-NAME            = SUPER.SUPER
  CREATOR-USER-TYPE            = USER  (255,255)
  CREATOR-NODENUMBER           = 86

  AUDIT-AUTHENTICATE-PASS  = NONE        AUDIT-MANAGE-PASS  = NONE
  AUDIT-AUTHENTICATE-FAIL  = NONE        AUDIT-MANAGE-FAIL  = NONE
  AUDIT-USER-ACTION-PASS   = ALL
  AUDIT-USER-ACTION-FAIL   = ALL

  TEXT-DESCRIPTION =

  BINARY-DESCRIPTION-LENGTH =   0

  CI-PROG  = * NONE *
  CI-LIB   = * NONE *
  CI-NAME  = * NONE *
  CI-SWAP  = * NONE *
  CI-CPU   = * NONE *
  CI-PRI   = * NONE *
  CI-PARAM-TEXT =

  INITIAL-PROGTYPE      = PROGRAM
  INITIAL-PROGRAM       =
  INITIAL-DIRECTORY     =

  PRIMARY-GROUP  = SOFTWARE
  GROUP          = SOFTWARE

 SUBJECT DEFAULT-PROTECTION SECTION UNDEFINED!

 SUBJECT OWNER-LIST SECTION UNDEFINED!
```

When the user is logged on as RalphW, any attempt to access a protected object causes the Safeguard software to check the object's access control list to ensure that user ID 4,32 (the underlying user ID) has the proper access authority.

The attributes in an alias authentication record are unique and unrelated to those in the underlying user authentication record. In particular, an alias can have a unique password, and an alias must have its own remote passwords established for remote access.

An alias, like a user ID, can be deleted, frozen, and thawed. For more information regarding aliases, see the *Safeguard Reference Manual*.

3 Managing User Groups

This section describes how to use the SAFECOM group commands to define and manage supplementary user groups. Groups created explicitly with the ADD GROUP command can exist independently of user definitions and are typically used for file-sharing purposes.

Groups with numbers ranging from 0 through 255 can be used as administrative groups. An administrative group exists primarily for user management although it can also be used for file sharing. Groups numbered above 255 cannot be used for user management. These groups exist solely for file sharing, particularly in the OSS environment.

Administrative groups can be created in either of two ways. You can use an ADD USER command to add the first user to a nonexistent group. This action implicitly creates the administrative group. You can also explicitly create an administrative group with an ADD GROUP command and later activate that group with an ADD USER command.

Regardless of the manner in which a group is created, you can use the ALTER GROUP command to manage that group. For example, you can use the MEMBER option of an ALTER GROUP command to extend the membership of an administrative group beyond 256 users for file-sharing purposes.

Note. In prior product versions, GROUP commands were used to manage Safeguard security groups. GROUP commands are now used to manage file-sharing groups, as described in this section. Security groups are now managed with the SECURITY-GROUP commands, as described in Section 6, Managing Security Groups.

The attributes in a group definition record allow you to specify the group's name and number, a text description, the group owner, and a list of group members.

Table 3-1 lists the group commands and gives a brief description of each. These commands are illustrated in the examples in the remainder of this section.

Table 3-1. Group Command Summary

Command	Description
ADD GROUP	Adds a group definition record with the specified group attribute values.
ALTER GROUP	Changes one or more attribute values in a group definition record.
DELETE GROUP	Deletes a group definition record.
INFO GROUP	Displays the existing attribute values in a group definition record.

Adding User Groups

Any super-group member can add a group definition group unless an OBJECTTYPE USER access control list exists. If an OBJECTTYPE USER record exists, only users with CREATE authority on that access control list can use the ADD GROUP command.

Assume that the user ADMIN.DON (user ID 16,24) has C authority on the OBJECTTYPE USER access control list. To create a group that could be subsequently activated as an administrative group, ADMIN.DON enters this command:

```
=ADD GROUP PROG4, NUMBER 144, DESCRIPTION Maintenance &
=programmers for Inventory System
```

The group name PROG4 and group number 144 meet the syntactical requirements for administrative group names and numbers. A group name specified in a GROUP command is case-sensitive. Therefore, an administrative group name must be entered in uppercase letters.

ADMIN.DON could create another group named ProG4:

```
=ADD GROUP ProG4, NUMBER 1144, DESCRIPTION Inventory system &
=programmers
```

Two distinct groups, PROG4 and ProG4, now exist concurrently. But only PROG4 can serve as an administrative group. The group ProG4 was given the group number 1144 because it is advisable to reserve the group numbers below 256 for administrative groups. An administrative group must have a group number from 0 to 255.

The group PROG4 is considered a file-sharing group unless a user is added to it with the ADD USER command. If this occurs so that the group PROG4 is activated as an administrative group, it can be specified on Safeguard access control lists. Only administrative groups can appear on Safeguard access control lists. File-sharing groups cannot appear on access control lists. The group name ProG4 cannot be specified on access control lists, but it can be used for file-sharing in an OSS environment.

To verify the results of the commands, use the INFO GROUP command:

```
=INFO GROUP PROG4, DETAIL
```

The display shows:

```
 GROUP NAME                        NUMBER   OWNER   LAST-MODIFIED
 PROG4                               144    16,24   23JUL94, 11:16
    CREATION-TIME              = 15JUL05,  2:03
    CREATOR-USER-NAME          = SUPER.SUPER
    CREATOcR-USER-TYPE          = USER  (255,255)
    CREATOR-NODENUMBER         = 86
    AUTO-DELETE = OFF
    DESCRIPTION = Maintenance programmers for Inventory System
GROUP OWNER-LIST SECTION UNDEFINED!
```

```
=INFO GROUP ProG4, DETAIL
```

The display shows:

```
 GROUP NAME                          NUMBER   OWNER   LAST-MODIFIED
 ProG4                                1144    16,24   23JUL94, 11:18
    CREATION-TIME              = 15JUL05,  2:03
    CREATOR-USER-NAME          = SUPER.SUPER
    CREATOR-USER-TYPE          = USER  (255,255)
    CREATOR-NODENUMBER         = 86
    AUTO-DELETE = OFF
    DESCRIPTION = Inventory system programmers
GROUP OWNER-LIST SECTION UNDEFINED!
```

Because ADMIN.DON is the owner of these groups, he can use the ALTER GROUP command to manage the groups. In addition, because ADMIN.DON is on the OBJECTTYPE USER access control list, he can use the ADD USER command to add users to the group PROG4 and thereby activate it as an administrative group.

To add the first user authentication record to the group PROG4, Don enters this command:

```
=ADD USER prog4.susan, 144,1, PASSWORD tempFix
```

Because USER commands are not case-sensitive, the group name portion of the new user's user name need not be entered in uppercase.

Use the INFO GROUP command to verify the results of the command:

```
=INFO GROUP PROG4, DETAIL
```

The display shows:

```
 GROUP NAME                          NUMBER   OWNER   LAST-MODIFIED
 PROG4                                 144    16,24   23JUL94, 11:22
    CREATION-TIME              = 15JUL05,  2:03
    CREATOR-USER-NAME          = SUPER.SUPER
    CREATOR-USER-TYPE          = USER  (255,255)
    CREATOR-NODENUMBER         = 86
    AUTO-DELETE = OFF
    DESCRIPTION = Maintenance programmers for Inventory System
    MEMBER = PROG4.SUSAN
GROUP OWNER-LIST SECTION UNDEFINED!
```

Note. The attribute OWNER-LIST is supported only on systems running H06.25 and later H-series RVUs and J06.14 and later J-series RVUs.

Adding and Deleting Group Members

You can add file-sharing members to a group with the MEMBER option in an ADD GROUP or ALTER GROUP command. Assume that ADMIN.DON wants to add three users to the group PROG4 so that they can access certain files reserved for the PROG4 group. He enters this command:

```
=ALTER GROUP NUMBER 144, MEMBER (test.phil, test.june, &
=Group-Super)
```

This command adds the users TEST.PHIL and TEST.JUNE and the user alias Group-Super to the group PROG4. The group owner, ADMIN.DON, has no administrative control over the user and alias authentication records for these users, but he does control their membership in the group PROG4.

Use the INFO GROUP command to verify the results of the commands:

```
=INFO GROUP PROG4, DETAIL
```

The display shows:

```
 GROUP NAME                          NUMBER   OWNER   LAST-MODIFIED
 PROG4                                 144    16,24   23JUL94, 11:49
    CREATION-TIME              = 15JUL05,  2:03
    CREATOR-USER-NAME          = SUPER.SUPER
    CREATOR-USER-TYPE          = USER  (255,255)
    CREATOR-NODENUMBER         = 86
    AUTO-DELETE = OFF
    DESCRIPTION = Maintenance programmers for Inventory System
    MEMBER = PROG4.SUSAN
    MEMBER = TEST.PHIL
    MEMBER = TEST.JUNE
    MEMBER = Group-Super
GROUP OWNER-LIST SECTION UNDEFINED!
```

Note. The attribute OWNER-LIST is supported only on systems running H06.25 and later H-series RVUs and J06.14 and later J-series RVUs.

Now any access control list entry that contains the entry 144,* is interpreted by the Safeguard software to include all members of group 144, including TEST.PHIL, TEST.JUNE, and Group-Super.

Suppose that ADMIN.DON later decides that the user TEST.PHIL no longer needs access to the protected files. He enters this command to remove TEST.PHIL from the group:

```
=ALTER GROUP PROG4, MEMBER -test.phil
```

The minus sign at the beginning of the member list specifies that the users in the list are to be removed from the group.

Using Wild-cards for Managing Group Members

The ADD GROUP command and the ALTER GROUP command support the following wild-card characters for adding and/or deleting group members:

* : An asterisk (*) matches any number of characters (zero, one, or more).

? : A question mark (?) matches a single character.

Note. These wild-card characters are supported only on systems running J06.08 and later J-series RVUs and H06.19 and later H-series RVUs.

For more information on using wild-card characters, see the *Safeguard Reference Manual.*

The following example illustrates the use of wild-card characters for adding group members using the ADD GROUP command:

```
=ADD GROUP PROG5, NUMBER 144, MEMBER (test.p*, test.user?, &
=my*p.user?nam, Group-Super, newalia?us*r)
```

Note.

- MEMBER *.* adds all the users and aliases in the Guardian user name format to the group.
- MEMBER * adds all the users and aliases to the group.

The following example illustrates the use of wild-card characters to delete group members using the ALTER GROUP command:

```
= ALTER GROUP NUMBER 144,MEMBER -(test.p*,test.user?,&
= my*p.user?nam,Group-Super,newalia?us*r)
```

Note.

- MEMBER *.* deletes all the users and aliases in the Guardian user name format from the group.
- MEMBER * deletes all the users and aliases from the group.

The following example illustrates the use of wild-card characters to include and delete group members using the ALTER GROUP command:

```
= ALTER GROUP NUMBER 144,MEMBER (test.p*,test.user?,&
= my*p.user?nam,Group-Super,newalia?us*r),&
= MEMBER-(super*.user?,old?alias*)
```

Transferring Group Ownership

You can transfer ownership of a group definition record to another user. For example, ADMIN.DON enters this command to give ownership of the PROG4 group definition record to ADMIN.FRAN (16,3):

```
=ALTER GROUP PROG4, OWNER admin.fran
```

To verify the results:

```
=INFO GROUP PROG4
```

The display shows:

```
GROUP NAME                          NUMBER   OWNER   LAST-MODIFIED
PROG4                                 144    16,3    26JUL94, 13:20
```

ADMIN.FRAN is now responsible for managing membership in the PROG4 group. However, ADMIN.FRAN has no authority to manage the user authentication records of the group members. The owner of a user authentication record retains responsibility for the administration of that user.

In this instance, ADMIN.DON retains administrative responsibility for PROG4.SUSAN because he added her user authentication record with the ADD USER command. The PROG4 group is the administrative group for PROG4.SUSAN.

Deleting Groups

The AUTO-DELETE flag in a group definition record indicates whether the group is deleted automatically when its last member is deleted. Administrative groups that are created with the ADD USER command are deleted automatically when the last user is deleted from the group. Groups created with the ADD GROUP command are not deleted automatically. You can have file-sharing groups without members.

To delete a file-sharing group, you must first delete all members from the group and then delete the group itself.

Assume that ADMIN.DON wants to delete the group ProG4. Because no members have been added to the group, he can enter this command directly:

```
=DELETE GROUP ProG4
```

4 Securing Volumes and Devices

The *Safeguard User's Guide* explains how to secure disk files, subvolumes, and processes. This section describes how to secure disk volumes and devices. By default, only super-group members can add volumes and devices to the Safeguard database. (However, you can also define a special group of users to be responsible for volumes and devices. To do so, use the appropriate OBJECTTYPE authorization, as described in Section 5, OBJECTTYPE Control.) This section explains how to secure volumes and devices.

You secure volumes and devices in generally the same manner that you secure other objects. You use the same basic set of nine security commands—ADD, ALTER, DELETE, INFO, SET, RESET, SHOW, FREEZE, and THAW. For example, if you want to add a device to the Safeguard database, use the ADD DEVICE command. Table 4-1 reviews these security commands.

Table 4-1. Security Commands for Volumes and Devices

Command	Description
SET	Establishes default values for the volume or device security attributes. When a volume or device is added to the system, these default values are used for any attributes not specified with the ADD command.
SHOW	Displays the current values of the default security attributes for volumes or devices.
ADD	Adds a volume or device to the Safeguard database by creating an object authorization record for it.
RESET	Resets the value of one or more default security attributes to predefined values.
INFO	Displays the current values of the security attributes defined for a volume or device.
ALTER	Changes one or more security attributes in the authorization record for a volume or device.
FREEZE	Suspends access authority to a volume or device.
THAW	Restores access authority to a frozen volume or device.
DELETE	Removes a volume or device from the Safeguard database by deleting the object authorization record.

The access authorities for volumes are the same as those for disk files and subvolumes. The access authorities for devices are limited to READ, WRITE, and OWNER because other authorities are not meaningful for devices. For your convenience, Table 4-2 on page 4-2 lists the valid access authorities for each type of system object.

You can also use LIKE, DENY, and the minus sign (-) to control attributes of volumes and devices in the same manner you use them with other system objects. And you can specify auditing in the same manner that you would specify it for other objects.

You can transfer ownership of a volume or device by changing the OWNER attribute. You can also designate additional owners by specifying OWNER authority in the access control list. Both forms of ownership allow an owner to modify the authorization record for the volume or device.

Table 4-2. Types of Objects and Valid Access Authorities

Type of Object	Valid Access Authorities					
Disk file	READ	WRITE	EXECUTE	PURGE	CREATE	OWNER
Diskfile-pattern	READ	WRITE	EXECUTE	PURGE	CREATE	OWNER
Volume	READ	WRITE	EXECUTE	PURGE	CREATE	OWNER
Subvolume	READ	WRITE	EXECUTE	PURGE	CREATE	OWNER
Device	READ	WRITE	-	-	-	OWNER
Subdevice	READ	WRITE	-	-	-	OWNER
Process	READ	WRITE	-	PURGE	CREATE	OWNER
Subprocess	READ	WRITE	-	-	-	OWNER

General Procedure for Securing Volumes and Devices

The general procedure for protecting a volume or device with the Safeguard software:

1. Establish default attributes using the SET or RESET commands.
2. Verify the default settings with the SHOW command.
3. Add the volume or device to the Safeguard database with the ADD command. This creates an authorization record for the object.
4. Verify the attributes in the authorization record with the INFO command.
5. Make any necessary changes to the authorization record with the ALTER command.

Considerations for Volumes

By default, only super-group users (255,*) can add a disk volume to the Safeguard database and specify the access authorities for the volume. If necessary, you can transfer ownership to a general user if that individual is to be responsible for protection of the volume.

A disk volume is usually added to the Safeguard database to control who can create files on that volume. By default, anyone can add a subvolume to the Safeguard database.

The valid access authorities for a volume are:

READ	The authority to read disk files on a Safeguard-protected volume
WRITE	The authority to write to disk files on a Safeguard-protected volume
EXECUTE	The authority to execute program files on a Safeguard-protected volume
PURGE	The authority to purge disk files on a Safeguard-protected volume
CREATE	The authority to create disk files on a Safeguard-protected volume
OWNER	The authority to change the authorization record for a Safeguard-protected volume

For example, this command adds an authorization record for the volume $DATA, gives CREATE authority to all members of group number 24, and gives ownership of the VOLUME authorization record to user 24,9:

```
=ADD VOLUME $data, OWNER 24,9, ACCESS 24,* C
```

The Safeguard software always checks volumes for CREATE authority, but it must be configured to check for the other access authorities at the volume and subvolume levels. For more information about configuration, see Configuring Disk-File Control on page 9-18.

Diskfile-pattern authorization records can indirectly secure volumes. Diskfile-patterns that use wild cards in the subvolume and filename elements may be used to determine the entire volume access depending on the CHECK-DISKFILE-PATTERN setting. For example, this command adds an diskfile-pattern authorization record that restricts all diskfile access to volume $DATA to group 24 for READ only:

```
=ADD DISKFILE-PATTERN $data.*.* ACCESS 24,* R
```

Considerations for Devices and Subdevices

By default, only super-group users (255,*) can add devices and subdevices to the Safeguard database. If necessary, ownership can be transferred to another user responsible for protection of that device or subdevice.

Until a device or subdevice is added to the Safeguard database, any process can open that device or subdevice for input or output. After a device or subdevice is under Safeguard control, only processes executing on behalf of users on the access control list can access the device or subdevice.

Valid access authorities for devices and subdevices are:

READ The authority to open a device or subdevice for input

WRITE The authority to open a device or subdevice for output

OWNER The authority to change the authorization record

This command adds an authorization record for the device $LASER and gives READ and WRITE authority to all users who are members of groups 24 and 25:

```
=ADD DEVICE $laser, ACCESS 24,* (R,W); 25,* (R,W)
```

The Safeguard software does not check access control lists for subdevices unless it is configured to check them. (For more information, see Configuring Device Control on page 9-16.)

5 OBJECTTYPE Control

So far, you have seen how to protect an individual object such as a disk volume by creating an authorization record for it. This section describes how to use the OBJECTTYPE commands to control who can create authorization records for objects of a given type.

By default, only super-group users can create authorization records for volumes, devices, and subdevices, but any user can create authorization records for processes, subprocesses, subvolumes, and disk files. The OBJECTTYPE commands allow you to change these restrictions by designating a specific set of users who can add new subjects and objects to the Safeguard database.

With the OBJECTTYPE commands, you can specify:

- Who can protect individual objects of a given type

  Note. Users specified in the OBJECTTYPE can modify ACLs only if they have Owner authority over them. For more information, see Protecting individual objects.

  Note. Starting with H06.26/J06.15 RVUs, the OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME is granted additional access permissions, WRITE (W) and PURGE (P), along with the existing CREATE (C) and OWNER (O) permissions. Members having the WRITE (W) permission on OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME can modify the respective DISKFILE/VOLUME/SUBVOLUME protection records. Members having the PURGE (P) permission on OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME can purge the respective DISKFILE/VOLUME/SUBVOLUME protection records.

- Who can add users, aliases, and groups to the system
- Who can add an OBJECTTYPE record to the Safeguard database
- Who has owner authority of an OBJECTTYPE record
- What auditing is applied to an OBJECTTYPE

Protecting individual objects

The following sample procedure shows how you can modify ACLs if you have Owner authority over them. Assume users E.F, A.B, and USER.USER1.

Note. Starting with H06.26/J06.15 RVUs, the OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME is granted additional access permissions, WRITE (W) and PURGE (P), along with the existing CREATE (C) and OWNER (O) permissions. Members having the WRITE (W) permission on OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME can modify the respective DISKFILE/VOLUME/SUBVOLUME protection records. Members having the PURGE (P) permission on OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME can purge the respective DISKFILE/VOLUME/SUBVOLUME protection records.

1. Log on as SUPER.SUPER.

   Note. SUPER.SUPER is the privileged user authorized to modify ACLs.

2. Create a DISKFILE OBJECTTYPE and set ACL to give C, O authority to user E.F.

   ```
   =ADD OBJECTTYPE DISCFILE, ACCESS E.F(C,O)

   =INFO OBJECTTYPE DISCFILE
   ```

   The display shows:

```
                  LAST-MODIFIED     OWNER             STATUS
DISCFILE
                  18JUN08,  8:25    SUPER.SUPER       THAWED

                  E.F                C,O
```

3. Log on as E.F and alter the DISKFILE OBJECTTYPE to set ACL to give C authority to user A.B.

   Note. User E.F has OWNER(O) authority and hence can set ACL to give C authority to user A.B.

   ```
   = ALTER OBJECTTYPE DISCFILE, ACCESS A.B (C)

   = INFO OBJECTTYPE DISCFILE
   ```

   The display shows:

```
                  LAST-MODIFIED     OWNER             STATUS
DISCFILE
                  18JUN08,  8:56    SUPER.SUPER       THAWED
                  A.B                 C
                  E.F                 C,O
```

4. Log on as A.B. When A.B tries to alter DISKFILE OBJECTTYPE, it results in security violation because A.B does not have OWNER(O) authority.

   ```
   = INFO OBJECTTYPE DISCFILE, DET
   ```

The display shows:

```
               LAST-MODIFIED     OWNER             STATUS
DISCFILE
                 18JUN08,  8:56     SUPER.SUPER       THAWED
                 A.B                    C
                 E.F                   C,O

OBJECT-TEXT-DESCRIPTION =

AUDIT-ACCESS-PASS = NONE        AUDIT-MANAGE-PASS = NONE
AUDIT-ACCESS-FAIL = NONE        AUDIT-MANAGE-FAIL = NONE

=alter objecttype diskfile, access user.user1 o
* ERROR *   RECORD FOR OBJECTTYPE DISCFILE: SECURITY VIOLATION
```

5. Log on as E.F(who has Owner authority) and provide O authority to A.B.

```
= ALTER OBJECTTYPE DISCFILE, ACCESS A.B (O)
= INFO OBJECTTYPE DISCFILE,DET
```

The display shows:

```
              LAST-MODIFIED     OWNER             STATUS
DISCFILE
                 18JUN08,  9:00     SUPER.SUPER       THAWED
               A.B                   C,O
               E.F                   C,O

  OBJECT-TEXT-DESCRIPTION =

  AUDIT-ACCESS-PASS = NONE        AUDIT-MANAGE-PASS = NONE
  AUDIT-ACCESS-FAIL = NONE        AUDIT-MANAGE-FAIL = NONE
```

6. Log on as A.B. You can now alter OBJECTTYPE DISKFILE, which was previously failing due to security violation.

```
= INFO OBJECTTYPE DISCFILE, DET
```

The display shows:

```
              LAST-MODIFIED     OWNER             STATUS
DISCFILE
                 18JUN08,  9:00     SUPER.SUPER       THAWED
               A.B                   C,O
               E.F                   C,O

 OBJECT-TEXT-DESCRIPTION =

 AUDIT-ACCESS-PASS = NONE        AUDIT-MANAGE-PASS = NONE
 AUDIT-ACCESS-FAIL = NONE        AUDIT-MANAGE-FAIL = NONE
```

```
= ALTER OBJECTTYPE DISCFILE, ACCESS USER.USER1 (O)
= INFO OBJECTTYPE DISCFILE, DET
```

The display shows:

```
              LAST-MODIFIED     OWNER             STATUS
DISCFILE
                 18JUN08,  9:01     SUPER.SUPER       THAWED
               USER.USER1             O
               A.B                   C,O
               E.F                   C,O

  OBJECT-TEXT-DESCRIPTION =

  AUDIT-ACCESS-PASS = NONE        AUDIT-MANAGE-PASS = NONE
  AUDIT-ACCESS-FAIL = NONE        AUDIT-MANAGE-FAIL = NONE
```

For the purposes of the OBJECTTYPE commands, the Safeguard software treats users, aliases, and groups as a single object type–OBJECTTYPE USER. Normally, only group managers and the super ID can add users and aliases to the system, and

super-group members can add user groups. However, by creating OBJECTTYPE USER, you can give any designated list of users the authority to add users, aliases, and groups. For more information, see Controlling Users as an Object Type on page 5-8.

An OBJECTTYPE authorization record has these attributes:

```
ACCESS
OWNER
OBJECT-TEXT-DESCRIPTION
AUDIT-ACCESS-PASS
AUDIT-MANAGE-PASS
AUDIT-ACCESS-FAIL
AUDIT-MANAGE-FAIL
```

Note. The OBJECT-TEXT-DESCRIPTION attribute is supported only on systems running J06.05 and later J-series RVUs, H06.16 and later H-series RVUs, and G06.32 and later G-series RVUs. For more information, see the *Safeguard Reference Manual.*

You specify these attributes with the commands listed in Table 5-1.

The OBJECTTYPE commands must be followed by a valid object type. For example, if you want to add an authorization record for the object type VOLUME, use the ADD OBJECTTYPE VOLUME command. The valid object types are:

```
DISKFILE
DISKFILE-PATTERN
SUBVOLUME
VOLUME
DEVICE
SUBDEVICE
PROCESS
SUBPROCESS
OBJECTTYPE
USER
```

Note. OBJECTTYPE USER also controls who can use the ADD ALIAS and ADD GROUP commands.

OBJECTTYPE DISKFILE has no effect on default protection for a user disk files. It only controls who can execute the ADD DISKFILE command.

Initially, only super-group users can create an OBJECTTYPE authorization record. However, you can transfer this authority to designated users with OBJECTTYPE OBJECTTYPE. For more information, see Controlling Who Can Add an Object Type on page 5-9.

Table 5-1. OBJECTTYPE Security Commands

Command	Description
ADD OBJECTTYPE	Creates an OBJECTTYPE authorization record with the specified OBJECTTYPE attribute values. By default, only a local super-group user can add an OBJECTTYPE authorization record.
ALTER OBJECTTYPE	Changes one or more attribute values in an OBJECTTYPE authorization record.
DELETE OBJECTTYPE	Deletes an OBJECTTYPE authorization record.
FREEZE OBJECTTYPE	Suspends access authorities granted to users on the OBJECTTYPE access control list. When an OBJECTTYPE is frozen, only the primary owner, the primary owner's group manager, owners on the access control list, and the local super ID can create individual authorization records for that type of object.
INFO OBJECTTYPE	Displays the existing attribute values in an OBJECTTYPE authorization record.
RESET OBJECTTYPE	Resets one or more default OBJECTTYPE attributes to predefined values.
SET OBJECTTYPE	Establishes default OBJECTTYPE attributes that you specify. Subsequent ADD OBJECTTYPE commands use these defaults for any attributes not specified in the ADD OBJECTTYPE command.
SHOW OBJECTTYPE	Displays the current default values of the OBJECTTYPE attributes.
THAW OBJECTTYPE	Restores a frozen OBJECTTYPE access control list. Users with CREATE authority can once again create individual authorization records for that type of object.

Note. The ASSUME session-control command, described in the *Safeguard User's Guide*, cannot be used with OBJECTTYPE.

An OBJECTTYPE authorization record can have any of the following access authorities:

CREATE The authority to add individual authorization records for that type of object

OWNER The authority to modify the OBJECTTYPE record

Note. Users with CREATE authority on an OBJECTTYPE access control list can add any object of that type regardless of the object's ownership. For example, a user with CREATE authority on OBJECTTYPE DISKFILE can create authorization records for any user's files that are not already protected by the Safeguard software. Normally, users can add only their own files. Therefore, you should not add an object type to the Safeguard database unless you are sure you do not want to use the standard Safeguard restrictions.

Note. Starting with H06.24/J06.13 RVUs, the OBJECTTYPE USER is granted additional access permissions, WRITE (W) and PURGE (P), along with the existing CREATE (C) and OWNER (O) permissions. Members having the WRITE (W) permission on OBJECTTYPE USER can modify any subject records. Members having the PURGE (P) permission on OBJECTTYPE USER can purge any subject records.

Note. Starting with H06.26/J06.15 RVUs, the OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME is granted additional access permissions, WRITE (W) and PURGE (P), along with the existing CREATE (C) and OWNER (O) permissions. Members having the WRITE (W) permission on OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME can modify the respective DISKFILE/VOLUME/SUBVOLUME protection records. Members having the PURGE (P) permission on OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME can purge the respective DISKFILE/VOLUME/SUBVOLUME protection records.

Controlling an Entire Object Type

Normally, only super-group users can add authorization records for volumes, devices, and subdevices. However, all users can add authorization records for disk files that they own as well as authorization records for any subvolumes, processes, or subprocesses.

If you want to change who has authority to add objects of a certain type, add the object type to the Safeguard database. Then create an access control list that gives CREATE authority to specific users.

After you add an object type to the Safeguard database, you can give ownership of the OBJECTTYPE authorization record to someone else by changing the OWNER attribute. Like other objects, OBJECTTYPE authorization records can only be changed by the primary owner, the primary owner's group manager, the super ID, or a user who has owner authority on the access control list.

Note. Starting with H06.24/J06.13 RVUs, the OBJECTTYPE USER is granted additional access permissions, WRITE (W) and PURGE (P), along with the existing CREATE (C) and OWNER (O) permissions. Members having the WRITE (W) permission on OBJECTTYPE USER can modify any subject records. Members having the PURGE (P) permission on OBJECTTYPE USER can purge any subject records.

Note. Starting with H06.26/J06.15 RVUs, the OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME is granted additional access permissions, WRITE (W) and PURGE (P), along with the existing CREATE (C) and OWNER (O) permissions. Members having the WRITE (W) permission on OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME can modify the respective DISKFILE/VOLUME/SUBVOLUME protection records. Members having the PURGE (P) permission on OBJECTTYPE DISKFILE/VOLUME/SUBVOLUME can purge the respective DISKFILE/VOLUME/SUBVOLUME protection records.

The OBJECTTYPE command restricts who can use SAFECOM to create protection records for a given type of object. For example, an OBJECTTYPE DISKFILE authorization record restricts who can use SAFECOM to create disk-file authorization records. However, OBJECTTYPE DISKFILE does not affect any default protection specified for a user's disk files. That is, the Safeguard software automatically creates these protection records regardless of the access control list associated with the OBJECTTYPE DISKFILE authorization record.

The following sample procedure shows how to add an object type to the Safeguard database with a simple access control list. In this case, only group 12 is given authority to add individual device names to the Safeguard database. After the access control list is created, ownership of the authorization record is transferred to user ID 12,8.

1. Create an authorization record for OBJECTTYPE DEVICE with an access control list that grants CREATE authority to all users who have group 12 as their administrative group:

   ```
   =ADD OBJECTTYPE DEVICE, ACCESS 12,* C
   ```

2. Transfer ownership to user ID 12,8:

   ```
   =ALTER OBJECTTYPE DEVICE, OWNER 12,8
   ```

3. Display the authorization record for OBJECTTYPE DEVICE:

   ```
   =INFO OBJECTTYPE DEVICE
   ```

The display shows:

```
                    LAST-MODIFIED      OWNER    STATUS  WARNING-MODE
OBJECTTYPE DEVICE
                  26JAN88, 11:00       12,8    THAWED       OFF

          012,*                 C
```

If you want the same group to control subdevice names, use the LIKE keyword when adding OBJECTTYPE SUBDEVICE, thereby giving its authorization record the same attributes as OBJECTTYPE DEVICE:

```
=ADD OBJECTTYPE SUBDEVICE, LIKE DEVICE
```

Now display the authorization record for OBJECTTYPE SUBDEVICE:

```
=INFO OBJECTTYPE SUBDEVICE
```

```
                    LAST-MODIFIED      OWNER     STATUS      WARNING-MODE
OBJECTTYPE SUBDEVICE
                  26JAN88, 11:10       12,8     THAWED           OFF

          012,*                 C
```

The authorization record has the same attributes as OBJECTTYPE DEVICE.

Now users whose administrative group is group 12 are the only users who can add authorization records for device and subdevice names.

Note. The super ID retains the ability to create protection records for an object type even if you add an OBJECTTYPE protection record for that object type. If you want to deny this authority, you must specifically deny it on the access control list for that object type. The super ID has all access authorities for all system objects unless you specifically deny those authorities on an object's access control list.

Controlling Users as an Object Type

Usually, only the super ID and group managers can add users to the system. If you add OBJECTTYPE USER to the Safeguard database, however, you can create an access control list that specifies who can add users. OBJECTTYPE USER also controls who can add aliases and groups.

To add users, aliases, or groups, a user must have CREATE authority on the access control list for OBJECTTYPE USER.

Note. The OBJECTTYPE USER is granted additional access permissions WRITE(W) and PURGE(P), along with the existing CREATE(C) and OWNER(O) permissions. OBJECTTYPE USER can modify the subject records using the WRITE(W) permission. OBJECTTYPE USER can purge any subject records using the PURGE(P) permission. This is applicable on systems running J06.13 and later J-series RVUs and H06.24 and later H-series RVUs.

To delete users or aliases, a user must have PURGE authority on the access control list for OBJECTTYPE USER. To delete a group, a user must own the individual protection record being deleted.

Suppose you want only group 10 to add users, aliases, and groups. Consider this command:

```
=ADD OBJECTTYPE USER, ACCESS 10,* *, OWNER 10,1
```

This command gives CREATE, OWNER, WRITE and PURGE authority to all users who have group 10 as their administrative group. They can add users by creating user authentication records. Group managers no longer have authority to add users, but the super ID retains this authority. This command also gives user ID 10,1 ownership of the authorization record for OBJECTTYPE USER.

These same users also have the authority to add groups. For security, adding an alias requires additional authority, as described in Assigning an Alias to a User on page 2-40.

To verify the settings of the authorization record for OBJECTTYPE USER, issue the INFO command:

```
=INFO OBJECTTYPE USER
```

The display shows:

```
                    LAST-MODIFIED      OWNER    STATUS   WARNING-MODE
OBJECTTYPE USER
                  27JAN88, 13:30       10,1    THAWED       OFF

          010,*            W,P,C,O
```

Controlling Who Can Add an Object Type

Normally, only super-group users can issue the ADD OBJECTTYPE command. To allow you to grant this authority to other users, the Safeguard software provides a special object type called OBJECTTYPE. Once an OBJECTTYPE OBJECTTYPE authorization record is created, only users with CREATE authority on the access control list for OBJECTTYPE OBJECTTYPE can add OBJECTTYPE authorization records.

This command adds an authorization record for OBJECTTYPE OBJECTTYPE and gives CREATE authority to only two users:

```
=ADD OBJECTTYPE OBJECTTYPE, ACCESS 200,12 C; 200,8 C
```

These commands give ownership of the authorization record to a security administrator (200,1) and deny the super ID all authorities for OBJECTTYPE OBJECTTYPE:

```
=ALTER OBJECTTYPE OBJECTTYPE, ACCESS 255,255 DENY *
=ALTER OBJECTTYPE OBJECTTYPE, OWNER 200,1
```

To verify the settings, use the INFO command:

```
=INFO OBJECTTYPE OBJECTTYPE
```

The display shows:

```
                    LAST-MODIFIED      OWNER    STATUS   WARNING-MODE
OBJECTTYPE OBJECTTYPE
                  27JAN88, 14:10      200,1    THAWED       OFF

          200,8                 C
          200,12                C
          255,255  DENY         C,O
```

OBJECTTYPE Auditing

All OBJECTTYPE authorization records provide auditing attributes. These attributes enable you to audit attempts to add individual authorization records as well as attempts to change the OBJECTTYPE authorization record.

The OBJECTTYPE audit attributes are:

AUDIT-ACCESS-PASS	Successful attempts to add an individual authorization record are audited.
AUDIT-ACCESS-FAIL	Unsuccessful attempts to add an individual authorization record are audited.
AUDIT-MANAGE-PASS	Successful attempts to change the OBJECTTYPE authorization record are audited.
AUDIT-MANAGE-FAIL	Unsuccessful attempts to change the OBJECTTYPE authorization record are audited.

The conditions for the audit attributes can be set to:

ALL	Local and remote attempts are audited.
LOCAL	Only local attempts are audited.
REMOTE	Only remote attempts are audited.
NONE	No attempts are audited.

For example, if you want to audit all successful attempts to change the authorization record for OBJECTTYPE DEVICE, issue this command:

```
=ALTER OBJECTTYPE DEVICE, AUDIT-MANAGE-PASS ALL
```

To check the audit settings, issue the INFO command with the DETAIL option:

```
=INFO OBJECTTYPE DEVICE, DETAIL
```

The display shows:

```
                   LAST-MODIFIED     OWNER    STATUS   WARNING-MODE
OBJECTTYPE DEVICE
                  26JAN88, 11:24      12,8    THAWED       OFF

         012,*                C

OBJECT-TEXT-DESCRIPTION =

AUDIT-ACCESS-PASS = NONE       AUDIT-MANAGE-PASS = ALL
AUDIT-ACCESS-FAIL = NONE         AUDIT-MANAGE-FAIL = NONE
```

AUDIT-MANAGE-PASS is now set to ALL.

For more information about auditing, see the *Safeguard Audit Service Manual*.

6 Managing Security Groups

The Safeguard subsystem allows you to define seven special security groups to control the use of certain restricted commands. The two groups—named SECURITY-ADMINISTRATOR and SYSTEM-OPERATOR—designate who can use the audit service commands, the third group—named SECURITY-OSS-ADMINISTRATOR—designate a list of users who are granted additional OSS security management privileges over the normal users for the operations, acl (ACL_SET), chown(2), chmod(2), chdir(2), and opendir(3),TERMINAL commands, EVENT-EXIT-PROCESS commands, ALTER SAFEGUARD command, and STOP SAFEGUARD command. A fourth group—named SECURITY-PRV-ADMINISTRATOR— designate a list of users or aliases that are granted additional security management privileges over normal users. A fifth group—named SECURITY-AUDITOR—designate a list of users who are not SUPER.SUPER, record owner, or record owner's group manager to view the subject and group records. Users who are part of this group will have read only privileges for the subject and group records. A sixth group, named SECURITY-MEDIA-ADMIN, designates a list of users who are responsible for management of the tape subsystem and have permission to execute the tape management commands. A seventh group named SECURITY-PERSISTENCE-ADMIN designates a list of users who have the same privileges as that of the super-group users for managing persistence processes. Security groups do not exist until you add them to the Safeguard database.

Note. In prior product versions, the Safeguard security groups were managed by GROUP commands. GROUP commands are now used to manage file-sharing groups, as described in Section 3, Managing User Groups. Security groups are now managed with the SECURITY-GROUP commands, as described in this section.

The SECURITY-OSS-ADMINISTRATOR security group is supported only on systems running G06.29 and later G-series RVUs and H06.08 and later H-series RVUs.

The SECURITY-PRV-ADMINISTRATOR group is supported only on systems running J06.11 and later J-series RVUs or H06.22 and later H-series RVUs.

The SECURITY-AUDITOR security group is supported only on systems running on J06.13 and later J-series RVUs, and H06.24 and later H-series RVUs.

The SECURITY-MEDIA-ADMIN security group is supported only on systems running on J06.15 and later J-series RVUs, and H06.26 and later H-series RVUs.

The SECURITY-PERSISTENCE-ADMIN security group is supported only on systems running on J06.16 and later J-series RVUs, and H06.27 and later H-series RVUs.

Note.

1. It is recommended that SUPER.SUPER must not to be added to either SOA/SPA security groups.
2. It is recommended that SOA/SPA security groups be added by any SUPER.* and not by SUPER.SUPER, so that super.super would not gain ownership on the security-groups.
3. SUPER.SUPER can be explicitly denied by using Safeguard ACL's in either SOA/SPA Security groups to prevent its access inadvertently.

   For example: alter sec-group sec-prv-admin,access super.super deny *

Use the ADD SECURITY-GROUP and ALTER SECURITY-GROUP commands to define membership in the security groups. Table 6-1 lists these groups and the functions allowed to their members. For a complete description of the commands used to manage the security groups, see the *Safeguard Reference Manual*.

Table 6-1. Security Groups and Restricted Commands

Command	SECURITY-ADMINISTRATOR	SYSTEM-OPERATOR
ADD AUDIT POOL	Yes	Yes
ALTER AUDIT POOL	Yes	Yes
ALTER AUDIT SERVICE	Yes	No
DELETE AUDIT POOL	Yes	Yes
NEXTFILE	No	Yes
RELEASE	No	Yes
SELECT	Yes	Yes
ADD TERMINAL	Yes	No
ALTER TERMINAL	Yes	No
DELETE TERMINAL	Yes	No
FREEZE TERMINAL	Yes	Yes
THAW TERMINAL	Yes	Yes
ADD EVENT-EXIT-PROCESS	Yes	No
ALTER EVENT-EXIT-PROCESS	Yes	No
DELETE EVENT-EXIT-PROCESS	Yes	No
ALTER SAFEGUARD	Yes	No
STOP SAFEGUARD	Yes	No

Note. Until you add the SECURITY-ADMINISTRATOR and SYSTEM-OPERATOR security groups, any super-group user (user ID 255,n) can use all the commands listed in Table 6-1.

Table 6-2 lists and describes the SECURITY-GROUP security commands.

Table 6-2. SECURITY-GROUP Command Summary

Command	Description
ADD SECURITY-GROUP	Adds a group authorization record with the specified group attribute values. If you do not specify attribute values, the current default is used. By default, only a member of the local super group can add an authorization record for a security group.
ALTER SECURITY-GROUP	Changes one or more attribute values in a group authorization record. For all attributes except ACCESS, the ALTER SECURITY-GROUP command replaces the current value with the specified value. For the ACCESS attribute, ALTER SECURITY-GROUP changes the existing access list to incorporate the access specification.
DELETE SECURITY-GROUP	Deletes a group authorization record. Afterward, only local super-group members can execute the restricted commands.
FREEZE SECURITY-GROUP	Temporarily disables authorities granted to users who have group access. Then, only the owners of a group authorization record, the primary owner's group manager, and the local super ID can execute the restricted commands.
INFO SECURITY-GROUP	Displays the existing attribute values of a group authorization record.
RESET SECURITY-GROUP	Sets one or more default group attribute values to the predefined values of the SET command.
SET SECURITY-GROUP	Sets one or more group attribute values to specified default values.
SHOW SECURITY-GROUP	Displays the current default values of the group attributes.
THAW SECURITY-GROUP	Reenables a frozen group. Then user IDs with EXECUTE authority on the group access list can execute the restricted commands once again.

Valid access authorities for groups are:

EXECUTE Execute the set of commands restricted to the group.

OWNER Manage the group authorization record.

Adding Security Groups

Initially, any super-group member can add the group authorization records for the SECURITY-ADMINISTRATOR, SYSTEM-OPERATOR, SECURITY-OSS-ADMINISTRATOR, SECURITY-PRV-ADMINISTRATOR, SECURITY-AUDITOR, SECURITY-MEDIA-ADMIN, and SECURITY-PERSISTENCE-ADMIN security groups. Once a group authorization record is created for a security group, only users with EXECUTE (E) authority on the access control list can execute the commands restricted to that security group. Only the record owner or users with OWNER (O) authority on the access control list can manage the group authorization record.

For example, assume that, as the local super ID, you initially want to define the SECURITY-ADMINISTRATOR group so that it contains two members—ADMIN.SUE (user ID 200,5) and ADMIN.KEVIN (user ID 200,6)—who will have EXECUTE authority. Use this SAFECOM command:

```
=ADD SECURITY-GROUP SECURITY-ADMINISTRATOR, ACCESS 200,5 E; &
=200,6 E
```

Use the INFO SECURITY-GROUP command to verify the results of the command:

```
=INFO SECURITY-GROUP SECURITY-ADMINISTRATOR
```

The display shows:

```
                    LAST-MODIFIED     OWNER    STATUS
GROUP SECURITY-ADMINISTRATOR
                  26JAN93, 11:12     255,255  THAWED

          200,5                E
          200,6                E
```

Except for the super ID, ADMIN.KEVIN and ADMIN.SUE are now the only users who can execute the restricted commands defined for the SECURITY-ADMINISTRATOR security group.

You also define membership in the SYSTEM-OPERATOR security group by adding an authorization record for that group. For example, this command creates the authorization record for the SYSTEM-OPERATOR security group and gives all authorities to SYSOP.DALE (user ID 255,12):

```
=ADD SECURITY-GROUP SYSTEM-OPERATOR, ACCESS 255,12 *
```

Verify the results of the command:

```
=INFO SECURITY-GROUP SYS-OPER
```

The display shows:

```
                    LAST-MODIFIED     OWNER    STATUS
GROUP SYSTEM-OPERATOR
                  26JAN93, 11:12     255,255  THAWED

          255,12               E,O
```

Except for SYSOP.DALE and the super ID, all super-group members are now prohibited from using the commands reserved for the SYSTEM-OPERATOR security group. Because SYSOP.DALE has both EXECUTE and OWNER authority on the access control list, he can execute these commands and also add other users to the SYSTEM-OPERATOR security group.

You can define membership in the SECURITY-OSS-ADMINISTRATOR security group by adding an authorization record for that group. For example, this command creates the authorization record for the SECURITY-OSS-ADMINISTRATOR security group and gives all authorities to TEST1.USER1 (204,001), TEST2.USER2 (240,002), TEST3.USER3 (240,003), and TEST4.USER4 (240,004):

```
=ADD SECURITY-GROUP SECURITY-OSS-ADMINISTRATOR, &
OWNER SUPER.TEST, OBJECT-TEXT-DESCRIPTION ''Record Created'',&
AUDIT-ACCESS NONE, &
AUDIT-MANAGE-PASS ALL, &
ACCESS TEST1.USER1 (E,O); TEST1.USER2 (E); TEST1.USER3(O)
```

Verify the results of the command:

```
=INFO SECURITY-GROUP SECURITY-OSS-ADMINISTRATOR
```

The display shows:

```
                  LAST-MODIFIED      OWNER    STATUS
SECURITY-OSS-ADMINISTRATOR
                 24MAY06,   1:29     255,5    THAWED

               240,001           E       O
               240,002           E
               240,003                    O

OBJECT-TEXT-DESCRIPTION = ''Record Created''

AUDIT-ACCESS-PASS = NONE       AUDIT-MANAGE-PASS = ALL
AUDIT-ACCESS-FAIL = NONE
```

Note. The OBJECT-TEXT-DESCRIPTION attribute is supported only on systems running J06.05 and later J-series RVUs, H06.16 and later H-series RVUs, and G06.32 and later G-series RVUs. For more information about OBJECT-TEXT-DESCRIPTION attribute, see the *Safeguard Reference Manual*.

You can define membership in the SECURITY-PRV-ADMINISTRATOR security group by adding an authorization record for that group. For example, this command creates the authorization record for the SECURITY-PRV-ADMINISTRATOR security group:

```
= ADD SECURITY-GROUP SECURITY-PRV-ADMINISTRATOR, ACCESS SECGRP.*
*
```

To verify the results of the command:

```
=INFO SECURITY-GROUP SECURITY-PRV-ADMINISTRATOR
```

The display shows:

```
                    LAST-MODIFIED     OWNER             STATUS
SECURITY-PRV-ADMINISTRATOR
                    1MAY10, 13:20     255,255           THAWED

 GROUP       SECGRP                                       E,O

=
```

You can define membership in the SECURITY-AUDITOR security group by adding an authorization record for that group. For example, the following command creates the authorization record for the SECURITY-AUDITOR security group:

```
= ADD SECURITY-GROUP SECURITY-AUDITOR, ACCESS SECGRP.* *
```

To verify the results of the command:

```
=INFO SECURITY-GROUP SECURITY-AUDITOR
```

The display shows:

```
                    LAST-MODIFIED     OWNER             STATUS
SECURITY-AUDITOR
                    1MAY10, 13:20     255,255           THAWED

 GROUP       SECGRP                                       E,O

=
```

You can define membership in the SECURITY-MEDIA-ADMIN security group by adding an authorization record for that group. For example, the following command creates the authorization record for the SECURITY-MEDIA-ADMIN security group:

```
= ADD SECURITY-GROUP SECURITY-MEDIA-ADMIN, OWNER SUPER.SUPER,
ACCESS 255,* *
```

To verify the results of the command:

```
=INFO SECURITY-GROUP SECURITY-MEDIA-ADMIN
```

```
                    LAST-MODIFIED     OWNER             STATUS
SECURITY-MEDIA-ADMIN
                    13JUL12, 14:30    255,255           THAWED

 GROUP       00255                                        E,O

=
```

You can define membership in the SECURITY-PERSISTENCE-ADMIN security group by adding an authorization record for that group. For example, the following command creates the authorization record for the SECURITY-PERSISTENCE-ADMIN security group:

```
= ADD SECURITY-GROUP SECURITY-PERSISTENCE-ADMIN, OWNER
SUPER.SUPER,ACCESS 255,* *
```

To verify the results of the command:

```
=INFO SECURITY-GROUP SECURITY-PERSISTENCE-ADMIN
```

```
                    LAST-MODIFIED       OWNER    STATUS
SECURITY-PERSISTENCE-ADMIN
                  18MAR13, 18:23     255,255    THAWED

 GROUP         00255                             E,   O
```

Transferring Security Group Ownership

You can transfer ownership of a group authorization record to another user. For example, this command gives ownership of the SECURITY-ADMINISTRATOR authorization record to ADMIN.BOB (200,8):

```
=ALTER SECURITY-GROUP SEC-ADM, OWNER admin.bob
```

You can abbreviate the security group name as SEC-ADM. To verify the results:

```
=INFO SECURITY-GROUP SEC-ADM
```

The display shows:

```
                    LAST-MODIFIED     OWNER    STATUS
GROUP SECURITY-ADMINISTRATOR
                  26JAN93, 11:17     200,8    THAWED

         200,5                E
         200,6                E
```

ADMIN.BOB is now responsible for managing membership in the SECURITY-ADMINISTRATOR security group.

You can transfer ownership of a group authorization record to another user by the ALTER command. For example, this command gives ownership of the SECURITY-OSS-ADMINISTRATOR authorization record to another user:

```
=ALTER SECURITY-GROUP SECURITY-OSS-ADMINISTRATOR, &
ACCESS TEST1.USER1 - (E) ; TEST1.USER4 (E, O), &
AUDIT-MANAGE ALL
```

To verify the results:

```
=INFO SECURITY-GROUP SECURITY-OSS-ADMINISTRATOR
```

The display shows:

```
                      LAST-MODIFIED       OWNER        STATUS
SECURITY-OSS-ADMINISTRATOR
                      24MAY06,  1:29      255,5        THAWED
                      240,001                O
                      240,002             E
                      240,003                O
                      240,004             E,   O
```

You can transfer ownership of a group authorization record to another user using the ALTER command. For example, this command gives ownership of the SECURITY-PRV-ADMINISTRATOR authorization record to another user:

```
=ALTER SECURITY-GROUP SECURITY-PRV-ADMINISTRATOR, &
ACCESS TEST1.USER1  (E,O) ; TEST1.USER4 (E), &
AUDIT-MANAGE ALL
```

To verify the results:

```
=INFO SECURITY-GROUP SECURITY-PRV-ADMINISTRATOR
```

The display shows:

```
                     LAST-MODIFIED          OWNER        STATUS
SECURITY-PRV-ADMINISTRATOR
                 11DEC10,  17:00            255,5        THAWED
                 20,001                     E,  O
                 220,004                    E
```

You can enable auditing of pass or fail protection record management operations using the ALTER command. For example, this command alters the SECURITY-AUDITOR security group to enable auditing of pass or fail protection record management operations:

```
=ALTER SECURITY-GROUP SECURITY-AUDITOR, AUDIT-MANAGE ALL
```

You can transfer ownership of a group authorization record to another user by using the ALTER command. For example, the following command gives ownership of the SECURITY-MEDIA-ADMIN authorization record to another user:

```
=ALTER SECURITY-GROUP SECURITY-MEDIA-ADMIN, ACCESS TEST.MGR
(E,O)
```

To verify the results, use the following command:

```
=INFO SECURITY-GROUP SECURITY-MEDIA-ADMIN
```

The display shows:

```
                     LAST-MODIFIED            OWNER          STATUS
SECURITY-MEDIA-ADMIN
                     13JUL12,   14:42         255,255        THAWED
                     005,255                  E,  O
GROUP                00255                    E,  O
```

You can transfer ownership of a group authorization record to another user by using the ALTER command. For example, the following command provides the ownership of SECURITY-PERSISTENCE-ADMIN authorization record to another user:

```
=ALTER SECURITY-GROUP SECURITY-PERSISTENCE-ADMIN, ACCESS
TEST.USER1 (E,O)
```

To verify the results, use the following command:

```
=INFO SECURITY-GROUP SECURITY-PERSISTENCE-ADMIN
```

The display shows:

```
                        LAST-MODIFIED              OWNER             STATUS
SECURITY-PERSISTENCE-ADMIN
                        18MAR13, 18:23             255,255           THAWED
                        022,001                    E,   O
GROUP                   00255                      E,   O
```

Freezing and Thawing Security Groups

A security group can be frozen by the primary owner or by any user with OWNER authority on the access control list for the group. When a group is frozen, the only individuals who can execute the commands restricted to that group are the primary owner, the primary owner's group manager, owners on the access control list, and the local super ID.

For example, ADMIN.BOB can use this command to suspend the EXECUTE authority granted to ADMIN.KEVIN and ADMIN.SUE:

```
=FREEZE SECURITY-GROUP SEC-ADM
```

ADMIN.BOB then uses the INFO SECURITY-GROUP command to verify the results:

```
=INFO SECURITY-GROUP SEC-ADM
```

The display shows:

```
                    LAST-MODIFIED      OWNER     STATUS
GROUP SECURITY-ADMINISTRATOR
                  26JAN93, 11:26      200,8     FROZEN

          200,5                 E
          200,6                 E
```

Until ADMIN.BOB thaws the group authorization record, only he, his group manager, and the local super ID can execute the commands restricted to the SECURITY-ADMINISTRATOR group.

To thaw the group, ADMIN.BOB issues this command:

```
=THAW SECURITY-GROUP SECURITY-ADMINISTRATOR
```

The SECURITY-OSS-ADMINISTRATOR security group can be frozen by the primary owner or by any user with OWNER authority on the access control list for the group. For example,

```
=FREEZE SECURITY-GROUP SECURITY-OSS-ADMINISTRATOR
```

To verify the results:

```
=INFO SECURITY-GROUP SECURITY-OSS-ADMINISTRATOR
```

The display shows:

```
                    LAST-MODIFIED       OWNER        STATUS
SECURITY-OSS-ADMINISTRATOR
                    24MAY06,  1:30      255,5       FROZEN

                    240,001                  O
                    240,002             E
                    240,003                  O
                    240,004             E,   O
```

To thaw the group:

```
= THAW SECURITY-GROUP SECURITY-OSS-ADMINISTRATOR
```

To verify the results:

```
= INFO SECURITY-GROUP SECURITY-OSS-ADMINISTRATOR
```

The display shows:

```
                    LAST-MODIFIED       OWNER        STATUS
SECURITY-OSS-ADMINISTRATOR
                    24MAY06,  1:31      255,5        THAWED

                     240,001                O
                     240,002            E
                     240,003                O
                     240,004            E,  O
```

The SECURITY-PRV-ADMINISTRATOR security group can be frozen by the primary owner or by any user with OWNER authority on the access control list for the group. For example,

```
=FREEZE SECURITY-GROUP SECURITY-PRV-ADMINISTRATOR
```

To verify the results:

```
=INFO SECURITY-GROUP SECURITY-PRV-ADMINISTRATOR
```

The display shows:

```
                       LAST-MODIFIED         OWNER        STATUS
SECURITY-PRV-ADMINISTRATOR
                  11DEC10,  17:01            255,5        FROZEN

                    220,001                  E, O
                    220,004                  E
```

To thaw the group:

```
= THAW SECURITY-GROUP SECURITY-PRV-ADMINISTRATOR
```

To verify the results:

```
= INFO SECURITY-GROUP SECURITY-PRV-ADMINISTRATOR
```

The display shows:

```
                       LAST-MODIFIED         OWNER        STATUS
SECURITY-PRV-ADMINISTRATOR
                    11DEC10,  17:02          255,5        THAWED

                    220,001                  E,  O
                    220,004                  E
```

The SECURITY-AUDITOR security group can be frozen using the following command:

```
=FREEZE SECURITY-GROUP SECURITY-AUDITOR
```

To verify the results:

```
=INFO SECURITY-GROUP SECURITY-AUDITOR
```

The display shows:

```
                        LAST-MODIFIED     OWNER    STATUS
SECURITY-AUDITOR
                  11DEC10,        17:01   204,1    FROZEN

   GROUP           00144           E
```

To thaw the group:

```
= THAW SECURITY-GROUP SECURITY-AUDITOR
```

To verify the results:

```
= INFO SECURITY-GROUP SECURITY-AUDITOR
```

The display shows:

```
                  LAST-MODIFIED     OWNER    STATUS
SECURITY-AUDITOR
                  11DEC10, 17:02    204,1    THAWED

   GROUP           00144             E
```

The SECURITY-MEDIA-ADMIN security group can be frozen using the following command:

```
=FREEZE SECURITY-GROUP SECURITY-MEDIA-ADMIN
```

To verify the results, use the following command:

```
=INFO SECURITY-GROUP SECURITY-MEDIA-ADMIN
```

The display shows:

```
                    LAST-MODIFIED           OWNER          STATUS
SECURITY-MEDIA-ADMIN
                   13JUL12,   14:43        255,255        FROZEN

                    005,255                 E,  O
GROUP               00255                   E,  O
```

To thaw the group, use the following command:

```
= THAW SECURITY-GROUP SECURITY-MEDIA-ADMIN
```

To verify the results, use the following command:

```
= INFO SECURITY-GROUP SECURITY-MEDIA-ADMIN
```

The display shows:

```
                    LAST-MODIFIED           OWNER          STATUS
SECURITY-MEDIA-ADMIN
                    13JUL12,   14:44        255,255       THAWED

                     005,255                E,  O
GROUP                00255                  E,  O
```

The SECURITY-PERSISTENCE-ADMIN security group can be frozen using the following command:

```
=FREEZE SECURITY-GROUP SECURITY-PERSISTENCE-ADMIN
```

To verify the results, use the following command:

```
=INFO SECURITY-GROUP SECURITY-PERSISTENCE-ADMIN
```

The display shows:

```
                       LAST-MODIFIED      OWNER            STATUS
SECURITY-PERSISTENCE-ADMIN
                     18MAR13, 18:24       255,255         FROZEN

                       022,001            E,  O
GROUP                  00255              E,  O
```

To thaw the group, use the following command:

```
= THAW SECURITY-GROUP SECURITY-PERSISTENCE-ADMIN
```

To verify the results, use the following command:

```
= INFO SECURITY-GROUP SECURITY-PERSISTENCE-ADMIN
```

The display shows:

```
                       LAST-MODIFIED      OWNER            STATUS
SECURITY-PERSISTENCE-ADMIN
                     18MAR13, 18:25      255,255          THAWED

                       022,001            E,  O
GROUP                  00255              E,  O
```

Deleting Security Groups and Group Members

You delete a member from a security group in the same way that you remove users from access control lists for other objects. To delete a user from a security group, remove the access authorities granted to that user.

For example, this command removes ADMIN.KEVIN (user ID 200,6) from the SECURITY-ADMINISTRATOR group:

```
=ALTER SECURITY-GROUP SEC-ADM, ACCESS 200,6 - *
```

If you want to delete an entire security group, use the DELETE SECURITY-GROUP command. For example, this command deletes the SYSTEM-OPERATOR security group:

```
=DELETE SECURITY-GROUP SYS-OPER
```

To delete an entire SECURITY-OSS-ADMINISTRATOR security group, use the DELETE SECURITY-GROUP command. For example, the following command deletes the SECURITY-OSS-ADMINISTRATOR security group:

```
=DELETE SECURITY-GROUP SECURITY-OSS-ADMINISTRATOR
```

To delete an entire SECURITY-PRV-ADMINISTRATOR security group, use the DELETE SECURITY-GROUP command. For example, the following command deletes the SECURITY-PRV-ADMINISTRATOR security group:

```
=DELETE SECURITY-GROUP SECURITY-PRV-ADMINISTRATOR
```

To delete an entire SECURITY-AUDITOR security group, use the DELETE SECURITY-GROUP command. For example, the following command deletes the SECURITY-AUDITOR security group:

```
=DELETE SECURITY-GROUP SECURITY-AUDITOR
```

To delete an entire SECURITY-MEDIA-ADMIN security group, use the DELETE SECURITY-GROUP command. For example, the following command deletes the SECURITY-MEDIA-ADMIN security group:

```
=DELETE SECURITY-GROUP SECURITY-MEDIA-ADMIN
```

To delete the SECURITY-PERSISTENCE-ADMIN security group, use the DELETE SECURITY-GROUP command. For example, the following command deletes the SECURITY-PERSISTENCE-ADMIN security group:

```
=DELETE SECURITY-GROUP SECURITY-PERSISTENCE-ADMIN
```

7 Securing Terminals

This section explains how to add a terminal definition to the Safeguard database so that the Safeguard software controls that terminal. When a terminal definition is added, the Safeguard software can perform the following additional security functions at the terminal:

- Start a specific command interpreter automatically after the user is authenticated
- Allow the user who is logged on at the terminal to have exclusive access to it

Terminal definitions can be added selectively for some or all the terminals on your system. The Safeguard user authentication controls are enforced regardless of whether or not the terminal is controlled by the Safeguard software.

Use the TERMINAL security commands to add and manage terminal definition records. Except for the INFO TERMINAL command, use of the TERMINAL commands is restricted to security group members. INFO TERMINAL can be executed by any user. Table 7-1 lists the security groups and shows which TERMINAL commands can be executed by group members. If you have not defined the SECURITY-ADMINISTRATOR and SYSTEM-OPERATOR groups, any super-group member can use the TERMINAL commands. (For more information about how to define security groups, see Section 6, Managing Security Groups.)

Table 7-1. Security Groups and TERMINAL Commands

Command	SECURITY-ADMINISTRATOR	SYSTEM-OPERATOR
ADD TERMINAL	Yes	No
ALTER TERMINAL	Yes	No
DELETE TERMINAL	Yes	No
FREEZE TERMINAL	Yes	Yes
INFO TERMINAL	Yes	Yes
THAW TERMINAL	Yes	Yes

Unlike SAFECOM commands for other objects, the TERMINAL commands do not allow you to specify an access control list. If you want to specify an access control list for a terminal, you must do so with DEVICE or SUBDEVICE commands, depending on how terminals are named on your system.

Terminals controlled by the Safeguard software can be configured for exclusive access. This feature ensures that any user who is logged on at a Safeguard terminal is given exclusive access to that terminal until the user logs off. All other users are denied access to the terminal during the authenticated user session. For more information on the TERMINAL-EXCLUSIVE-ACCESS attribute, see Configuring Exclusive Access at Safeguard Terminals on page 9-36.

Table 7-2 on page 7-2 summarizes the TERMINAL commands.

Table 7-2. TERMINAL Command Summary

Command	Description
ADD TERMINAL	Adds a terminal definition record with the specified terminal attribute values.
ALTER TERMINAL	Changes one or more attribute values in a terminal definition record.
DELETE TERMINAL	Deletes a terminal definition record.
FREEZE TERMINAL	Disables a terminal from accepting the LOGON command.
INFO TERMINAL	Displays the existing attribute values in a terminal definition record.
THAW TERMINAL	Reenables a frozen terminal so that it accepts the LOGON command.

Control of the Logon Dialog

When you add a terminal definition record, the Safeguard software takes over control of the logon dialog at that terminal. In earlier RVUs, certain extended features, such as warning of a pending password expiration, were available only at a Safeguard terminal. Effective with D30 product versions, the TACL command interpreter also provides these logon features when Safeguard is running on the system.

From the user perspective, the Safeguard logon dialog is the same as the TACL logon dialog. As long as Safeguard is running on your system, all security controls and logon features, such as the password expiry grace period, are enforced whether or not the terminal is defined as a Safeguard terminal.

For more information about the logon dialog at a Safeguard terminal, see the *Safeguard User's Guide*.

Starting a Command Interpreter

When you add a terminal definition record, you can specify that a particular command interpreter be started automatically at that terminal after user authentication. A command interpreter can also be specified in a user authentication record and in the Safeguard configuration record.

The Safeguard software resolves any conflicts among these records by searching for a command interpreter specification in the following order: user record, terminal record, Safeguard configuration record. The first specification found during the search is the command interpreter that is started after user authentication. Therefore, a command interpreter specified in a user authentication record always takes precedence over one specified in a terminal record or Safeguard configuration record.

If no command interpreter is specified in the user authentication record or in the terminal definition record, the command interpreter defined in the Safeguard configuration record is used. Unless you use the ALTER SAFEGUARD command to

change the configuration record, the command interpreter defined in that record is $SYSTEM.SYSTEM.TACL.

The Safeguard software can honor the command interpreter specification only at a terminal that it controls. If the Safeguard software does not control the logon dialog at a terminal, all command interpreter specifications are ignored at that terminal.

Adding a Terminal Definition

Before you add a terminal definition to the Safeguard database, be sure to stop any process running at that terminal. If you do not stop the other process, the Safeguard software competes with it for control of the terminal.

In the most basic form, you can use the ADD TERMINAL command without any additional parameters to add a terminal definition. Simply specify the name of the terminal to be added. As is true when you add most other object names, you cannot use wild-card characters when you specify a terminal name in an ADD command.

This command adds a terminal definition for the terminal $TFOX.#T014:

```
=ADD TERMINAL $tfox.#t014
```

This form of the command uses default values for the command interpreter to be started at the terminal after a user is authenticated.

Note. When you add a terminal definition record to the Safeguard database, that terminal becomes frozen. You must use the THAW TERMINAL command to enable the terminal.

To thaw terminal $TFOX.#T014, issue this command:

```
=THAW TERMINAL $tfox.#t014
```

To verify the results of the commands, use this INFO TERMINAL command:

```
=INFO TERMINAL $tfox.#t014
```

The display shows:

```
TERMINAL  $TFOX.#T014                    STATUS  THAWED

 PROG  = * NONE *
 LIB   = * NONE *
 PNAME = * NONE *
 SWAP  = * NONE *
 CPU   = * NONE *
 PRI   = * NONE *

 PARAM-TEXT =
```

The display shows that the terminal is thawed and no command interpreter has been defined for the terminal.

You can specify a particular command interpreter to be started automatically after user authentication. To do so, you must specify the name of the object file of the command interpreter to be invoked. You can also specify several optional parameters, such as

the priority at which the command interpreter is to execute. For more information about these parameters, see the *Safeguard Reference Manual*.

For example, this command adds terminal $TFOX.#T015 and causes TACL to be started after user authentication at the terminal:

```
=ADD TERMINAL $tfox.#t015, PROG $system.system.tacl
```

Use the INFO TERMINAL command to verify the results:

```
=INFO TERMINAL $tfox.#t015
```

The display shows:

```
TERMINAL  $TFOX.#T015                       STATUS  FROZEN

 PROG  = $SYSTEM.SYSTEM.TACL
 LIB   = * NONE *
 PNAME = * NONE *
 SWAP  = * NONE *
 CPU   = * NONE *
 PRI   = * NONE *

 PARAM-TEXT =
```

The display shows that TACL is the command interpreter defined for this terminal. The terminal is frozen because the terminal definition was just added to the Safeguard database. To thaw the terminal:

```
THAW TERMINAL $tfox.#t015
```

Altering a Terminal Definition

Like other Safeguard protection records, an existing terminal definition record can be altered.

For example, suppose you want to change the definition for terminal $TFOX.#T015 to tailor the execution parameters for TACL. This command specifies that TACL is to execute with a priority of 150 in CPU 4 and that $DATA2 is to be used as the swap volume. The command also includes parameter text of 5, which TACL interprets as the backup CPU number.

```
=ALTER TERMINAL $tfox.#t015, PRI 150, CPU 4, &
=SWAP $data2, PARAM-TEXT 5
```

Use the INFO TERMINAL command to verify the results:

```
=INFO TERMINAL $tfox.#t015
```

The display shows:

```
TERMINAL  $TFOX.#T015                      STATUS  THAWED

 PROG  = $SYSTEM.SYSTEM.TACL
 LIB   = * NONE *
 PNAME = * NONE *
 SWAP  = $DATA2
 CPU   = 4
 PRI   = 150

 PARAM-TEXT =5
```

Freezing and Thawing a Terminal

When you freeze a Safeguard terminal, all logon attempts at that terminal are disallowed. For example, this command freezes the terminal $TFOX.#T014:

```
FREEZE TERM $tfox.#t014
```

To reenable users to log on at the terminal:

```
THAW TERM $tfox.t014
```

The FREEZE and THAW commands are valid only for a Safeguard terminal. If no terminal definition exists for a terminal, FREEZE and THAW have no effect on that terminal.

Deleting a Terminal Definition

You can delete a terminal definition so that the terminal is no longer controlled by the Safeguard software. After you delete the definition, you must start another command interpreter, such as TACL, to control the logon dialog at that terminal.

For example, this command deletes the definition for the terminal $TFOX.#T015:

```
=DELETE TERMINAL $tfox.#t015
```

This terminal is not usable until you start another command interpreter to handle its logon dialog.

8 Warning Mode

Warning mode is a special state that allows you to test the reliability and effectiveness of Safeguard protection on your system. In warning mode, the Safeguard software allows access to objects that have a protection record even if the protection record does not grant access. The Safeguard software audits any access attempt that would normally have been denied.

Objects that are not protected by the Safeguard software are unaffected in warning mode. For example, if a disk file does not have a Safeguard protection record and ACL-REQUIRED-DISKFILE is OFF, access to that disk file is unaffected by warning mode. In addition, in warning mode, the Safeguard software does not consider object types and security groups to be objects although they might have access control lists. Object types and security groups are unaffected in warning mode.

Safeguard provides two levels of warning-mode:

- System level (System-Warning-Mode), where all objects protected by Safeguard are evaluated in warning mode
- Object level (Object-Warning-Mode), where only selected objects protected by Safeguard are evaluated in warning mode

Note. Because warning mode causes the Safeguard software to grant access that it would otherwise deny, use caution in setting the ACL-REQUIRED Safeguard configuration attributes. With these attributes set, access is normally denied for any object that does not have an access control list. In warning mode, however, access to all such objects is granted.

Warning mode is intended to be used as a debugging tool to help you check changes to your security policy. By examining the audit records generated in warning mode, you can check the effectiveness of your access control lists. When the access control lists are implemented satisfactorily, you can turn off warning mode to return to a production environment in which all Safeguard controls are enforced.

Table 8-1 shows how the Safeguard software handles access requests for most objects protected by access control lists in standard mode and in warning mode. However, special considerations can apply to disk files and processes, as described later in this section.

Table 8-1. Warning Mode Rulings on Object ACLs

Safeguard ACL Ruling	Guardian Security	Access Result	Audit Record Generated	Outcome in Audit Record
Standard Mode				
Grants	N.A.	Yes	As specified	Granted
Denies	N.A.	No	As specified	Denied
Warning Mode				
Grants	N.A.	Yes	As specified	Granted
Denies	N.A.	Yes*	Always*	Warning*

* Indicates that access result is due to warning mode evaluation of the access control list.

Considerations for Disk Files and Processes

Because disk files and processes have Guardian security associated with them, special circumstances can apply in warning mode when Safeguard protection is bypassed. For these two types of objects, you can specify that warning mode be run with a fallback option. The fallback option is controlled by a Safeguard global configuration attribute that can be set to either GUARDIAN or GRANT.

The GUARDIAN setting invokes Guardian security rulings when Safeguard protection that denies access has been bypassed in warning mode. The fallback option allows you to test the Safeguard security settings while maintaining Guardian protection.

With the fallback option set to GRANT, the Safeguard software ignores the Guardian security settings and grants the access that it would otherwise deny. This can be useful, for example, if you know that Guardian security has not been kept current with your security policy. This method of operation can also be useful in certain emergency situations when routine security measures need to be suspended.

Diskfile patterns can be used to reduce administrative burden by supplying one pattern that matches many subvolumes or filenames. For more information on diskfile patterns, see the *Safeguard User's Guide*.

Disk-File Security

In warning mode with the fallback option set to GUARDIAN, the Safeguard software treats disk-file access attempts in the following manner. If the disk file's access control list does not permit the access attempt, the Safeguard software checks the Guardian disk-file security string before granting the attempt. If the Guardian security string grants the access, the Safeguard software allows the access and writes an audit record with the outcome WARNING. If the security string does not grant the access, the Safeguard software denies the access attempt. No audit record is written in this instance unless auditing is specified for the disk file.

Table 8-2 on page 8-3 shows how the Safeguard software handles disk files in standard mode and in warning mode with the fallback option set to GUARDIAN and GRANT. The fallback option is meaningful only when a Safeguard protection record exists.

Table 8-2. Warning Mode Rulings on Disk-File ACLs

Safeguard ACL Ruling	Guardian Security	Access Result	Audit Record Generated	Outcome in Audit Record
Standard Mode				
Grants	N.A.	Yes	As specified	Granted
Denies	N.A.	No	As specified	Denied
No record	Use Guardian	Yes/No~	No	N.A.
Warning Mode Fallback Guardian				
Grants	N.A.	Yes	As specified	Granted
Denies	Grants	Yes*	Always	Warning*
Denies	Denies	No*	As specified	Denied
No record	Use Guardian	Yes/No~	No	N.A.
Warning Mode Fallback Grant				
Grants	N.A.	Yes	As specified	Granted
Denies	N.A.	Yes*	Always*	Warning*
No record	Use Guardian	Yes/No~	No	N.A.

* Indicates that access result is due to warning mode evaluation of the access control list.

~ Indicates that access result is determined by Guardian security string.

Process Stop Mode Security

Although processes do not have preexisting Guardian security, they do have stop modes, which influence whether or not a task can be stopped by another process. Definitions of the three stop modes follow:

- Mode 0 indicates that this process can be stopped by any other process.
- Mode 1 indicates that this process can be stopped only by the super ID, a process whose PAID is the same as this process's PAID or CAID or, a process whose PAID is the same the PAID or CAID of the group manager.
- Mode 2 indicates that this process cannot be stopped by any other process.

Table 8-3 on page 8-4 shows how the Safeguard software handles process stop attempts in standard mode and in warning mode with the fallback option set to GUARDIAN and GRANT.

If a process has stop mode 2 and the access attempt is granted, the Safeguard software writes an audit record with the outcome of either WARNING or GRANTED. However, the process is not actually stopped because the Guardian stop mode of 2 always takes precedence over the Safeguard ruling.

As Table 8-3 on page 8-4 shows, the single difference between the GUARDIAN and GRANT settings of the fallback option is that the Safeguard software adheres to Guardian rules for a process in stop mode 1 when the fallback is GUARDIAN.

For more information about Guardian stop modes, see the SETSTOP procedure in the *Guardian Procedure Calls Reference Manual*.

Table 8-3. Warning Mode Rulings on Process ACLs

Process Stop / Safeguard ACL Ruling	Guardian Security	Access Result	Audit Record Generated	Outcome in Audit Record
Standard Mode				
Grants	Mode 0, 1	Yes	As specified	Granted
	Mode 2	No**	As specified	Granted
Denies	Mode 0, 1	No%	As specified	Denied%
	Mode 2	No**%	As specified	Denied%
No record	Use Guardian	Yes/No	No	N.A.
Warning Mode Fallback Guardian				
Grants	Mode 0,1	Yes	As specified	Granted
	Mode 2	No**	As specified	Granted
Denies	Mode 0,1	Yes*#%	Always*	Warning*%
	Mode 2	No**%	Always*	Warning*%
No record	Use Guardian	Yes/No	No	N.A.
Warning Mode Fallback Grant				
Grants	Mode 0, 1	Yes	As specified	Granted
	Mode 2	No**	As specified	Granted
Denies	Mode 0, 1	Yes*%	Always*	Warning*%
	Mode 2	No**%	Always*	Warning*%
No record	Use Guardian	Yes/No	No	N.A.

* Indicates that access result is due to warning mode evaluation of the access control list.

** Attempts to stop a process at stop mode 2 proceed without a security violation message, but does not succeed in stopping the process until the process sets itself to a lower stop mode. These requests are pending and are audited as GRANTED or WARNING.

\# Guardian rules are enforced for processes at stop mode 1. For more information, see the *Guardian Procedure Calls Reference Manual*.

% Indicates if the requester is the one who started the process, the outcome will be GRANTED.

Using Warning Mode

Warning mode puts your system into a special state in which Safeguard security is bypassed. To invoke warning mode, use the ALTER SAFEGUARD command to set the SYSTEM-WARNING-MODE global configuration attribute to ON:

```
=ALTER SAFEGUARD, SYSTEM-WARNING-MODE ON
```

If you want to run warning mode with the Guardian fallback option disabled, you must also set the WARNING-FALLBACK-SECURITY attribute to GRANT:

```
=ALTER SAFEGUARD, WARNING-FALLBACK-SECURITY GRANT
```

To verify the results of the commands:

```
=INFO SAFEGUARD
```

The display shows:

```
AUTHENTICATE-MAXIMUM-ATTEMPTS =      3
AUTHENTICATE-FAIL-TIMEOUT     =     60 SECONDS
AUTHENTICATE-FAIL-FREEZE      = OFF

PASSWORD-REQUIRED = OFF         PASSWORD-HISTORY =  0
PASSWORD-ENCRYPT  = ON          PASSWORD-MINIMUM-LENGTH =  0

PASSWORD-MAXIMUM-LENGTH =  8
PASSWORD-ALGORITHM = DES
PASSWORD-COMPATIBILITY-MODE = ON
PASSWORD-UPPERCASE-REQUIRED = OFF
PASSWORD-LOWERCASE-REQUIRED = OFF
PASSWORD-NUMERIC-REQUIRED = OFF
PASSWORD-SPECIALCHAR-REQUIRED = OFF
PASSWORD-SPACES-ALLOWED = OFF
PASSWORD-ALPHA-REQUIRED = OFF
PASSWORD-MIN-QUALITY-REQUIRED = 0
PASSWORD-MIN-UPPERCASE-REQ = 0
PASSWORD-MIN-LOWERCASE-REQ = 0
PASSWORD-MIN-NUMERIC-REQ = 0
PASSWORD-MIN-SPECIALCHAR-REQ = 0
PASSWORD-MIN-ALPHA-REQ = 0
PASSWORD-ERROR-DETAIL= OFF
PASSWORD-MAY-CHANGE   =       0 DAYS BEFORE-EXPIRATION
PASSWORD-EXPIRY-GRACE =       0 DAYS-AFTER-EXPIRATION

SYSTEM-WARNING-MODE = OFF      WARNING-FALLBACK-SECURITY = GUARDIAN
OBJECT-WARNING-MODE = OFF

ALLOW-NODE-ID-ACL       = OFF
DIRECTION-DEVICE      = DEVICE-FIRST        CHECK-DEVICE     = ON
COMBINATION-DEVICE    = FIRST-ACL           CHECK-SUBDEVICE  = OFF
ACL-REQUIRED-DEVICE   = OFF

DIRECTION-PROCESS     = PROCESS-FIRST       CHECK-PROCESS    = ON
COMBINATION-PROCESS   = FIRST-ACL           CHECK-SUBPROCESS = OFF
ACL-REQUIRED-PROCESS  = OFF

DIRECTION-DISKFILE    = VOLUME-FIRST        CHECK-VOLUME     = OFF
COMBINATION-DISKFILE  = ALL                 CHECK-SUBVOLUME  = ON
ACL-REQUIRED-DISKFILE = OFF                 CHECK-FILENAME   = ON
CLEARONPURGE-DISKFILE = OFF                 CHECK-DISKFILE-PATTERN = OFF

ALLOW-DISKFILE-PERSISTENT = NORMAL
```

You can now test your Safeguard access control lists in warning mode. Guardian security is not checked for objects that have Safeguard protection records because the warning mode fallback option has been set to GRANT.

To verify the results of warning mode access attempts, you can use SAFEART to extract the audit records that were generated as a result of warning mode. For example, the following sequence of commands prints each audit record in the current audit file that has WARNING in the OUTCOME field. The example assumes that the current audit file is named $SECURE.AUDIT.A0000005.

```
=SAFEART
<=AUDIT FILE $secure.audit.A0000005
```

```
<=SET DESTINATION FILE report1
<=SET WHERE OUTCOME=warning
<=START
```

To disable warning mode when you are finished testing the Safeguard security settings:

```
ALTER SAFEGUARD, SYSTEM-WARNING-MODE OFF
```

9 Configuration

This section describes the restricted command ALTER SAFEGUARD. It is intended for trusted users who are members of the SECURITY-ADMINISTRATOR security group. If you have not defined a SECURITY-ADMINISTRATOR group, any super-group user can alter the Safeguard configuration or stop the Safeguard software. (For information about defining security groups, see Section 6, Managing Security Groups.)

Safeguard Attributes

Many of the Safeguard control features are determined by attributes in the configuration file. One of these attributes, for example, controls the minimum password length allowed by the Safeguard software.

You can configure the following aspects of the Safeguard software:

- User authentication attempts (such as the number of failed logon attempts before a timeout occurs)
- Password control (such as requiring a minimum password length and granting a grace period during which a user can change an expired password)
- Priority of access control lists between devices and subdevices
- Priority of access control lists between processes and subprocesses
- Priority of access control lists among volumes, subvolumes, disk files, and diskfile-patterns
- Auditing (such as setting systemwide auditing in addition to the auditing specified in the individual authorization records)
- The logon dialog (such as prohibiting the use of user IDs for logon)
- The command interpreter to be started after a user logs on at a Safeguard terminal
- Exclusive access for the user logged on at a Safeguard terminal
- Client subsystem auditing
- System-level warning mode

You can configure the Safeguard software to suit your own security policy. However, any changes you make are systemwide and might affect system performance and security. For example, configuring the software to audit all system objects might cause severe performance delays. In general, change only attributes that must be changed to implement your security policy.

Table 9-1 on page 9-2 lists the initial values for the configurable Safeguard attributes. In most cases, these initial values are also the default values. The next subsections explain these attributes in detail.

At any time, you can display the current settings of the attributes by issuing the INFO SAFEGUARD command from SAFECOM.

Table 9-1. Safeguard Attribute Default Values (page 1 of 4)

Attribute Name	Initial or Default Value
AUTHENTICATE-MAXIMUM-ATTEMPTS	3
AUTHENTICATE-FAIL-TIMEOUT	60 SECONDS
AUTHENTICATE-FAIL-FREEZE	OFF
PROMPT-BEFORE-STOP[4]	OFF
PASSWORD-HISTORY	0
PASSWORD-MINIMUM-LENGTH*	6
PASSWORD-MAY-CHANGE	0
PASSWORD-EXPIRY-GRACE	0
PASSWORD-REQUIRED	OFF
PASSWORD-ERROR-DETAIL	OFF
PASSWORD-ENCRYPT*	ON
CHECK-DEVICE	ON
CHECK-SUBDEVICE	OFF
DIRECTION-DEVICE	DEVICE-FIRST
COMBINATION-DEVICE	FIRST-ACL
ACL-REQUIRED-DEVICE	OFF
CHECK-PROCESS	ON
CHECK-SUBPROCESS	OFF
DIRECTION-PROCESS	PROCESS-FIRST
COMBINATION-PROCESS	FIRST-ACL
ACL-REQUIRED-PROCESS	OFF
CHECK-VOLUME	OFF
CHECK-SUBVOLUME	OFF

* For systems running H06.06 and later H-series RVUs and G06.29 and later G-series RVUs the default value for PASSWORD-MINIMUM-LENGTH is six and PASSWORD-ENCRYPT is ON.

** Supported only on systems running H06.08 and later H-series RVUs and G06.29 and later G-series RVUs. AUDIT-CLIENT-GUARDIAN is a synonym for AUDIT-CLIENT-SERVICE.

^ Supported only on systems running H06.09 and later H-series RVUs and G06.31 and later G-series RVUs.

^^ Supported only on systems running H06.11 and later H-series RVUs and G06.32 and later G-series RVUs.

& Supported only on systems running J06.03 and later J-series RVUs, H06.14 and later H-series RVUs, and G06.32 and later G-series RVUs.

1 Supported only on systems running J06.08 and later J-series RVUs, H06.19 and later H-series RVUs, and G06.32 and later G-series RVUs.

2 Supported only on systems running J06.10 and later J-series RVUs and H06.21 and later H-series RVUs. PASSWORD-ERROR-DETAIL is supported only on systems running on J06.14 and later J-series RVUs and H06.25 and later H-series RVUs.

3 Supported only on systems running H06.08 and later H-series RVUs and G06.31 and later G-series RVUs.

4 Supported only on systems running J06.16 and later J-series RVUs and H06.27, and later H-series RVUs.

Table 9-1. Safeguard Attribute Default Values (page 2 of 4)

Attribute Name	Initial or Default Value
CHECK-FILENAME	ON
DIRECTION-DISKFILE	VOLUME-FIRST
COMBINATION-DISKFILE	ALL
ACL-REQUIRED-DISKFILE	OFF
CLEARONPURGE-DISKFILE	OFF
AUDIT-AUTHENTICATE-PASS	NONE
AUDIT-AUTHENTICATE-FAIL	NONE
AUDIT-SUBJECT-MANAGE-PASS	NONE
AUDIT-SUBJECT-MANAGE-FAIL	NONE
AUDIT-DEVICE-ACCESS-PASS	NONE
AUDIT-DEVICE-ACCESS-FAIL	NONE
AUDIT-DEVICE-MANAGE-PASS	NONE
AUDIT-DEVICE-MANAGE-FAIL	NONE
AUDIT-PROCESS-ACCESS-PASS	NONE
AUDIT-PROCESS-ACCESS-FAIL	NONE
AUDIT-PROCESS-MANAGE-PASS	NONE
AUDIT-PROCESS-MANAGE-FAIL	NONE
AUDIT-DISKFILE-ACCESS-PASS	NONE
AUDIT-DISKFILE-ACCESS-FAIL	NONE
AUDIT-DISKFILE-MANAGE-PASS	NONE
AUDIT-DISKFILE-MANAGE-FAIL	NONE
AUDIT-DISKFILE-PRIV-LOGON ^^	OFF
AUDIT-OBJECT-ACCESS-PASS	NONE
AUDIT-OBJECT-ACCESS-FAIL	NONE
AUDIT-OBJECT-MANAGE-PASS	NONE

* For systems running H06.06 and later H-series RVUs and G06.29 and later G-series RVUs the default value for PASSWORD-MINIMUM-LENGTH is six and PASSWORD-ENCRYPT is ON.

** Supported only on systems running H06.08 and later H-series RVUs and G06.29 and later G-series RVUs. AUDIT-CLIENT-GUARDIAN is a synonym for AUDIT-CLIENT-SERVICE.

^ Supported only on systems running H06.09 and later H-series RVUs and G06.31 and later G-series RVUs.

^^ Supported only on systems running H06.11 and later H-series RVUs and G06.32 and later G-series RVUs.

& Supported only on systems running J06.03 and later J-series RVUs, H06.14 and later H-series RVUs, and G06.32 and later G-series RVUs.

1 Supported only on systems running J06.08 and later J-series RVUs, H06.19 and later H-series RVUs, and G06.32 and later G-series RVUs.

2 Supported only on systems running J06.10 and later J-series RVUs and H06.21 and later H-series RVUs. PASSWORD-ERROR-DETAIL is supported only on systems running on J06.14 and later J-series RVUs and H06.25 and later H-series RVUs.

3 Supported only on systems running H06.08 and later H-series RVUs and G06.31 and later G-series RVUs.

4 Supported only on systems running J06.16 and later J-series RVUs and H06.27, and later H-series RVUs.

Table 9-1. Safeguard Attribute Default Values (page 3 of 4)

Attribute Name	Initial or Default Value
AUDIT-OBJECT-MANAGE-FAIL	NONE
AUDIT-CLIENT-GUARDIAN**	ON
TERMINAL-EXCLUSIVE-ACCESS	OFF
CI-PROG	$SYSTEM.SYSTEM.TACL
CI-LIB	NONE
CI-SWAP	NONE
CI-CPU	ANY
CI-PRI	149
CI-PARAM-TEXT	NONE
CMON	OFF
CMONTIMEOUT	30 SECONDS
CMONERROR	ACCEPT
BLINDLOGON	ON
NAMELOGON	ON
SYSTEM-WARNING-MODE	OFF
OBJECT-WARNING-MODE	OFF
WARNING-FALLBACK-SECURITY	GUARDIAN
ALLOW-DISKFILE-PERSISTENT	NORMAL
ALLOW-NODE-ID-ACL	OFF
CHECK-DISKFILE-PATTERN	OFF
AUDIT-CLIENT-OSS**	ON
PASSWORD-ALGORITHM*	DES
PASSWORD-MAXIMUM-LENGTH 3	8
PASSWORD-COMPATIBILITY-MODE 3	ON
PASSWORD-UPPERCASE-REQUIRED ^	OFF

* For systems running H06.06 and later H-series RVUs and G06.29 and later G-series RVUs the default value for PASSWORD-MINIMUM-LENGTH is six and PASSWORD-ENCRYPT is ON.

** Supported only on systems running H06.08 and later H-series RVUs and G06.29 and later G-series RVUs. AUDIT-CLIENT-GUARDIAN is a synonym for AUDIT-CLIENT-SERVICE.

^ Supported only on systems running H06.09 and later H-series RVUs and G06.31 and later G-series RVUs.

^^ Supported only on systems running H06.11 and later H-series RVUs and G06.32 and later G-series RVUs.

& Supported only on systems running J06.03 and later J-series RVUs, H06.14 and later H-series RVUs, and G06.32 and later G-series RVUs.

1 Supported only on systems running J06.08 and later J-series RVUs, H06.19 and later H-series RVUs, and G06.32 and later G-series RVUs.

2 Supported only on systems running J06.10 and later J-series RVUs and H06.21 and later H-series RVUs. PASSWORD-ERROR-DETAIL is supported only on systems running on J06.14 and later J-series RVUs and H06.25 and later H-series RVUs.

3 Supported only on systems running H06.08 and later H-series RVUs and G06.31 and later G-series RVUs.

4 Supported only on systems running J06.16 and later J-series RVUs and H06.27, and later H-series RVUs.

Table 9-1. Safeguard Attribute Default Values (page 4 of 4)

Attribute Name	Initial or Default Value
PASSWORD-LOWERCASE-REQUIRED ^	OFF
PASSWORD-NUMERIC-REQUIRED ^	OFF
PASSWORD-SPECIALCHAR-REQUIRED ^	OFF
PASSWORD-SPACES-ALLOWED ^	OFF
PASSWORD-MIN-QUALITY-REQUIRED ^	0
AUDIT-EXCLUDE-FIELD&	NONE
AUDIT-EXCLUDE-VALUE&	NONE
AUDIT-OSS-FILTER1	OFF
AUDIT-TACL-LOGOFF1	OFF
DYNAMIC-PROC-UPDATE2	OFF

* For systems running H06.06 and later H-series RVUs and G06.29 and later G-series RVUs the default value for PASSWORD-MINIMUM-LENGTH is six and PASSWORD-ENCRYPT is ON.
** Supported only on systems running H06.08 and later H-series RVUs and G06.29 and later G-series RVUs. AUDIT-CLIENT-GUARDIAN is a synonym for AUDIT-CLIENT-SERVICE.
^ Supported only on systems running H06.09 and later H-series RVUs and G06.31 and later G-series RVUs.
^^ Supported only on systems running H06.11 and later H-series RVUs and G06.32 and later G-series RVUs.
& Supported only on systems running J06.03 and later J-series RVUs, H06.14 and later H-series RVUs, and G06.32 and later G-series RVUs.
1 Supported only on systems running J06.08 and later J-series RVUs, H06.19 and later H-series RVUs, and G06.32 and later G-series RVUs.
2 Supported only on systems running J06.10 and later J-series RVUs and H06.21 and later H-series RVUs. PASSWORD-ERROR-DETAIL is supported only on systems running on J06.14 and later J-series RVUs and H06.25 and later H-series RVUs.
3 Supported only on systems running H06.08 and later H-series RVUs and G06.31 and later G-series RVUs.
4 Supported only on systems running J06.16 and later J-series RVUs and H06.27, and later H-series RVUs.

Configuring User Authentication

Intruders often gain access to a system by making multiple attempts to guess a user's password. You can configure the Safeguard software to help prevent this possibility by setting the maximum number of logon attempts for a user ID at one time, and setting a delay or freeze if that number is exceeded.

These Safeguard attributes relate to user authentication:

`AUTHENTICATE-MAXIMUM-ATTEMPTS`

The maximum number of failed logon attempts for a single user ID before the Safeguard software freezes the user ID or causes a timeout to occur. The default value is 3.

AUTHENTICATE-FAIL-TIMEOUT

The specified timeout for a user ID if AUTHENTICATE-MAXIMUM-ATTEMPTS is exceeded. The default is 60 seconds. The command interpreter process at the terminal remains locked for the duration of the timeout period.

△ **Caution.** Because the command interpreter process at the terminal remains locked for the duration of the AUTHENTICATE-FAIL-TIMEOUT period, avoid specifying an unreasonably long period. The terminal is effectively not usable during this period. The only recovery is to start a new process at the terminal.

AUTHENTICATE-FAIL-FREEZE

Freezes the user ID if AUTHENTICATE-MAXIMUM-ATTEMPTS is exceeded. (The user ID can be thawed with the THAW USER command.) The initial value is OFF.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to allow a maximum of five failed logon attempts for a single user ID:

```
=ALTER SAFEGUARD, AUTHENTICATE-MAXIMUM-ATTEMPTS 5
```

To freeze the user ID if this value is exceeded:

```
=ALTER SAFEGUARD, AUTHENTICATE-FAIL-FREEZE ON
```

△ **Caution.** If you set AUTHENTICATE-FAIL-FREEZE ON, a user can freeze the user IDs of others by attempting to log on with those others' user names or user IDs.

You can change more than one attribute with a single command. For example, to change the maximum number of failed logon attempts to six and the timeout period to five minutes:

```
=ALTER SAFEGUARD, AUTHENTICATE-MAXIMUM-ATTEMPTS 6, &
=AUTHENTICATE-FAIL-TIMEOUT 5 MINUTES
```

Configuring Password Control

The Safeguard software provides systemwide control over the use of passwords. For example, you can require a minimum length for all passwords and not allow anyone to reuse recent passwords.

These Safeguard attributes relate to password control:

PASSWORD-COMPATIBILITY-MODE

Specifies that only first eight characters of the password will be considered during password change. This attribute can take effect only when PASSWORD-ENCRYPT is ON and PASSWORD-ALGORITHM is HMAC256. The initial value is ON.

Note. This attribute is supported only on systems running H06.08 and later H-series RVUs and G06.31 and later G-series.

PASSWORD-HISTORY

Records a specified number of previously used passwords for each user and does not allow a user to change his or her password to any password in this history. You can specify a history of 0 to 60 passwords. (If you specify a history of more than 20 passwords, you must convert the USERID files as described in Section 10, Installation and Management.) Values of 0 and 1 allow the user to reuse any password, even if used recently. The initial value is 0.

PASSWORD-MAXIMUM-LENGTH

Specifies the maximum acceptable length of a password. The initial value is eight and the maximum value is 64.

Note. This attribute is supported only on systems running H06.08 and later H-series RVUs and G06.31 and later G-series RVUs.

PASSWORD-MINIMUM-LENGTH

Specifies the minimum acceptable length of a password. A value of 0 indicates that null passwords can be accepted. The initial value is 0.

Note. The default value is six only on systems running H06.06 and later H-series RVUs and G06.29 and later G-series RVUs

PASSWORD-REQUIRED

Requires the use of a password by all users when logging on as another user. This includes logon attempts by the local super ID and group managers. The initial value is OFF.

PASSWORD-MAY-CHANGE

Specifies the number of days prior to expiration that a user can change a password. (Expiration is determined by the PASSWORD-MUST-CHANGE attribute in the user authentication record.) A value of 0 allows the password to be changed at any time. The default is 0.

If the PASSWORD-MAY-CHANGE period is greater than the PASSWORD-MUST-CHANGE period in a user authentication record, that user password can be changed at any time.

Note. Setting PASSWORD-MAY-CHANGE for the super ID has no impact.

PASSWORD-UPPERCASE-REQUIRED {ON / OFF}

Specifies whether the user password will be enforced to have at least one uppercase character. The initial value is OFF.

The PASSWORD-UPPERCASE-REQUIRED attribute can be set to ON when PASSWORD-ALGORITHM is HMAC256 and PASSWORD-ENCRYPT is ON.

Note.

- On systems running J06.11 and later J-series RVUs and H06.22 and later H-series RVUs, the PASSWORD-UPPERCASE-REQUIRED attribute supports the DES and HMAC256 password algorithms. Therefore, the PASSWORD-UPPERCASE-REQUIRED attribute can be set to ON when PASSWORD-ENCRYPT is ON.
- The PASSWORD-UPPERCASE-REQUIRED attribute will take effect only when the PASSWORD-MIN-QUALITY-REQUIRED attribute is set to a value greater than 0.
- The PASSWORD-UPPERCASE-REQUIRED attribute is supported only on systems running H06.09 and later H-series RVUs and G06.31 and later G-series RVUs.

`PASSWORD-LOWERCASE-REQUIRED {ON / OFF}`

Specifies whether the user password will be enforced to have at least one lowercase character. The initial value is OFF.

The PASSWORD-LOWERCASE-REQUIRED attribute can be set to ON when PASSWORD-ALGORITHM is HMAC256 and PASSWORD-ENCRYPT is ON.

Note.

- On systems running J06.11 and later J-series RVUs and H06.22 and later H-series RVUs, the PASSWORD-LOWERCASE-REQUIRED attribute supports the DES and HMAC256 password algorithms. Therefore, the PASSWORD-LOWERCASE-REQUIRED attribute can be set to ON when PASSWORD-ENCRYPT is ON.
- The PASSWORD-LOWERCASE-REQUIRED attribute will take effect only when the PASSWORD-MIN-QUALITY-REQUIRED attribute is set to a value greater than 0.
- The PASSWORD-LOWERCASE-REQUIRED attribute is supported only on systems running H06.09 and later H-series RVUs and G06.31 and later G-series RVUs.

`PASSWORD-NUMERIC-REQUIRED {ON / OFF}`

Specifies whether the user password will be enforced to have at least one numeric character. The initial value is OFF.

The PASSWORD-NUMERIC-REQUIRED attribute can be set to ON when PASSWORD-ALGORITHM is HMAC256 and PASSWORD-ENCRYPT is ON.

Note.

- On systems running J06.11 and later J-series RVUs and H06.22 and later H-series RVUs, the PASSWORD-NUMERIC-REQUIRED attribute supports the DES and HMAC256 password algorithms. Therefore, the PASSWORD-NUMERIC-REQUIRED attribute can be set to ON when PASSWORD-ENCRYPT is ON.
- The PASSWORD-NUMERIC-REQUIRED attribute will take effect only when the PASSWORD-MIN-QUALITY-REQUIRED attribute is set to a value greater than 0.
- The PASSWORD-NUMERIC-REQUIRED attribute is supported only on systems running H06.09 and later H-series RVUs and G06.31 and later G-series RVUs.

PASSWORD-SPECIALCHAR-REQUIRED {ON / OFF}

Specifies whether the user password will be enforced to have at least one special character. The initial value is OFF.

The PASSWORD-SPECIALCHAR-REQUIRED attribute can be set to ON when PASSWORD-ALGORITHM is HMAC256 and PASSWORD-ENCRYPT is ON.

Note.

- On systems running J06.11 and later J-series RVUs and H06.22 and later H-series RVUs, the PASSWORD-SPECIALCHAR-REQUIRED attribute supports DES and HMAC256 password algorithms. Therefore, the PASSWORD-SPECIALCHAR-REQUIRED attribute can be set to ON when PASSWORD-ENCRYPT is ON.
- The PASSWORD-SPECIAL-CHAR-REQUIRED attribute will take effect only when the PASSWORD-MIN-QUALITY-REQUIRED attribute is set to a value greater than 0.
- The PASSWORD-SPECIALCHAR-REQUIRED attribute is supported only on systems running H06.09 and later H-series RVUs and G06.31 and later G-series RVUs.

PASSWORD-SPACES-ALLOWED {ON / OFF}

Specifies whether the user password will be allowed to have spaces. The initial value is OFF.

Note. This attribute is supported only on systems running H06.09 and later H-series RVUs and G06.31 and later G-series RVUs.

PASSWORD-MIN-QUALITY-REQUIRED

Specifies the minimum quality criteria that must be met when a password is set or changed. The valid values of PASSWORD-MIN-QUALITY-REQUIRED range from 0 to 5. The initial value is 0.

Note. When any of the following password quality attributes is enabled, PASSWORD-MIN-QUALITY-REQUIRED will be automatically set from 0 to 1:

- PASSWORD-UPPERCASE-REQUIRED
- PASSWORD-LOWERCASE-REQUIRED
- PASSWORD-NUMERIC-REQUIRED
- PASSWORD-SPECIALCHAR-REQUIRED
- PASSWORD-ALPHA-REQUIRED

Note. This attribute is supported only on systems running H06.09 and later H-series RVUs and G06.31 and later G-series RVUs.

The following considerations apply to the PASSWORD-MIN-QUALITY-REQUIRED attribute:

- PASSWORD-MIN-QUALITY-REQUIRED can be modified only when PASSWORD-ENCRYPT is ON.

  Note. On systems running J06.11 and later J-series RVUs and H06.22 and later H-series RVUs, the PASSWORD-MIN-QUALITY-REQUIRED attribute can be modified when PASSWORD-ENCRYPT is ON.

- When PASSWORD-ENCRYPT is OFF, an attempt to alter the quality attributes results in an error. The error messages displayed are:

```
THIS ATTRIBUTE CANNOT BE MODIFIED UNLESS PASSWORD-ENCRYPT = ON; COMMAND
NOT EXECUTED.
```

- On systems running J06.11 and later J-series RVUs and H06.22 and later H-series RVUs, when PASSWORD-ENCRYPT is OFF, an attempt to alter the quality attributes results in an error. The error messages displayed are:

```
THIS ATTRIBUTE CANNOT BE MODIFIED UNLESS PASSWORD-ENCRYPT = ON; COMMAND
NOT EXECUTED.
```

- PASSWORD-MIN-QUALITY-REQUIRED set to a value greater than 0, indicates that the PASSWORD-UPPERCASE-REQUIRED, PASSWORD-LOWERCASE-REQUIRED, PASSWORD-NUMERIC-REQUIRED, PASSWORD-SPECIALCHAR-REQUIRED and PASSWORD-ALPHA-REQUIRED attributes, if enabled, meet the password quality criteria.
- When setting PASSWORD-MIN-QUALITY-REQUIRED to a value greater than 0, ensure that the sum of the values of the effective password quality attributes (PASSWORD-MIN-UPPERCASE-REQ, PASSWORD-MIN-LOWERCASE-REQ, PASSWORD-MIN-NUMERIC-REQ, PASSWORD-MIN-SPECIALCHAR-REQ or PASSWORD-MIN-ALPHA-REQ) must not be greater than the value of the PASSWORD-MAXIMUM-LEN attribute.
- When PASSWORD-MIN-QUALITY-REQUIRED is set to a value greater than 0, and if PASSWORD-ENCRYPT is changed from ON to OFF, or PASSWORD-ALGORITHM is changed from HMAC256 to DES, the PASSWORD-MIN-QUALITY-REQUIRED attribute is reset to 0.

  Note. On systems running J06.11 and later J-series RVUs and H06.22 and later H-series RVUs, when PASSWORD-MIN-QUALITY-REQUIRED is set to a value greater than 0, and if PASSWORD-ENCRYPT is changed from ON to OFF, the PASSWORD-MIN-QUALITY-REQUIRED attribute is reset to 0.

`PASSWORD-MIN-UPPERCASE-REQ`

Specifies the minimum number of uppercase characters required in a user password when it is set or changed.

The PASSWORD-MIN-UPPERCASE-REQ attribute can have values from 0 through 8. The initial value is 0.

Note. This attribute is supported only on systems running J06.11 or later J-series RVUs and H06.22 or later H-series RVUs.

The following considerations apply to the PASSWORD-MIN-UPPERCASE-REQ attribute:

- The PASSWORD-MIN-UPPERCASE-REQ attribute will take effect only when the PASSWORD-UPPERCASE-REQUIRED attribute is enabled.
- When the PASSWORD-UPPERCASE-REQUIRED attribute is changed from OFF to ON, Safeguard sets the numeric value of the PASSWORD-MIN-UPPERCASE-REQ attribute to 1.
- When the PASSWORD-UPPERCASE-REQUIRED attribute is changed from ON to OFF, Safeguard sets the numeric value of the PASSWORD-MIN-UPPERCASE-REQ attribute to 0.
- The sum of the values of the effective password quality attributes (PASSWORD-MIN-UPPERCASE-REQ, PASSWORD-MIN-LOWERCASE-REQ, PASSWORD-MIN-NUMERIC-REQ, PASSWORD-MIN-SPECIALCHAR-REQ or PASSWORD-MIN-ALPHA-REQ) must not be greater than the value of the PASSWORD-MAXIMUM-LEN attribute.

`PASSWORD-MIN-LOWERCASE-REQ`

Specifies the minimum number of lowercase characters required in a user password when it is set or changed.

The PASSWORD-MIN-LOWERCASE-REQ attribute can have values from 0 through 8. The initial value is 0.

Note. This attribute is supported only on systems running J06.11 or later J-series RVUs and H06.22 or later H-series RVUs.

The following considerations apply to the PASSWORD-MIN-LOWERCASE-REQ attribute:

- The PASSWORD-MIN-LOWERCASE-REQ attribute will take effect only when the PASSWORD-LOWERCASE-REQUIRED attribute is enabled.
- When the PASSWORD-LOWERCASE-REQUIRED attribute is changed from OFF to ON, Safeguard sets the numeric value of the PASSWORD-MIN-LOWERCASE-REQ attribute to 1.
- When the PASSWORD-LOWERCASE-REQUIRED attribute is changed from ON to OFF, Safeguard sets the numeric value of the PASSWORD-MIN-LOWERCASE-REQ attribute to 0.
- The sum of the values of the effective password quality attributes (PASSWORD-MIN-UPPERCASE-REQ, PASSWORD-MIN-LOWERCASE-REQ,

PASSWORD-MIN-NUMERIC-REQ, PASSWORD-MIN-SPECIALCHAR-REQ or PASSWORD-MIN-ALPHA-REQ) must not be greater than the value of the PASSWORD-MAXIMUM-LEN attribute.

`PASSWORD-MIN-NUMERIC-REQ`

Specifies the minimum number of numeric characters required in a user password when it is set or changed.

The PASSWORD-MIN-NUMERIC-REQ attribute can have values from 0 through 8. The initial value is 0.

Note. This attribute is supported only on systems running J06.11 or later J-series RVUs and H06.22 or later H-series RVUs.

The following considerations apply to the PASSWORD-MIN-NUMERIC-REQ attribute:

- The PASSWORD-MIN-NUMERIC-REQ attribute will take effect only when the PASSWORD-NUMERIC-REQUIRED attribute is enabled.
- When the PASSWORD-NUMERIC-REQUIRED attribute is changed from OFF to ON, Safeguard sets the numeric value of the PASSWORD-MIN-NUMERIC-REQ attribute to 1.
- When the PASSWORD-NUMERIC-REQUIRED attribute is changed from ON to OFF, Safeguard sets the numeric value of the PASSWORD-MIN-NUMERIC-REQ attribute to 0.
- The sum of the values of the effective password quality attributes (PASSWORD-MIN-UPPERCASE-REQ, PASSWORD-MIN-LOWERCASE-REQ, PASSWORD-MIN-NUMERIC-REQ, PASSWORD-MIN-SPECIALCHAR-REQ or PASSWORD-MIN-ALPHA-REQ) must not be greater than the value of the PASSWORD-MAXIMUM-LEN attribute.

`PASSWORD-MIN-SPECIALCHAR-REQ`

Specifies the minimum number of special characters required in a user password when it is set or changed.

The PASSWORD-MIN-SPECIALCHAR-REQ attribute can have values from 0 through 8. The initial value is 0.

Note. This attribute is supported only on systems running J06.11 or later J-series RVUs and H06.22 or later H-series RVUs.

The following considerations apply to the PASSWORD-MIN-SPECIALCHAR-REQ attribute:

- The PASSWORD-MIN-SPECIALCHAR-REQ attribute will take effect only when the PASSWORD-SPECIALCHAR-REQUIRED attribute is enabled.

- When the PASSWORD-SPECIALCHAR-REQUIRED attribute is changed from OFF to ON, Safeguard sets the numeric value of the PASSWORD-MIN-SPECIALCHAR-REQ attribute to 1.
- When the PASSWORD-SPECIALCHAR-REQUIRED attribute is changed from ON to OFF, Safeguard sets the numeric value of the PASSWORD-MIN-SPECIALCHAR-REQ attribute to 0.
- The sum of the values of the effective password quality attributes (PASSWORD-MIN-UPPERCASE-REQ, PASSWORD-MIN-LOWERCASE-REQ, PASSWORD-MIN-NUMERIC-REQ, PASSWORD-MIN-SPECIALCHAR-REQ or PASSWORD-MIN-ALPHA-REQ) must not be greater than the value of the PASSWORD-MAXIMUM-LEN attribute.

`PASSWORD-ALPHA-REQUIRED`

Specifies whether the user password will be enforced to have at least one alphabetic character. The initial value is 0.

The PASSWORD-SPECIALCHAR-REQUIRED attribute can be set to ON when PASSWORD-ENCRYPT is ON.

Note. This attribute is supported only on systems running J06.11 and later J-series RVUs and H06.22 and later H-series RVUs.

`PASSWORD-MIN-ALPHA-REQ`

Specifies the minimum number of alphabetic characters required in a user password when it is set or changed.

The PASSWORD-MIN-ALPHA-REQ attribute can have values from 0 through 8. The initial value is 0.

Note. This attribute is supported only on systems running J06.11 or later J-series RVUs and H06.22 or later H-series RVUs.

The following considerations apply to the PASSWORD-MIN-ALPHA-REQ attribute:

- The PASSWORD-MIN-ALPHA-REQ attribute will take effect only when the PASSWORD-ALPHA-REQUIRED attribute is enabled.
- When the PASSWORD-ALPHA-REQUIRED attribute is changed from OFF to ON, Safeguard sets the numeric value of the PASSWORD-MIN-ALPHA-REQ attribute to 1.
- When the PASSWORD-ALPHA-REQUIRED attribute is changed from ON to OFF, Safeguard sets the numeric value of the PASSWORD-MIN-ALPHA-REQ attribute to 0.
- The sum of the values of the effective password quality attributes (PASSWORD-MIN-UPPERCASE-REQ, PASSWORD-MIN-LOWERCASE-REQ, PASSWORD-MIN-NUMERIC-REQ, PASSWORD-MIN-SPECIALCHAR-REQ or

PASSWORD-MIN-ALPHA-REQ) must not be greater than the value of the PASSWORD-MAXIMUM-LEN attribute.

PASSWORD-ERROR-DETAIL

Determines whether a detailed error message is displayed to the user when the password supplied does not meet the minimum complexity as defined.

Detailed error message is displayed when PASSWORD-ERROR-DETAIL is ON as per the password minimum required complexity. A default error message is displayed when PASSWORD-ERROR-DETAIL is OFF.

However, the default value is OFF. This attribute defines part of the SAFEGUARD global configuration.

Note. This attribute is supported only on systems running on J06.14 and later J-series RVUs and H06.25 and later H-series RVUs.

PASSWORD-EXPIRY-GRACE

Specifies the number of days after password expiration during which users are allowed to change their expired passwords during logon. A value of 0 means no grace period. The default is 0.

PASSWORD-EXPIRY-GRACE can also be specified in individual user authentication records. If the value of this attribute is not specified in a user authentication record, the Safeguard software uses the value specified in the configuration record.

PASSWORD-ENCRYPT

Specifies if, when a password for any user ID is changed, an encrypted version of the clear-text password in case of Data Encryption Standard (DES) or the message digest (hashed password) in case of HMAC256, is stored in the user database. When set to OFF, the password is stored as clear text in the user database. The initial value is ON.

Note. The default value is ON only on systems running H06.06 and later H-series RVUs and G06.29 and later G-series RVUs.

Note. If the password is encrypted by the PASSWORD program, the Safeguard software receives an encrypted version of the password and cannot check for PASSWORD- MINIMUM-LENGTH. However, if the Safeguard software performs the encryption through the PASSWORD-ENCRYPT attribute, it checks PASSWORD-MINIMUM-LENGTH before it encrypts the password.

PASSWORD-ALGORITHM

Indicates the algorithm to encrypt passwords when they are changed. The initial value is DES.

Note. This attribute is supported only on systems running H06.06 and later H-series RVUs and G06.29 and later G-series RVUs.

DES

Indicates to use the DES algorithm to encrypt passwords. This is the initial value. Encrypted passwords are stored in the L/USERID and L/USERAX files.

HMAC256

Indicates to use HMAC with SHA-256 algorithm to encrypt passwords, when PASSWORD-ENCRYPT is ON. Encrypted passwords are stored in the L/USERAX files.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to maintain a history of the last 10 passwords for each user (and not allow reuse of these passwords):

```
=ALTER SAFEGUARD, PASSWORD-HISTORY 10
```

To grant users a 15-day grace period during which they can change their expired passwords during logon:

```
=ALTER SAFEGUARD, PASSWORD-EXPIRY-GRACE 15 DAYS
```

You can change more than one attribute with a single command. To require a minimum password length of six characters and to have passwords encrypted:

```
=ALTER SAFEGUARD, PASSWORD-MINIMUM-LENGTH 6, &
=PASSWORD-ENCRYPT ON
```

Note. Each time a user password is changed or the user PASSWORD-MUST-CHANGE period is changed, the Safeguard software uses the PASSWORD-MAY-CHANGE value to calculate the new date on which that user password can be changed. It also calculates a PASSWORD-EXPIRES date for the user based on the PASSWORD-MUST-CHANGE period defined in the user authentication record. A user can change the password anytime between the PASSWORD-MAY-CHANGE date and the PASSWORD-EXPIRES date. These dates are calculated differently depending on who changes the password.

If the user changes the password, the PASSWORD-EXPIRES date is calculated by adding the PASSWORD-MUST-CHANGE period to the current date. The PASSWORD-MAY-CHANGE date is calculated by subtracting the PASSWORD-MAY-CHANGE period from the PASSWORD-EXPIRES date.

If the owner of the user authentication record changes the password, the PASSWORD-MAY-CHANGE date is set to *NONE* so that the user can change the password immediately. In this instance, the PASSWORD-EXPIRES date is calculated by adding the PASSWORD-MUST-CHANGE period to the current date.

Consider this example with the attributes set as:

```
PASSWORD-MUST-CHANGE EVERY = 20 DAYS
PASSWORD-MAY-CHANGE        =  5 DAYS
```

On July 1, the owner of the user authentication record changes the user's password. These dates are calculated:

```
PASSWORD-MAY-CHANGE = * NONE *
PASSWORD-EXPIRES    = 21JULY1993
```

The user must change the password in the next 20 days because the password expires on July 21.

On July 21, the user changes the password. These new dates are calculated:

```
PASSWORD-MAY-CHANGE = 17JULY1993
PASSWORD-EXPIRES    = 22JULY1993
```

The user cannot change the password until July 17. The user then has only five days to change the password before it expires. If someone learns the user's password before July 17, the user should ask the owner of the user authentication record to change the password.

Configuring Device Control

If access control lists exist for both devices and subdevices, the Safeguard software must know which one to use. You can set the attributes that control how this is determined.

These Safeguard attributes relate to device access control lists:

`CHECK-DEVICE`

Access control lists are checked at the device level. The initial value is ON.

`CHECK-SUBDEVICE`

Access control lists are checked at the subdevice level. The initial value is OFF.

`DIRECTION-DEVICE`

Determines which direction to search for an access control list when both CHECK-DEVICE and CHECK-SUBDEVICE are ON. The value can be either DEVICE-FIRST or SUBDEVICE-FIRST. This attribute is used in conjunction with COMBINATION-DEVICE. (For more information, see the following note.) The initial value is DEVICE-FIRST.

`COMBINATION-DEVICE`

Determines how to resolve conflicts between device and subdevice access control lists. This attribute is used in conjunction with DIRECTION-DEVICE. (For more

information, see the following note.) The value can be FIRST-ACL, FIRST-RULE, or ALL. The initial value is FIRST-ACL.

`ACL-REQUIRED-DEVICE`

If no access control list is found, access is denied. If this attribute is OFF, and no access control list is found, Guardian rules apply. The initial value is OFF.

Note. COMBINATION-DEVICE resolves conflicts between access control lists if CHECK-DEVICE and CHECK-SUBDEVICE are both ON. The Safeguard software searches for an access control list in the order determined by DIRECTION-DEVICE. If you want to use the first access control list it finds, specify FIRST-ACL. If you want the search to continue until it finds an access control list that involves the user ID (either ACCESS or DENY), specify FIRST-RULE. If you want to allow access only if specified on both access control lists, specify ALL.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to check access control lists at both the device and subdevice levels:

```
=ALTER SAFEGUARD, CHECK-DEVICE ON, CHECK-SUBDEVICE ON
```

This command specifies that the Safeguard software is to use the first access control list it finds, checking at the device level first:

```
=ALTER SAFEGUARD, COMBINATION-DEVICE FIRST-ACL, &
=DIRECTION-DEVICE DEVICE-FIRST
```

Configuring Process Control

If access control lists exist for both process and subprocess names, the Safeguard software must know which one to use. You can set the attributes that control how this is determined.

These Safeguard attributes relate to process access control lists:

`CHECK-PROCESS`

Access control lists are checked at the process level. The initial value is ON.

`CHECK-SUBPROCESS`

Access control lists are at the subprocess level. The initial value is OFF.

`DIRECTION-PROCESS`

Determines which direction to search for an access control list if both CHECK-PROCESS and CHECK-SUBPROCESS are ON. The value can be either PROCESS-FIRST or SUBPROCESS-FIRST. This attribute is used in conjunction with COMBINATION-PROCESS. (See the following note.) The initial value is PROCESS-FIRST.

COMBINATION-PROCESS

Determines how conflicts are resolved between process and subprocess access control lists. This attribute is used in conjunction with DIRECTION-PROCESS. (For more information, see the following note.) The value can be FIRST-ACL, FIRST-RULE, or ALL. The initial value is FIRST-ACL.

ACL-REQUIRED-PROCESS

If no access control list is found, access is denied. If this attribute is OFF, and no access control list is found, Guardian rules apply. The initial value is OFF.

Note. COMBINATION-PROCESS resolves conflicts between access control lists if CHECK-PROCESS and CHECK-SUBPROCESS are both ON. The Safeguard software searches for an access control list in the order determined by DIRECTION-PROCESS. If you want to use the first access control list it finds, specify FIRST-ACL. If you want the search the search to continue until it finds an access control list that involves the user ID (either ACCESS or DENY), specify FIRST-RULE. If you want to allow access only if specified on both access control lists, specify ALL.

If you use the special NAMED and UNNAMED process protection records, specify FIRST-RULE to ensure this feature functions as intended.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to check access control lists at both the subprocess and process levels:

```
=ALTER SAFEGUARD, CHECK-PROCESS ON, CHECK-SUBPROCESS ON
```

This command tells the Safeguard software to check at the process level first and to search until it finds an ACL with the user ID of the user requesting access:

```
=ALTER SAFEGUARD, DIRECTION-PROCESS PROCESS-FIRST, &
=COMBINATION-PROCESS FIRST-RULE
```

Configuring Disk-File Control

If access control lists exist for volumes, subvolumes, and disk files, the Safeguard software must know which one to use. You can set the attributes that control how this is determined.

These Safeguard attributes relate to disk-file access control lists:

CHECK-VOLUME

Access control lists are checked at the volume level. The initial value is OFF. The Safeguard software checks for CREATE authority at the volume level even when CHECK -VOLUME is OFF.

CHECK-SUBVOLUME

Access control lists are checked at the subvolume level. The initial value is OFF. The Safeguard software checks for CREATE authority at the subvolume level even when CHECK-SUBVOLUME is OFF.

CHECK-FILENAME

Access control lists are checked at the disk-file level. The initial value is ON.

DIRECTION-DISKFILE

Determines which direction to search for access control lists if more than one of the preceding attributes is ON. The value can be either VOLUME-FIRST or FILENAME-FIRST. This attribute is used in conjunction with COMBINATION-DISKFILE. (For more information, see the following note.) The initial value is VOLUME-FIRST.

COMBINATION-DISKFILE

Determines how conflicts are resolved among volume, subvolume, and disk file access control lists. This attribute is used in conjunction with DIRECTION-DISKFILE (for more information, see the following note). The value can be FIRST-ACL, FIRST-RULE, or ALL. The initial value is ALL.

Note. COMBINATION-DISKFILE resolves conflicts among volume, subvolume, and disk-file access control lists. The Safeguard software searches for an access control list in the order determined by DIRECTION-DISKFILE. If you want to use the first access control list it finds, specify FIRST-ACL. If you want the search to continue until it finds an access control list that involves the user ID (either ACCESS or DENY), specify FIRST-RULE. If you want to allow access only if specified on all access control lists, specify ALL.

ACL-REQUIRED-DISKFILE

If no access control list is found, access is denied. If this attribute is OFF, and no Safeguard protection record is found, the Guardian security settings are used. The initial value is OFF.

△ **Caution.** If you set ACL-REQUIRED-DISKFILE ON, you must have an access control list for the SAFECOM program file that grants execute authority to you. Otherwise, when you complete your current session, you cannot control Safeguard through the SAFECOM command interpreter because you cannot run SAFECOM.

CLEARONPURGE-DISKFILE

Sets disk files to all zeros when purged. If set to OFF, the disk space is deallocated when purged, but the data is not set to zeros. The initial value is OFF.

Note. When set to ON, the value of this attribute remains effective even after the Safeguard subsystem is stopped. If you want this feature to be effective only when Safeguard is up, you must set the value to OFF before stopping Safeguard.

`CHECK-DISKFILE-PATTERN`

Specifies how to search diskfile patterns.

`OFF`

specifies that no pattern searches will occur.

`FIRST`

specifies that pattern searching will occur first, if and only if the result is NORECORD then the normal search for a protection record will occur.

`LAST`

specify that pattern searching will occur after the normal search if and only if the normal search result is NORECORD.

`ONLY`

specifies that only pattern searching will occur. That is, normal non-pattern searching will not be performed even if the pattern search returns NORECORD.

`MID`

Specifies that pattern based protection records will be searched:

- After the diskfile protection record search returns NORECORD when Direction-Diskfile is set to Filename-First.
- Before the diskfile protection record search, when the Direction-Diskfile is set to VOLUME-FIRST, and the VOLUME and SUBVOLUME protection record search returns NORECORD.

Note. The MID option is supported only on systems running J06.08 and later J-series RVUs and H06.19 and later H-series RVUs.

△ **Caution.** Any user can add a diskfile-pattern to the database and thereby is able to control file access across an entire volume. If the CHECK-DISKFILE-PATTERN FIRST/LAST/ONLY configuration is needed, use the ADD OBJECTTYPE DISKFILE-PATTERN command to specify who can control diskfile-patterns. For more information, see Section 5, OBJECTTYPE Control.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to check access control lists at the volume, subvolume, and disk file levels, issue this command:

```
=ALTER SAFEGUARD, CHECK-DISKFILE ON, CHECK-VOLUME ON, &
=CHECK-SUBVOLUME ON
```

This command specifies that the Safeguard software is to use the first access control list it finds in this following order—disk file, subvolume, volume:

```
=ALTER SAFEGUARD, COMBINATION-DISKFILE FIRST-ACL, &
=DIRECTION-DISKFILE DISKFILE-FIRST
```

△ **Caution.** If you set CHECK-SUBVOLUME ON and set DIRECTION-DISKFILE to VOLUME-FIRST, any user can gain access to someone else's files. All files that are in subvolumes that have not been added to the Safeguard database are vulnerable. This situation occurs because any user can add the subvolume to the database and thereby own it. If this configuration is needed, use the ADD OBJECTTYPE or ALTER OBJECTTYPE command to specify who can control subvolumes. For more information, see Section 5, OBJECTTYPE Control.

Configuring Safeguard Auditing

Normally, the Safeguard software audits only items that have auditing specified in their protection records. However, you can configure systemwide auditing so that auditing is performed even if it is not specified in individual protection records. You can configure Safeguard auditing:

- All attempts relating to user authentication
- All devices and their authorization records
- All processes and their authorization records
- All disk files and their authorization records
- All system objects (devices, processes, and disk files) and their authorization records

Auditing specified by configuration supplements the settings in the individual authorization records (if the Safeguard software is configured to check the individual record). For example, if an individual disk file record is set to audit ALL access attempts and the Safeguard configuration is set to audit NONE of the disk file access attempts, both local and remote access attempts are audited for the individual disk file. Specifying both systemwide and individual auditing does not cause duplicate records to be generated for audited events.

Note. Some of the global configuration attributes that control systemwide auditing also affect client auditing. For details concerning these attributes, refer to the *Safeguard Audit Service Manual*.

For more information about systemwide auditing, see the *Safeguard Audit Service Manual.*

Configuring User Authentication Auditing

You can configure systemwide auditing of user authentication in addition to the audit settings in the individual user authentication records.

These Safeguard attributes relate to user authentication auditing:

`AUDIT-AUTHENTICATE-PASS`

Successful user and alias logon attempts are audited. This setting supplements the audit settings in the user or alias authentication record. The conditions can be ALL, NONE, or LOCAL. The default is NONE.

`AUDIT-AUTHENTICATE-FAIL`

Unsuccessful user and alias logon attempts are audited. This setting supplements the audit settings in the user or alias authentication record. The conditions can be ALL, NONE, or LOCAL. If set to ALL, logon attempts with invalid user IDs are also audited. The default is NONE.

`AUDIT-SUBJECT-MANAGE-PASS`

Successful attempts to create or manage a user or alias authentication record or a group definition record are audited. This setting supplements the audit settings in the user or alias record. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

`AUDIT-SUBJECT-MANAGE-FAIL`

Unsuccessful attempts to create or manage a user or alias authentication record or a group definition record are audited. This setting supplements the audit settings in the user or alias record. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to audit successful and unsuccessful local logon attempts:

```
=ALTER SAFEGUARD, AUDIT-AUTHENTICATE LOCAL
```

Note the use of audit specification shorthand in this command. For more information, see the *Safeguard Audit Service Manual*.

Configuring Device Auditing

You can configure systemwide auditing of all nondisk devices in addition to the audit settings in the individual device authorization records. Devices can be audited at the local level, at the remote level, or at both levels (ALL).

These Safeguard attributes relate to device auditing:

`AUDIT-DEVICE-ACCESS-PASS`

Successful attempts to access all devices or subdevices on the system are audited. This setting supplements the audit settings for individual devices or subdevices. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-DEVICE-ACCESS-FAIL

Unsuccessful attempts to access all devices or subdevices on the system are audited. This setting supplements the audit settings for individual devices or subdevices. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-DEVICE-MANAGE-PASS

Successful attempts to create or manage the authorization record of a device or subdevice are audited. This setting supplements the audit settings for individual devices or subdevices. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-DEVICE-MANAGE-FAIL

Unsuccessful attempts to create or manage the authorization record for a device or subdevice are audited. This setting supplements the audit settings for individual devices or subdevices. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to specify auditing of successful attempts to access a device or subdevice, in addition to the audit settings in the individual device or subdevice authorization record:

```
=ALTER SAFEGUARD, AUDIT-DEVICE-ACCESS-PASS ALL
```

Configuring Process Auditing

You can configure systemwide auditing of all process names in addition to the audit settings in the individual process authorization records. Processes can be audited at the local level, at the remote level, or at both levels (ALL).

These Safeguard attributes relate to auditing processes:

AUDIT-PROCESS-ACCESS-PASS

Successful attempts to access all processes or subprocesses on the system are audited. This setting supplements the audit settings for individual processes or subprocesses. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-PROCESS-ACCESS-FAIL

Unsuccessful attempts to access all processes or subprocesses on the system are audited. This setting supplements the audit settings for individual processes or subprocesses. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-PROCESS-MANAGE-PASS

Successful attempts to create or manage the authorization record for a process or subprocess are audited. This setting supplements the audit settings for individual processes or subprocesses. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-PROCESS-MANAGE-FAIL

Unsuccessful attempts to create or manage the authorization record for a process or subprocess are audited. This setting supplements the audit settings for individual processes or subprocesses. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to specify auditing of all successful attempts to access a process or subprocess, in addition to the audit settings in the individual process or subprocess authorization records:

```
=ALTER SAFEGUARD, AUDIT-PROCESS-ACCESS-PASS ALL
```

Configuring Disk File Auditing

You can configure systemwide auditing of all disk volumes, subvolumes, and files in addition to the audit settings in the individual volume, subvolume, and disk-file authorization records. These objects can be audited at the local level, at the remote level, or at both levels (ALL).

These Safeguard attributes relate to auditing disk files:

AUDIT-DISKFILE-ACCESS-PASS

Successful attempts to access a volume, subvolume, or disk file are audited. This setting supplements the audit settings for these individual objects. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-DISKFILE-ACCESS-FAIL

Unsuccessful attempts to access a volume, subvolume, or disk file are audited. This setting supplements the audit settings for these individual objects. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-DISKFILE-MANAGE-PASS

Successful attempts to create or manage authorization records for a volume, subvolume, or disk file are audited. This setting supplements the individual audit settings. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-DISKFILE-MANAGE-FAIL

Unsuccessful attempts to create or manage authorization records for a volume, subvolume, or disk file are audited. This setting supplements the individual audit settings. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-DISKFILE-PRIV-LOGON

specifies conditions for auditing attempts to perform a priv logon on the system. This setting supplements the individual audit settings. The conditions can be ON or OFF. The default is OFF.

For example, to specify auditing of successful and unsuccessful local attempts to access a volume, subvolume, or disk file, in addition to the audit settings of the individual authorization records for those objects:

```
=ALTER SAFEGUARD, AUDIT-DISKFILE-ACCESS LOCAL
```

Configuring Auditing of All System Objects

You can configure systemwide auditing of all system objects in addition to the audit settings in the individual authorization records.

These Safeguard attributes relate to the auditing of system objects:

AUDIT-OBJECT-ACCESS-PASS

Successful attempts to access any system object are audited. This setting is supplements the audit settings for the individual objects. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-OBJECT-ACCESS-FAIL

Unsuccessful attempts to access any system object are audited. This setting supplements the audit settings for the individual objects. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-OBJECT-MANAGE-PASS

Successful attempts to create or manage authorization records for any system object are audited. This setting supplements the audit settings for the individual objects. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

AUDIT-OBJECT-MANAGE-FAIL

Unsuccessful attempts to create or manage authorization records for any system object are audited. This setting supplements the audit settings for the individual objects. The conditions can be ALL, NONE, LOCAL, or REMOTE. The default is NONE.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to audit all successful attempts to manage an authorization record for any system object:

```
=ALTER SAFEGUARD, AUDIT-OBJECT-MANAGE-PASS ALL
```

△ **Caution.** Configuring the Safeguard software to audit all system objects might cause system performance problems. Be sure you have adequate system resources to handle extensive auditing.

Configuring Client Auditing

You can configure the Safeguard software so that it does not accept audit records from privileged clients. If your site has no interest in client audit records, you can use this feature to reduce the quantity of audit records written to the Safeguard audit files.

These Safeguard attributes control client auditing:

`AUDIT-CLIENT-GUARDIAN`

ON specifies that the Safeguard software will accept guardian related audit records from privileged client subsystems and write those records in the Safeguard audit files. OFF specifies that the Safeguard software will not accept client guardian related audit records. The initial value is ON.

`AUDIT-CLIENT-OSS`

ON specifies that the Safeguard software will accept OSS related audit records from privileged client subsystems and write those records in the Safeguard audit files. OFF specifies that the Safeguard software will not accept client OSS related audit records. The initial value is ON.

Note. The AUDIT-CLIENT-GUARDIAN and AUDIT-CLIENT-OSS attributes are supported only on systems running G06.29 and later G-series RVUs and H06.08 and later H-series RVUs.

`AUDIT-OSS-FILTER`

indicates if user level attributes, AUDIT-USER-ACTION-PASS and AUDIT-USER-ACTION-FAIL, enable or disable OSS auditing. The AUDIT-OSS-FILTER attribute takes effect only if the Safeguard global configuration attribute AUDIT-CLIENT-OSS is enabled. The initial value is OFF.

Note. The AUDIT-OSS-FILTER attribute is supported only on systems running J06.04 and later J-series RVUs and H06.15 and later H-series RVUs.

`AUDIT-TACL-LOGOFF`

controls generation of audits for the TACL LOGOFF or TACL EXIT operations. When set to TRUE, audits for the TACL LOGOFF or TACL EXIT operations are

generated based on the value of the AUDIT-AUTHENTICATE-PASS and AUDIT-AUTHENTICATE-FAIL attributes.

When set to FALSE, audits for the TACL LOGOFF or TACL EXIT operations are generated based on the value of the AUDIT-CLIENT-GUARDIAN, AUDIT-PROCESS-ACCESS-PASS, and AUDIT-PROCESS-ACCESS-FAIL attributes. The initial value is FALSE.

Note. The AUDIT-TACL-LOGOFF attribute is supported only on systems running J06.08 and later J-series RVUs, H06.19 and later H-series RVUs, and G06.32 and later G-series RVUs.

`DYNAMIC-PROC-UPDATE`

ON specifies that the process identity attributes (AUDIT-USER-ACTION-PASS, AUDIT-USER-ACTION-FAIL, primary group, supplementary group list, and group count) are updated dynamically when the audit and group attributes of the corresponding user are modified.

The default value is OFF.

Note. The DYNAMIC-PROC-UPDATE attribute is supported only on systems running J06.10 and later J-series RVUs and H06.21 and later H-series RVUs.

For more information about client subsystem auditing, see the *Safeguard Audit Service Manual*.

Configuring Audit Exclusion of NonStop Client Events

You can filter generation of certain NonStop client audit events by using the systemwide audit exclusion parameters regardless of individual and global audit configuration settings. The filtering of audit events is based on the selected fields in the audit report.

To exclude the events from being audited, you must provide a field name and its appropriate values.

The following configuration attributes control systemwide audit exclusion:

`AUDIT-EXCLUDE-FIELD`

specifies the field name of an audit record. All NonStop client audit events containing the specified field name are not generated by the Safeguard subsystem. The default value is NONE.

Table 9-2 lists the different AUDIT-EXCLUDE-VALUES each AUDIT-EXCLUDE-FIELD can take.

Table 9-2. AUDIT-EXCLUDE-FIELDs and their corresponding values (page 1 of 5)

AUDIT-EXCLUDE-FIELD	Values for AUDIT-EXCLUDE-VALUE
Operation	● READ ● WRITE ● EXECUTE ● DELETE ● CREATE ● UPDATE ● CHANGE ● RENAME ● START ● STOP ● SUSPEND ● REVIVE ● OPEN ● CLOSE ● GRANT ● REVOKE ● PURGE ● SELECT ● INSERT ● REFERENCE ● CHANGE_OWNER ● BACK_OUT ● ONLINE_DUMP ● OTHER ● ABORT ● ADD ● ENABLE ● EXCLUDE ● INITIALIZE ● NEXT

Table 9-2. AUDIT-EXCLUDE-FIELDs and their corresponding values (page 2 of 5)

AUDIT-EXCLUDE-FIELD	Values for AUDIT-EXCLUDE-VALUE
	• RESOLVE • ACCEPT • NEXTTAPE • REJECT • RESET • SCRATCH • SET • USETAPE • DEBUG • CHANGEPRIORITY • CHANGESTEPMOM • ALTER • GIVE • LICENSE • PROGID • NEWPROCESS • SECURITY • COMPOSITE • ACCESS • DIRSEARCH • KILL • LINK • OSSRESOLVE • TRUST • TACLLOGOFF

Table 9-2. AUDIT-EXCLUDE-FIELDs and their corresponding values (page 3 of 5)

AUDIT-EXCLUDE-FIELD	Values for AUDIT-EXCLUDE-VALUE
OUTCOME	• GRANTED • DENIED • MAYBE • PASSED • FAILED • NORECORD • OTHER • PARTIAL_SUCCE • PENDING • WARNING
OBJECTTYPE	• DISKFILE • SUBVOLUME • VOLUME • DEVICE • SUBDEVICE • PROCESS • SUBPROCESS • SUBSYSTEM • COMMAND • USER • GUARDIAN_USER • SQL_TABLE • SQL_VIEW • SQL_INDEX • SQL_CATALOG • USER_RECORD • PROT_RECORD • CONROLLER • PATH

Table 9-2. AUDIT-EXCLUDE-FIELDs and their corresponding values (page 4 of 5)

AUDIT-EXCLUDE-FIELD	Values for AUDIT-EXCLUDE-VALUE
	• SQL_TABLE • SQL_VIEW • SQL_INDEX • SQL_CATALOG • USER_RECORD • PROT_RECORD • CONTROLLER • PATH • CONFIG_RECORD • SFG_CONFIG_REC • AUD_TR_CONFIG_REC • TMF_TRANSACTION • TMF_AUDITTRAIL • TMF_TAPEMEDIA • TMF_AUDITDUMP • TMF_BACKOUT • TMF_CATALOG • TMF_DUMPS • SFG_LU_RECORD • USER_REMPASS • TAPEMOUNT • TAPEVOLUME • SYSTEMDEVICE • SHAREDSEGMNT • GROUP • SFG_PROC_RECORD • DIRECTORY • FIFO • OSSDISKFILE

Table 9-2. AUDIT-EXCLUDE-FIELDs and their corresponding values (page 5 of 5)

AUDIT-EXCLUDE-FIELD	Values for AUDIT-EXCLUDE-VALUE
	• OSSFILESET • SOCKET • SYMLINK • TTY • PROCESSGROUP • OSSPROCESS
OWNERISREMOTE	• REMOTE • LOCAL • NONE • UNKNOWN

The following AUDIT-EXCLUDE-FIELD values have dynamic variable names, therefore, no enums are defined.

- OWNERUSERNAME
- OWNERUSERNUMBER
- SUBJECTUSERNAME
- SUBJECTUSERNUMBER
- SUBJECTSYSTEMNAME
- SUBJECTCREATORNAME
- SUBJECTCREATORNUMBER
- SUBJECTSYSTEMNUMBER
- SUBJECTPROCESSNAME
- SUBJECTAUTHLOCNAME
- SUBJECTTERMINALNAME
- SUBJECTAUTHLOCNUMBER
- CREATORUSERNAME
- CREATORUSERNUMBER
- CREATORSYSTEMNAME
- CREATORCREATORNAME
- CREATORCREATORNUMBER

- CREATORSYSTEMNUMBER
- CREATORPROCESSNAME
- CREATORAUTHLOCNAME
- CREATORTERMINALNAME
- CREATORAUTHLOCNUMBER
- OBJECTNAME

`AUDIT-EXCLUDE-VALUE`

specifies a set of values (up to five) for the respective field names in the AUDIT-EXCLUDE-FIELD. Combination of field names and the values determine the exclusion of NonStop client audit events. The default value is NONE.

Note. The AUDIT-EXCLUDE-FIELD and AUDIT-EXCLUDE-VALUE attributes must be used in a single command line while filtering the audit records.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to specify exclusion of audit events generated by a NonStop client subsystem when a READ operation is performed on the system:

```
ALTER SAFEGUARD, AUDIT-EXCLUDE-FIELD OPERATION, AUDIT-EXCLUDE-
VALUE READ
```

To specify exclusion of audit records generated when READ, WRITE, and EXECUTE operations are performed on the system:

```
ALTER SAFEGUARD, AUDIT-EXCLUDE-FIELD OPERATION, AUDIT-EXCLUDE-
VALUE (READ:WRITE:EXECUTE)
```

When the configuration attributes AUDIT-CLIENT-GUARDIAN and AUDIT-CLIENT-OSS are both set to OFF, the audit exclusion values set by AUDIT-EXCLUDE-FIELD and AUDIT-EXCLUDE-VALUE do not take effect.

Note. The AUDIT-EXCLUDE-FIELD and AUDIT-EXCLUDE-VALUE attributes are supported only on systems running J06.03 and later J-series RVUs, H06.14 and later H-series RVUs, and G06.32 and later G-series RVUs.

Configuring a Default Command Interpreter

When the Safeguard software controls a terminal, it automatically starts a particular command interpreter (process) at that terminal each time a user logs on successfully at the terminal. This command interpreter can also be specified in a user authentication record and in a terminal definition record. If neither of these records contains a command interpreter specification, the Safeguard software uses the command interpreter (and its associated parameters) specified in the configuration record.

These attributes specify the command interpreter to be started at Safeguard terminals:

`CI-PROG`

Specifies the name of the command interpreter's object file. The file name must be a local file name. The initial value for CI-PROG file is $SYSTEM.SYSTEM.TACL. A null entry for CI-PROG sets the value to null. If the value of CI-PROG is null and no CI-PROG is defined for the user or the terminal, a command interpreter is not started at the terminal.

`CI-LIB`

Specifies the file name of the library to be used with the command interpreter. The file name must be a local file name. The default is no library.

`CI-SWAP`

Specifies the name of the swap volume or file to be used with the command interpreter. The volume must be a local volume. The default is same volume that contains the command interpreter.

`CI-CPU`

Specifies the number of the processor in which the command interpreter is to run. The default value is ANY, which means that any processor is used.

`CI-PRI`

Specifies the priority at which the command interpreter is to run. The initial value for CI-PRI is 149. A null entry for CI-PRI sets the priority to null, and CI-PROG is started with a priority the same as that of the Safeguard $ZSMP process.

`CI-PARAM-TEXT`

Specifies the startup parameter text to be supplied to the command interpreter when it is started. The default is no parameter text.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to specify that the command interpreter is to run in CPU 3 and that $DOOFUS2 is to be used as the swap volume:

```
=ALTER SAFEGUARD, CI-CPU 3, CI-SWAP $doofus2
```

Configuring Communication With $CMON

You can set attributes in the configuration record to specify the manner in which the Safeguard software communicates with $CMON during processing at a Safeguard terminal.

These attributes relate to $CMON:

CMON

ON specifies that the Safeguard software is to communicate with the $CMON process during the following events: logon, illegal logon attempts, logoff, and newprocess of the command interpreter. If CMON is OFF, there is no communication with $CMON during these events. The initial value is OFF.

CMONTIMEOUT

Specifies the number of seconds that the Safeguard software is to wait for any $CMON operation. The default is 30 seconds.

CMONERROR

ACCEPT specifies that failures to communicate with $CMON for any reason are ignored. DENY specifies that failures to communicate with $CMON result in the current authentication being denied. The default is ACCEPT.

Use the ALTER SAFEGUARD command to change any of these attributes. For example, this command specifies that the $CMON timeout period is to be changed to 15 seconds:

```
=ALTER SAFEGUARD, CMONTIMEOUT 15
```

Configuring Logon Dialog

These attributes specify aspects of the logon dialog:

BLINDLOGON

ON specifies that passwords will not be accepted if they are typed on the same line as the user name and that they must be entered on a separate line following the password prompt. OFF specifies that passwords can be entered on the same line as the user name during logon. The initial value is OFF.

NAMELOGON

ON specifies that only a user name (group name.member name) is accepted when a user logs on. OFF specifies that a user can enter either a user name (group name.member name) or a user ID (group number,member number). The initial value is ON.

Use the ALTER SAFEGUARD command as necessary to change these configuration attributes. For example, use this command to allow the use of either user names or user IDs during logon:

```
=ALTER SAFEGUARD, NAMELOGON OFF
```

Configuring Exclusive Access at Safeguard Terminals

You can set the TERMINAL-EXCLUSIVE-ACCESS attribute so that a user who is logged on at a Safeguard terminal has exclusive access to the terminal. This attribute applies only to terminals that are controlled by the Safeguard software.

`TERMINAL-EXCLUSIVE-ACCESS`

ON specifies that access at a Safeguard terminal is exclusively reserved for the user who is currently logged on. No other user can open the terminal during the authenticated user session. OFF specifies that exclusive access is not guaranteed to a user who is logged on at a Safeguard terminal. The initial value is OFF.

Configuring Warning Mode

You can configure warning mode, which allows you to test the effectiveness of your access control lists, as described in Section 8, Warning Mode.

These attributes relate to warning mode:

`SYSTEM-WARNING-MODE`

ON specifies that warning mode is to be enabled. OFF specifies that warning mode is to be disabled. The initial value is OFF.

`OBJECT-WARNING-MODE`

ON specifies that warning mode is to be enabled. OFF specifies that warning mode is to be disabled. The initial value is OFF.

`WARNING-FALLBACK-SECURITY`

GUARDIAN specifies that warning mode is to be run with the Guardian fallback option enabled. GRANT specifies that warning mode is to be run with the Guardian fallback option disabled. The initial value is GUARDIAN.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to enable warning mode and to disable the Guardian fallback option:

```
=ALTER SAFE, WARNING-MODE ON, WARNING-FALLBACK-SECURITY GRANT
```

Configuring Persistence

Use the ADD command to configure persistence, which allows you to create protection records for disk files. The NORMAL value of this attribute is designed to preserve backward compatibility. The ALWAYS value provides access to the persistence feature.

Note. PROGID, LICENSE, TRUST, PRIV-LOGON, and CLEARONPURGE are reset for a disk file with a persistent protection record when the file is created.

This attribute relates to persistence:

`ALLOW-DISKFILE-PERSISTENT`

ALWAYS allows the creation of disk-file protection records for files that might not exist. NORMAL restricts creation of disk-file protection records to files that exist. The initial value is NORMAL.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM. For example, to enable disk file persistence:

```
=ALTER SAFE, ALLOW-DISKFILE-PERSISTENT ALWAYS
```

Configuring Attributes for Node Specific Subjects in ACLs

Use the global configuration attribute ALLOW-NODE-ID-ACL to define ACL entries containing explicit node identifiers for subjects to be consulted to determine access. The initial ALLOW-NODE-ID-ACL value is off, ignoring ACL entries containing explicit node identifiers.

This attribute relates to node identifiers:

`ALLOW-NODE-ID-ACL`

OFF ignores ACL entries containing explicit node identifiers. ON defines ACL entries containing explicit node identifiers for subjects to be consulted to determine access.

To change any of these values, issue the ALTER SAFEGUARD command from SAFECOM.

Configuring Dynamic Process Updates

Use the global configuration attribute, DYNAMIC-PROC-UPDATE, to dynamically update the process identity attributes (AUDIT-USER-ACTION-PASS, AUDIT-USER-ACTION-FAIL, primary group, supplementary group list, and group count) when the audit and group attributes of the corresponding user are modified.

The following SAFECOM commands trigger a dynamic update to the process security attributes, when the DYNAMIC-PROC-UPDATE attribute is ON:

```
SAFECOM ALTER USER/ALIAS [user/alias name] , AUDIT-USER-ACTION
LOCAL/REMOTE/ALL/NONE

SAFECOM ALTER GROUP [ group name ], MEMBER [user/alias name]
```

Note.

- HP recommends that you use the dynamic process update feature for maintenance purposes only.
- This feature is supported only on systems running on J06.10 and later J-series RVUs and H06.21 and later H-series RVUs.

10 Installation and Management

This section is intended for the security administrator or trusted user who is responsible for installing, supervising, and maintaining the Safeguard subsystem.

This section includes an overview of the Safeguard software components, procedures for installing the Safeguard subsystem, and guidelines for securing the Safeguard software.

Safeguard Components

Before you install the Safeguard software, you should have a basic understanding of its software components. The Safeguard software consists of a number of processes and security database files. The processes cooperate to manage the contents of the security database, to authenticate users attempting to log on to the system, and to authorize all attempts to access protected objects.

These Safeguard components reside on every system where the Safeguard software is installed:

- The subject database, which contains a user authentication record for every user who is authorized to use the system.
- The object database, which contains object authorization records for every disk file, disk volume, disk subvolume, device, subdevice, process, subprocess, and OBJECTTYPE that has been placed under Safeguard protection.
- SMP (Security Manager Process, with the process name $ZSMP), which is responsible for managing all changes to the subject and object databases and for authenticating user logon attempts.
- SAFECOM, the Safeguard command interpreter (SAFECOM provides an interactive command interface to the SMP).
- SMON (Security Monitor), which is responsible for authorizing all attempts to access protected objects. A separate SMON process runs in each processor in a Safeguard protected system. Each SMON is responsible for authorizing all access to objects controlled by I/O processes running in its processor and to all processes with Safeguard protected names running in its processor.

The Security Manager Process (SMP)

The SMP runs on every system on which the Safeguard software is installed. The SMP has three major responsibilities:

- Security database management
- User authentication
- Security Monitor (SMON) process management

Security Database Management

The SMP makes all changes to the subject and object databases on the local system. You make changes to the databases with SAFECOM commands. SAFECOM interprets the commands and communicates with the SMP to change the database.

When a SAFECOM user requests information about a user or a protected object, SAFECOM requests the information from the SMP. The SMP then reads the subject or object database to reply to the SAFECOM request.

The SMP also creates audit records of attempts to access the subject and object databases.

User Authentication

The SMP authenticates all user attempts to log on to the system in which it is running. When a user attempts to log on, the user's command interpreter sends a user authentication request to the SMP. The SMP reads the subject database to authenticate the logon request and then replies to the command interpreter with the results of the authentication check. The SMP also authenticates any authentication request made by an application process.

Security Monitor (SMON) Process Management

The SMP is responsible for starting all the SMON processes on the system in which it is running. When a processor fails, the SMP is responsible for restarting the SMON in that processor after the processor has been reloaded.

The Security Monitors (SMONs)

A separate SMON process runs in every processor of a system in which the Safeguard software is installed. Each SMON process is responsible for authorizing all attempts to access protected objects with primary I/O processes running in its processor. Similarly, when an attempt is made to access a running named process that is protected, the access must be authorized by the associated SMON process. The SMON processes read the object database to authorize attempts to access protected objects.

The SMON processes are also responsible for creating audit records of attempts to access the objects under their protection.

Safeguard Helper Process (SHP)

The Safeguard Helper Process (SHP) assists the SMP to identify and update process attributes when the following user attributes in the user database files are modified:

- AUDIT-USER-ACTION-PASS
- AUDIT-USER-ACTION-FAIL
- Primary group ID

- Supplementary group list
- Group count

A separate SHP process runs in every processor in a protected system. Each SHP updates the process attributes of every process in its own processor running with the user identity whose above-mentioned user attributes are changed. The SMP ensures that all SHPs are operational.

Note. SHP is supported on systems running on J06.10 and later J-series RVUs and H06.21 and later H-series RVUs.

Process Considerations for the SMP and SAFECOM

The system uses a process identification number (PIN) to identify a process. When the system creates a new process, it assigns a PIN to the process. Processes on a system running D-series or G-series RVUs can have either a high or a low PIN as:

- A low PIN ranges from 0 through 254.
- A high PIN ranges from 256 through the maximum number supported by the processor.

PIN 255 is used only for a synthetic process ID, which is described in the *Guardian Application Conversion Guide*.

By default, SAFECOM runs at low PINs. The SMONs and SMP run at high PINs. If you have a single system running D-series or G-series RVUs or a network that consists only of systems running D-series and G-series RVUs, you can optionally allow SAFECOM to run at high PINs by using the SET HIGHPIN option of Binder.

However, if you have a network of mixed systems running C-series and D-series or G-series RVUs, recompile the low PIN processes on systems running C-series RVUs using the HIGHREQUESTORS option to communicate with high PIN SMP.

Swap Space Migration Considerations

The SMON processes now require additional swap space on each processor. The space is needed for an allocated flat data segment used for cache (for temporary storage of pattern protection records) This cache improves performance considerably. The SMONs require as much as an additional 64MB of virtual memory. System administrators must ensure that enough swap space is available on each processor for this new feature.

Note. Even if CHECK-DISKFILE-PATTERN is OFF, the additional swap space is still required.

Safeguard Subsystem Management Commands

The Safeguard subsystem management commands are entered through SAFECOM. Table 10-1 on page 10-4 described them briefly. The syntax of these commands is described in detail in the *Safeguard Reference Manual*. The procedures you use to install and monitor the Safeguard software are described later in this section.

Table 10-1. Safeguard Subsystem Management Commands

Command	Description
ALTER SAFEGUARD	Changes one or more Safeguard configuration attributes. For more information, see Section 9, Configuration.
INFO SAFEGUARD	Displays the configurable Safeguard attributes. For more information, see Section 9, Configuration.
START [SAFEGUARD]	No longer functional. However, SAFECOM still accepts the command for compatibility with previous product versions. When the SMP is started, it automatically starts the SMON processes. For more information, see Starting the SMP on page 10-7.
STOP [SAFEGUARD]	Stops all the SMON and SMP processes.

General Installation Procedure

These general steps are required to set up and manage the Safeguard subsystem:

1. Install the Safeguard software. Choose the appropriate method, described in Installing the Safeguard Software on page 10-5.
2. Start the SMP process. If the Safeguard software is not installed to start automatically, you must start it manually, as described in Starting the SMP on page 10-7.
3. Configure the Safeguard subsystem and set up basic access control lists to secure sensitive components of your system. For more information, see Guidelines for Securing the Safeguard Subsystem on page 10-12.
4. Monitor the status of your system, as described in Monitoring the Safeguard Subsystem on page 10-14.

Installing the Safeguard Software

The method you use to install the Safeguard software is based on the software RVU you are running and manner in which you want the Safeguard software to be started and stopped.

- If you want the Safeguard software to run continuously from the time the system is loaded until the time it is stopped:
  - For G-series RVUs, you must use the SCF ADD command to add the Safeguard software to the Kernel subsystem and system configuration database as a persistent process.
  - For D-series RVUs, you must configure the Safeguard software in your CONFTEXT file and run SYSGEN to include it in the OSIMAGE file.
- If you want to start the Safeguard software sometime after the system is loaded and then stop it without stopping the system, use DSM/SCM to install the software according to standard installation procedures. For more information about DSM/SCM usage, see the *DSM/SCM User's Guide*. For more information about Safeguard installation instructions, see your Safeguard softdoc.

Note. Regardless of the method used to install the Safeguard software, you can make the super ID undeniable on the local system by adding the following line to the ALLPROCESSORS PARAGRAPH of the CONFTEXT file:

```
SUPER_SUPER_IS_UNDENIABLE;
```

If you add this line, the Safeguard software ignores explicit denials of access authorities for the super ID. The SUPER_SUPER_IS_UNDENIABLE parameter takes effect when the system is loaded with the OSIMAGE file that was produced from the CONFTEXT file containing this parameter. (This specification does not apply to remote nodes.)

Guardian file security settings, Safeguard ACLs, and SEEP Authorization rules are ignored if the PAID of the user is 255,255. Security checks within SAFECOM are ignored for users with a PAID of 255,255.

If SUPER.SUPER is declared undeniable. That is if a Safeguard ACL denies access to SUPER.SUPER, that denial is ignored. This applies to both aliases of SUPER.SUPER, and the SUPER.SUPER user because all the checks are done by User ID only. The authentication process is not affected if the SUPER ID is undeniable, because it is applicable for authorization checks only. The DENIABLE/UNDENIABLE setting has no effect on OSS file access.

Adding the Safeguard Software to the Kernel Subsystem (G-Series RVUs)

To add the Safeguard software to the Kernel subsystem as a persistent process, you must execute an SCF ADD PROCESS command. This command shows recommended settings for the command attributes:

```
-> ADD PROCESS $ZZKRN.#ZSMP, &
   AUTORESTART 10, &
```

```
BACKUPCPU 1, &
PRIMARYCPU 0, &
DEFAULTVOL $SYSTEM.SYSTEM, &
EXTSWAP $SWAP01, &
HIGHPIN ON, &
HOMETERM $ZHOME, &
NAME $ZSMP, &
OUTFILE $ZHOME, &
PRIORITY 198, &
PROGRAM $SYSTEM.SYSTEM.OSMP, &
SAVEABEND OFF, &
STARTMODE KERNEL or SYSTEM, &
STARTUPMSG "<BCKP-CPU>", &
STOPMODE STANDARD, &
TYPE OTHER, &
USERID SUPER.SUPER
```

Regarding the attribute values shown in the example:

- The values for NAME, PRIORITY, SAVEABEND, PROGRAM, STARTUPMSG, STOPMODE, TYPE, and USERID must be entered as shown.
- The values shown for AUTORESTART, BACKUPCPU, EXTSWAP, DEFAULTVOLUME, HIGHPIN, HOMETERM, OUTFILE, and PRIMARYCPU are recommended.
- The subvolume specified for EXTSWAP should be on the same processor pair as PRIMARYCPU and BACKUPCPU.
- If $SYSTEM.SYSTEM.OSMP is specified for PROGRAM, the OSMP program file in the current SYS*nn* subvolume is used.
- For OUTFILE and HOMETERM, $ZHOME is a good choice if $VHS is unavailable.
- For STARTMODE, specify KERNEL if you want the Safeguard software to start early in the system load process. Specify SYSTEM if you want it to start at the end of the system load.

For more information about the SCF ADD command, see the *SCF Reference Manual for the Kernel Subsystem*.

If you need to stop the Safeguard software when it is installed in this manner, you must first execute an SCF ABORT command. Then you can use the SAFECOM STOP command. To restart the Safeguard software, use the SCF START command.

Including the Safeguard Software in the OSIMAGE File (D-Series RVUs)

To configure the Safeguard software in your CONFTEXT file, you must add the Safeguard files OSMP and OSMON to the SYSTEM_PROCESS_CODE_FILES entry of the ALLPROCESSORS paragraph. The entry should contain these definitions:

```
SYSTEM_PROCESS_CODE_FILES  $dsv-vol.ZSAFEGRD.OSMON,
                           $dsv-vol.ZSAFEGRD.OSMP,
                           TANDEM^PROCESS^CODE^FILES;
```

where *$dsv-vol* is the name of the volume containing the ZSAFEGRD DSV.

If the Safeguard subsystem is included in the OSIMAGE file, it is started automatically when the system is loaded, and it cannot be stopped without stopping the system.

If you include the Safeguard software in the OSIMAGE file or start the Safeguard software as part of the CIIN file, you should keep another OSIMAGE file in a backup SYS*nn* subvolume on $SYSTEM. This OSIMAGE file should not include either the Safeguard software or a CIIN file. If necessary, you can use this backup SYSnn subvolume to recover from an inadvertent security lockout without performing a tape load.

If the Safeguard software is included in the OSIMAGE file, take these precautions to prevent auditing from being suspended during a system load:

1. Before shutting down the system, ensure that the current audit pool resides on a disk that is connected to the same processor as the $SYSTEM disk.
2. When the system load is complete, you can select a new audit pool on another disk volume if necessary.

Starting the SMP

The Safeguard subsystem must be started manually if it is not included in your OSIMAGE file. The local super ID starts the SMP by executing the OSMP file with a command interpreter RUN command. The SMP process must be named $ZSMP. Furthermore, $ZSMP should be in the same processor as $SYSTEM, and a backup processor should be specified. The SMP is always started with a priority of 198, regardless of any priority specified in the RUN command.

For example, if the Safeguard object files were stored in the current SYS*nn* subvolume on $SYSTEM, which is configured between processors 3 and 4, the local super ID starts the SMP with this TACL command:

```
2> OSMP/NAME $ZSMP, CPU 3, NOWAIT/4
```

This command is typically part of your CIIN or system startup files.

Note. Because the OSMP object program file contains PRIV code, it can only be run by the super ID. If other users are to be allowed to start the SMP, the super ID must license the OSMP and OSMON. For example:

```
VOLUME $SYSTEM.SYSnn
FUP LICENSE (OSMP, OSMON)
```

In this example, the backup SMP process is created in CPU 4. If you do not specify a backup processor, no backup process is created. Once the SMP is running, it automatically creates these files:

$SYSTEM.SAFE.GUARD	Contains object authorization records for all disk objects (volumes, subvolumes, and disk files) on $SYSTEM.
$SYSTEM.SAFE.OTHER	Contains object authorization records for all local system objects other than disk (devices, subdevices, processes, subprocesses, and OBJECTTYPEs).
$SYSTEM.SAFE.CONFIG	Contains the attributes that determine Safeguard configuration.
$SYSTEM.SAFE.CONFIGP [1]	Contains the attributes that stores the new password configuration information.
$SYSTEM.SAFE.CONFIGA	Contains audit configuration information and terminal definition records.
$SYSTEM.SAFE.A000000n	Contains the initial audit files that serve as a secondary audit pool after other audit pools are added.
$SYSTEM.SAFE.LUSERID	Contains user alias information and OSS user attributes.
$SYSTEM.SAFE.LUSERIDG	Is the alternate key file for LUSERID.
$SYSTEM.SAFE.PATGUARD	Contains diskfile patterns.
$SYSTEM.SAFE.LUSERAX	Contains password, text description and binary description records for an alias. Opens in secure mode.
$SYSTEM.SYSTEM.USERAX	Contains password, text description and binary description records for an user. Opens in secure mode.

[1] The $SYSTEM.SAFE.CONFIGP file is supported only on systems running G06.29 and later G-series RVUs and H06.06 and later H-series RVUs.

Note.

- The $SYSTEM.SAFE.SPTGUARD file is created when the SAVED-DISKFILE-PATTERN protection record is created.
- The $SYSTEM.SAFE.SPTGUARD file is supported only on systems running J06.10 and later J-series RVUs and H06.21 and later H-series RVUs.

The SMP also starts the SMON processes. It uses this naming convention for the SMON process names:

```
$ZS00    -  SMON running in CPU 0
$ZS01    -  SMON running in CPU 1
$ZS02    -  SMON running in CPU 2
$ZS03    -  SMON running in CPU 3
  .
  .
  .
$ZSnn    -  SMON running in CPU nn
```

The SMP starts all the SMON processes with a priority of 199.

Converting to the Safeguard Subsystem

When the Safeguard software is installed on a system with an existing user community, it takes over the existing USERID file. The next time each user logs on, his or her record is expanded to contain security attributes, defined as:

- OWNER is set to the user ID of the user's group manager. (For example, the OWNER attribute for a person with a user ID of 4,56 is set to 4,255.) The Safeguard software does not verify that a group manager exists. The authentication records for users who belong to a group without a group manager are owned by a nonexistent user.
- PASSWORD does not change. (The user keeps the existing logon password.)
- USER-EXPIRES is set to null. (The user's ability to log on to the system does not expire.)
- PASSWORD-MUST-CHANGE EVERY *num* DAYS is set to null. (The user's password does not expire.)
- AUDIT-ACCESS-PASS, AUDIT-ACCESS-FAIL, AUDIT-MANAGE-PASS, and AUDIT-MANAGE-FAIL are all set to NONE. (No auditing is performed.)
- REMOTEPASSWORD does not change. (All remote passwords currently defined for a user are retained.)
- DEFAULT-PROTECTION is not specified for a user's disk files. (Guardian protection applies.)

For Safeguard product versions prior to D30, HP recommends that the ADDUSER, DELUSER, and RPASSWRD program object files be deleted when the Safeguard software is installed on a system. With D30, it is no longer necessary to delete these programs because they now coordinate requests for their services through the Safeguard software.

When the Safeguard software is installed for the first time, Expand line handlers need to be restarted. This action enables the line handlers to open the LUSERID file which is created by the Safeguard subsystem to manage user alias information and OSS user attributes. If line handlers are not restarted, any access using an alias will generate security violations across nodes.

Updating the Safeguard Software

Some current Safeguard capabilities are incompatible with previous product versions of the Safeguard software, and they might cause operational difficulties during installation and operation. This is also true for the audit files, which are incompatible with previous product versions of the Safeguard software. Read this section carefully before attempting to update your system from a previous Safeguard RVU.

Updating a Previous RVU With the Safeguard Software Running

If the Safeguard software must always be running, the following steps are suggested to update your system from a previous RVU:

1. Bring all applications to an orderly shutdown, just as you would for a system load.
2. Rename the current Safeguard program files, and then load the new program files from tape. If necessary, use the old program files to establish Safeguard security for the new program files.
3. Perform all the steps normally taken to shut down a system.
4. Reset all processors, and perform a system load.
5. Start the Safeguard software normally. As the Safeguard software starts, it determines that the audit trail file names do not exist, and creates these files, with the correct size and structure. This is the key step in any changeover.
6. If Safeguard DDL files are used to build a DDL dictionary or to create precompiled Enform queries, then build a new DDL dictionary and recompile those Enform queries.

Updating a Previous RVU With the Safeguard Software Stopped

The main points to remember in replacing a previous RVU with the Safeguard software stopped:

1. Stop the Safeguard software.
2. Restore the new program files.

3. Start the Safeguard software.
4. Make a new DDL dictionary if you use one.

Returning to a Previous RVU

To return to a previous RVU, follow the steps used to change over to this RVU, but use the previous RVU's programs. Make a new DDL dictionary if you use one.

Note. Consult the softdoc for migration and fallback issues.

Guidelines for Securing the Safeguard Subsystem

After you install the Safeguard subsystem, take steps to ensure the security of its components. To do so:

1. Secure the SAFECOM program object file as necessary. If you create an access control list for SAFECOM, you can restrict the use of the command interpreter to certain users. Protecting the SAFECOM object file has no effect on users of the Safeguard Subsystem Programmatic Interface (SPI).

   To restrict the use of SAFECOM, you must add a disk file authorization record for the SAFECOM file and specify an access control list that names the qualified users. Give EXECUTE authority to each user who needs to use the command interpreter.

   Depending on your security policy, the use of SAFECOM might be unrestricted, or it might be limited to only a few qualified personnel.

   For example, if all users are expected to use Safeguard to secure their files, they must be able to execute SAFECOM. This command allows such access:

   ```
   =ADD DISKFILE $SYSTEM.SYSnn.SAFECOM, ACCESS *.* e
   ```

   If your security policy is restrictive so that the Safeguard software is to be used by only a few individuals, specify only individuals on the access control list. For example, this command provides EXECUTE authority to only users who are members of the group SECURE:

   ```
   =ADD DISKFILE $SYSTEM.SYSnn.SAFECOM, ACCESS secure.* e
   ```

   Also make sure no other copies of SAFECOM (other SYS*nn*) are secured less restrictively.

   For example, if you want a diskfile having process access records and the new process that is launched from that diskfile to inherit process access record from the same, and for the same owner of the record and disk file, this command allows such access:

   ```
   =ADD DISKFILE $SYSTEM.SYSnn.SAFECOM, PROCESS-ACCESS TEST.USER
   (R,W,C)
   ```

2. For each object type, determine which individual objects on the system are sensitive and should be protected. Some of these objects are:

   - Sensitive disk files, such as the Safeguard audit files, the USERID file, and certain files used by your applications.
   - Sensitive disk volumes and subvolumes such as the system disk ($SYSTEM) and system subvolume ($SYSTEM.SYS*nn*), as well as all important production and application disks.
   - Sensitive devices, including certain terminals, printers, or communication lines.

- Sensitive process names, including those used by the operating system, by the Safeguard software, or by your applications. For example, you might want to secure $CMON and process names associated with the spooler and Pathway monitor. You might also want to create the special NAMED and UNNAMED protection records for processes. (For more information, see the *Safeguard Reference Manual*.)

For all these objects, list the users who should be able to read, write, or create process names, devices, volumes, subvolumes, and disk files. For more information on securing objects, see the *Security Management Guide*.

Note. Do not secure the process name $ZSMP or the subprocess name $ZSMP.#ZSPI. Also, you cannot secure the process name $0 with the Safeguard software.

You need not establish an access control list for Safeguard SPI commands. These commands are subject to the same restrictions as their equivalent SAFECOM commands. For example, if you have defined a SECURITY-ADMINISTRATOR security group, only members of that group can execute the ALTER SAFEGUARD command and its equivalent SPI command ALTER SUBSYSTEM.

---

3. Create OBJECTTYPE protection records as appropriate to restrict the set of users who can add protection records for various types of objects. It is especially advisable to use the OBJECTTYPE PROCESS command to define a limited set of users can who add protection records for process names. This approach prevents general users from securing critical processes and thereby controlling those processes. For more information regarding the OBJECTTYPE commands, see Section 5, OBJECTTYPE Control.

4. Implement additional Safeguard controls for existing users with the SAFECOM ALTER USER command. For example, this command requires AUDIT.BOB to change his password every 15 days:

   ```
   =ALTER USER audit.bob, PASSWORD-MUST-CHANGE EVERY 15 DAYS
   ```

   The same controls can be implemented for new users with the ADD USER command.

5. Define the SECURITY-ADMINISTRATOR and SYSTEM-OPERATOR security groups to control the use of the audit service commands, TERMINAL commands, ALTER SAFEGUARD command, and STOP SAFEGUARD command. For more information about the security groups, see Section 6, Managing Security Groups.

6. Use ADD TERMINAL commands to add terminal definitions for those terminals to be controlled by the Safeguard software. For more information about Safeguard terminals, see Section 7, Securing Terminals.

7. If necessary, develop and install software tools to allow users who are restricted from SAFECOM to get information that they need about their Safeguard status.

Monitoring the Safeguard Subsystem

Monitoring the Safeguard subsystem comprises checking the system console for Safeguard status and internal error messages, and managing the Safeguard audit files to prevent data loss.

Safeguard Console Messages

The Safeguard subsystem reports both status messages and internal error messages on the system console.

Event messages report on events such as starting and stopping the Safeguard software, changing the Safeguard configuration, and opening a new audit file.

For a description of the Safeguard console messages, see the *Operator Messages Manual*.

Managing Safeguard Audit Files

Initially, the Safeguard software writes audit records to the audit files on $SYSTEM.SAFE. You can add other audit pools on different volumes and subvolumes, and you can choose which audit pool is to be used as the current audit pool—that is, which audit pool is to receive audit records. You can also define the next audit pool to be used when the current audit pool is filled. Because an audit pool can contain several audit files, your system might have several different volumes and subvolumes containing multiple audit files.

When the current audit file is filled, the Safeguard software automatically switches to the next available file in that audit pool. Alternatively, you can monitor usage of audit files and manually switch to the next file, or even switch to another audit pool, as necessary.

As long as unused or released audit files remain available in the current audit pool, there is no danger of audit data being lost. Even that danger is minimized if you have specified the next audit pool to be used. Therefore, part of the task of monitoring the audit service activity is to release (purge) audit files that are no longer needed so that they can be reused.

You can use the INFO AUDIT SERVICE command to determine the current audit pool and to verify that the next audit pool has been specified. For more information, see the *Safeguard Audit Service Manual*.

The Safeguard subsystem writes a message to the system console each time it switches from one audit file to another. Therefore, system console messages can also help you to determine when to extract data from a used audit file.

A SAFECOM Command Syntax

This appendix summarizes the syntax of all the SAFECOM commands. The commands are listed in alphabetic order.

SAFECOM reserved words can be abbreviated. Typically, a reserved word can be abbreviated to its first three characters unless a longer abbreviation is necessary to distinguish between similar reserved words.

The syntax notation conventions used here and throughout this manual are listed in Notation Conventions on page xiv.

For more information about syntax, including examples, see the *Safeguard Reference Manual*.

Common Syntax Elements

These syntax elements are common to many SAFECOM commands:

user-spec

can be any of:

```
group-name . member-name
group-name . *
*.*
group-num , member-num
group-num , *
* , *
```

group-name or *user-name* can contain * and ? wild-card characters.

user-list

is either:

```
net-user-spec
(net-user-spec [, net-user-spec ...])
```

net-user-spec

can be any of:

```
[\node-spec.]group-name . member-name
[\node-spec.]group-name . *
[\node-spec.]*.*
[\node-spec.]group-num , member-num
[\node-spec.]group-num , *
[\node-spec.]* , *
```

node-spec

can take any of these forms:

```
* | node-name | node-number
```

node-name

specifies the system name.

node-number

specifies the Expand node number.

sec-group-list

has the form:

```
{   sec-group-spec                              }
{ ( sec-group-spec [ , sec-group-spec ] ... ) }
```

sec-group-spec

can be any of:

```
SECURITY-ADMINISTRATOR
SYSTEM-OPERATOR
SECURITY-OSS-ADMINISTRATOR
```

Note. The SECURITY-OSS-ADMINISTRATOR security group is supported only on systems running G06.29 and later G-series RVUs and H06.08 and later H-series RVUs.

---

object-list

has the form:

```
{   object-spec                             }
{ ( object-spec [ , object-spec ] ... ) }
```

object-spec

For disk files, can be either a fully or a partially qualified disk file name, or a disk file set.

For diskfile patterns, can be a template representing a fully qualified name.

For volumes, can be either a fully or a partially qualified volume name.

For subvolumes, can be either a fully or a partially qualified subvolume name, or a subvolume set.

For devices, can be either a fully or a partially qualified device name.

For subdevices, can be either a fully or a partially qualified subdevice name.

For processes, can be either a fully or a partially qualified process name.

For subprocesses, can be either a fully or a partially qualified subprocess name.

For OBJECTTYPE, there is no *object-spec*.

object-spec can contain * and ? wild-card characters except in ADD commands for devices, subdevices, processes, and subprocesses.

object-name

is the name of an existing protected object of the same type as the *object-type* of the command; used in the LIKE clause.

terminal-name

is a fully or partially qualified device or subdevice name.

user-attribute

is any valid security attribute for users. A complete list of user attributes is given under the SET USER command.

object-attribute

is any valid security attribute for the appropriate *object-type* of the command. A complete list of object attributes is given under the SET *object-type* command.

SAFECOM Command Syntax

These diagrams show the syntax of each SAFECOM command:

```
ADD object-type  object-list [ , ]
   [ LIKE object-name | object-attribute ]
   [ , object-attribute  ] ...
```

```
ADD ALIAS alias [ , ]
   { group-name.user-name | group-num,user-num }
   [ LIKE user | user-attribute ]
   [ , user-attribute ] ...
```

```
ADD AUDIT POOL $vol.subvol [ , ]
   [ file-spec [ , file-spec ]...] [ LIKE [ $vol.subvol ] ]
```

file-spec is one of:

```
EXTENTSIZE (primary-ext [ , secondary-ext ] )
MAXEXTENTS n
MAXFILES n
AUDITCLEARONPURGE { ON | OFF }
```

Note. AUDITCLEARONPURGE is supported only on systems running J06.03 and later J-series RVUs, H06.12 and later H-series RVUs, and G06.32 and later G-series RVUs.

```
ADD EVENT-EXIT-PROCESS name  [ [ , ] exit-attribute ]

   [ , exit-attribute ] ...
```

exit-attribute specifies the name of the event-exit-process attribute to be set. The exit attributes are:

```
ENABLED { ON | OFF }
RESPONSE-TIMEOUT [ n [ SECONDS ] ]
ENABLE-AUTHENTICATION-EVENT { ON | OFF }
ENABLE-AUTHORIZATION-EVENT { ON | OFF }
ENABLE-PASSWORD-EVENT { ON | OFF }
PROG [ prog-filename ]
LIB [ lib-filename ]
SWAP [ $vol  [ .subvol.filename ] ]
PNAME [ process-name ]
CPU [ cpu-number | ANY ]
PRI [ priority ]
PARAM-TEXT [ startup-param-text ]
```

```
ADD GROUP [ NAME ] group-name [ , ] NUMBER group-num [ , ]

   [ group-attribute [ , group-attribute ] ] ...
```

group-attribute is one of:

```
OWNER [ owner-id ]
MEMBER member-list
DESCRIPTION [ text ]
OWNER-LIST [user-list]
```

Note. The attribute OWNER-LIST is supported only on systems running H06.25 and later H-series RVUs and J06.14 and later J-series RVUs.

```
ADD SECURITY-GROUP sec-group-list [ , ]

   [ LIKE sec-group-spec | sec-group-attribute ]

   [ , sec-group-attribute ] ...
```

```
ADD TERMINAL terminal-name [ , ]

   [ LIKE terminal-name | term-param ]

   [ , term-param ] ...
```

term-param is one of:

```
PROG prog-filename
LIB lib-filename
CPU cpu-number
PNAME process-name
SWAP swap-vol
PRI priority
PARAM-TEXT startup-param-text
```

```
ADD USER group-name.user-name , group-num , user-num [ , ]

   [ LIKE user | user-attribute ] [ , user-attribute ] ...
```

```
ALTER object-type  object-list [ , ]

   { LIKE object-name | object-attribute }

   [ , object-attribute ] ...
```

```
ALTER ALIAS { alias | ( alias [ , alias ] ... ) }

   [ , ] { LIKE user | user-attribute }

   [ , user-attribute ] ... [ [,] WHERE expression ]
```

```
ALTER AUDIT POOL  [ $vol.subvol ] [ , ]

   file-spec [ , file-spec ]...
```

```
ALTER AUDIT SERVICE [ , ] operating-mode

    [ , operating mode ] ...
```

operating-mode is one of:

```
WRITE-THROUGH CACHE { ON | OFF }
EOF REFRESH { ON | OFF }
RECOVERY recovery
```

recovery is one of:

```
RECYCLE [ FILES ]
SUSPEND AUDIT
DENY GRANTS
```

```
ALTER EVENT-EXIT-PROCESS  name [ , ] exit-attribute
   [ , exit-attribute ] ...
```

```
ALTER GROUP { [ NAME ] name-list | NUMBER num-list } [ , ]
   [ group-attribute [ , group-attribute ] ] ...
```

```
ALTER SAFEGUARD [ , ] attribute [ , attribute ] ...
```

```
ALTER SECURITY-GROUP sec-group-list [ , ]
   { LIKE sec-group-spec | sec-group-attribute  }
   [ , sec-group-attribute ] ...
```

```
ALTER TERMINAL terminal-name [ , ]
   { LIKE terminal-name | term-param }
   [ , term-param ] ...
```

```
ALTER USER { user-spec | ( user-spec [ , user-spec  ] ... ) }
   [ , ] { LIKE user-id | user-attribute }
   [ , user-attribute  ] ... [ [,] WHERE expression ]
```

```
ASSUME [ object-type | USER | ALIAS | TERMINAL |
   EVENT-EXIT-PROCESS ] --(but not OBJECTTYPE)--
```

```
DELETE object-type  object-list [ [ , ] WHERE option-list ]
```

WHERE *option-list* applies to disk files and diskfile-patterns only.

```
DELETE ALIAS { alias | ( alias [ , alias ] ... ) }
   [ [,] WHERE expression ]
```

```
DELETE AUDIT POOL  $vol.subvol
```

```
DELETE EVENT-EXIT-PROCESS  name
```

```
DELETE GROUP { [ NAME ] name-list | NUMBER num-list }
```

```
DELETE SECURITY-GROUP sec-group-list
```

```
DELETE TERMINAL terminal-name
```

```
DELETE USER { user-spec |( user-spec [ , user-spec  ] ... ) }

   [ [,] WHERE expression ]
```

```
DISPLAY command [ , command ] ...
```

command is one these DISPLAY commands:

```
[ AS ] COMMANDS [ ON | OFF ]
DETAIL [ ON | OFF ]
HEADERS [ ON | OFF | ONCE ]
PROMPT [ prompt-item ]
       [ ( prompt-item [ , prompt-item ] ) ... ]
USER [ AS ] { NAME | NUMBER }
WARNINGS [ ON | OFF ]
```

prompt-item can be:

```
"string"
ASSUME OBJECTTYPE
COMMAND NUMBER
CPU
DATE
END
PROCESS NAME
PROCESS NUMBER
SUBVOLUME
SYSTEM NAME
SYSTEM NUMBER
TIME
USER NAME
USER NUMBER
VOLUME
```

```
ENV [ / OUT listfile / ] [ env-parm [ , env-parm ] ... ]
```

env-parm is one of:

```
SYSTEM
VOLUME
OUT
LOG
ASSUME
```

```
WARNINGS
USER
DETAIL
AS COMMANDS
HEADERS
PROMPT
```

```
EXIT
```

```
FC [ string   ]
   [ "string" ]
   [ linenum  ]
   [ -linenum ]
```

```
FREEZE object-type  object-list[ [ , ] WHERE option-list ]
```

WHERE *option-list* applies to disk files and diskfile-patterns only.

```
FREEZE ALIAS { alias | ( alias [ , alias ] ... ) }

   [ [,] WHERE expression ]
```

```
FREEZE SECURITY-GROUP sec-group-list
```

```
FREEZE TERMINAL terminal-name
```

```
FREEZE USER { user-spec | ( user-spec [ , user-spec ] ... ) }

   [ [,] WHERE expression ]
```

```
HELP [ / OUT listfile / ]  [ keyword     ]
                           [ ALL         ]
                           [  *          ]
                           [ GRAMMAR     ]
```

The asterisk (*) provides help for all keywords except the keyword "ALL"

```
HISTORY [ lines      ]
        [ RESET LAST ]
        [ RESET ALL  ]
```

```
INFO [ / OUT listfile / ] object-type  object-list [ , ]

   [ display-option ] [ , display-option ] ...
```

display-option is one of:

```
DETAIL [ ON | OFF ]
WARNINGS [ ON | OFF ]
WHERE option-list
```

WHERE options apply to disk files and diskfile-patterns only.

WARNINGS options apply to disk files only. .

```
INFO [ / OUT listfile / ] ALIAS

   { alias | (  alias [ , alias ] ... ) }

   [ [ , ] option ] [ , option  ] ...
```

option is one of:

```
GENERAL
DETAIL
AUDIT
CI
OSS
REMOTEPASSWORD
DEFAULT-PROTECTION
GROUP
OWNER-LIST
TEXT-DESCRIPTION
WHERE expression
```

```
AUDIT POOL [ audit-trail ]
```

```
INFO AUDIT SERVICE
```

```
INFO EVENT-EXIT-PROCESS  name
```

```
INFO GROUP { [ NAME ] name-list | NUMBER num-list }

   [ , DETAIL ][, OWNER-LIST]
```

```
INFO SAFEGUARD [ [ , ] option ] [ , option  ] ...
```

option is one of:

```
GENERAL
DETAIL
AUDIT
```

```
CI
COMPARE
```

```
INFO [ / OUT listfile / ] SECURITY-GROUP [ , ] sec-group-list
   [ [ , ] DETAIL ]
```

```
INFO [ / OUT listfile / ] TERMINAL terminal-name
```

```
INFO [ / OUT listfile / ] USER
   { user-spec | (  user-spec [ , user-spec ] ... ) }
   [ [ , ] option ] [ , option  ] ...
```

option is one of:

```
GENERAL
DETAIL
AUDIT
CI
OSS
REMOTEPASSWORD
DEFAULT-PROTECTION
GROUP
OWNER-LIST
TEXT-DESCRIPTION
ALIAS
WHERE expression
```

```
LOG [ logfile ]
```

```
NEXTFILE
```

```
O[BEY] [ / OUT listfile / ] command-file
```

```
OUT [ listfile ]
```

```
RELEASE afile [ , afile ] ... [ IN $vol.subvol ]
```

afile is one of:

```
audit-file
audit-file : audit-file
```

```
RESET object-type [ [ , ] object-attribute-keyword ]

   [ , object-attribute-keyword ] ...
```

```
RESET ALIAS [ [ , ] user-attribute-keyword ]

   [ , user-attribute-keyword ] ...
```

```
RESET SECURITY-GROUP [ [ , ] sec-group-attribute-keyword ]

   [ , sec-group-attribute-keyword ] ...
```

```
RESET USER [ [ , ] user-attribute-keyword ]

   [ , user-attribute-keyword ] ...
```

```
RUN  program-file [ [ / run-option [ , run-option ] ... / ]
                  [ param-set ] ]
```

program-file

is the name of the file containing the object program to be run.

run-option

is any of these run options, which are described in the *TACL Reference Manual*:

```
CPU cpu-number
INSPECT { OFF | ON | SAVEABEND }
IN [ file-name ]
LIB [ file-name ]
MEM num-pages
NAME [ $process-name ]
NOWAIT
OUT [ list-file ]
PRI priority
TERM [\system-name.]$terminal-name
```

param-set

is a program parameter or series of parameters sent to the new process in the startup message.

```
SET  object-type [ , ] { LIKE object-name | object-attribute
}

   [ , object-attribute ] ...
```

object-attribute is one of:

```
OWNER [owner-id]
ACCESS access-spec [ ; access-spec ] ...
AUDIT-ACCESS-PASS [audit-spec]
AUDIT-ACCESS-FAIL [audit-spec]
AUDIT-MANAGE-PASS [audit-spec]
AUDIT-MANAGE-FAIL [audit-spec]
```

Disk files also have these attributes:

```
LICENSE      {ON|OFF}
PROGID       {ON|OFF}
CLEARONPURGE {ON|OFF}
PERSISTENT   {ON|OFF}
TRUST        {ME|SHARED|OFF} (H-series only)
AUDIT-PRIV-LOGON{ON|OFF}
PRIV-LOGON {ON|OFF}
```

access-spec has this form:

user-list [-] [DENY] *authority-list*

user-list is one of:

```
{   net-user-spec                          }
{ ( net-user-spec [ , net-user-spec ] ... ) }
```

authority-list is one of:

```
{   authority                        }
{ ( authority [ , authority ] ... ) }
{      *                             }
```

authority is one of:

for disk files:	R[EAD], W[RITE], E[XECUTE], P[URGE], C[REATE], O[WNER]
for diskfile-patterns	R[EAD], W[RITE], E[XECUTE], P[URGE], C[REATE], O[WNER]
for volumes and subvolumes:	R[EAD], W[RITE], E[XECUTE], P[URGE], C[REATE], O[WNER]
for processes:	R[EAD], W[RITE], C[REATE], P[URGE], O[WNER]
for subprocesses:	R[EAD], W[RITE], O[WNER]
for devices and subdevices:	R[EAD], W[RITE], O[WNER]

audit-spec is one of:

```
ALL
LOCAL
```

```
REMOTE
NONE
```

```
SET ALIAS [ , ] { LIKE alias | user-attribute }

   [ , user-attribute ] ...
```

```
SET SECURITY-GROUP [ , ]

   { LIKE sec-group-spec | sec-group-attribute }

   [ , sec-group-attribute ] ...
```

sec-group-attribute is one of:

```
OWNER [owner-id]
ACCESS access-spec [ ; access-spec ] ...
AUDIT-MANAGE-PASS [audit-spec]
AUDIT-MANAGE-FAIL [audit-spec]
```

access-spec has this form:

user-list [-] [DENY] *authority-list*

user-list is one of:

```
{   net-user-spec                          }
{ ( net-user-spec [ , net-user-spec ] ... ) }
```

authority-list is one of:

```
{   authority                        }
{ ( authority [ , authority ] ... ) }
{      *                             }
```

authority is one of:

```
E[XECUTE]
O[WNER]
```

audit-spec is one of:

```
ALL
LOCAL
REMOTE
NONE
```

```
SET USER [ , ] { LIKE user-id | user-attribute }

   [ , user-attribute ] ...
```

user-attribute is one of:

```
OWNER [owner-id]
OWNER-LIST [[-]user-list]
PASSWORD [password]
```

```
USER-EXPIRES [ date [ , time] ]
PASSWORD-MUST-CHANGE [EVERY num DAYS]
PASSWORD-EXPIRY-GRACE [num [DAYS]]
PASSWORD-EXPIRES [ date [ , time] ]
AUDIT-AUTHENTICATE-PASS [audit-spec]
AUDIT-AUTHENTICATE-FAIL [audit-spec]
AUDIT-MANAGE-PASS [audit-spec]
AUDIT-MANAGE-FAIL [audit-spec]
AUDIT-USER-ACTION-PASS [audit-spec]
AUDIT-USER-ACTION-FAIL [audit-spec]
TEXT-DESCRIPTION "[text]"
REMOTEPASSWORD \system-name remote-password
DEFAULT-PROTECTION [ obj-attr ]
                   [ ( obj-attr [ , obj-attr ] ...) ]
GUARDIAN [DEFAULT] SECURITY ["]string["]
GUARDIAN [DEFAULT] [SUB]VOLUME [\system.]$vol.subvol
INITIAL-DIRECTORY [dir-path]
INITIAL-PROGRAM [prog-path]
INITIAL-PROGTYPE [prog-type]
CI-PROG [prog-filename]
CI-LIB [lib-filename]
CI-CPU [cpu-number | ANY]
CI-NAME [process-name]
CI-SWAP [$vol.[subvol.filename]]
CI-PRI [priority]
CI-PARAM-TEXT [startup-param-text]
```

date is either *dd mmm yyyy* or *mmm dd yyyy*.

time is *hh:mm* (24-hour clock).

audit-spec is one of:

```
ALL
LOCAL
REMOTE
NONE
```

```
SHOW [ / OUT listfile / ] object-type
```

```
SHOW [ / OUT listfile / ] ALIAS
```

```
SHOW [ / OUT listfile / ] SECURITY-GROUP
```

```
SHOW [ / OUT listfile / ] USER
```

```
STOP [ SAFEGUARD ]
```

```
SYNTAX [ ONLY ] { ON | OFF }
```

```
SYSTEM [ \system-name ]
```

```
THAW object-type  object-list [ [ , ] WHERE option-list ]
```

WHERE *option-list* applies to disk files and diskfile-patterns only.

```
THAW ALIAS { alias | ( alias [ , alias ] ... ) }

   [ [,] WHERE expression ]
```

```
THAW SECURITY-GROUP sec-group-list
```

```
THAW TERMINAL terminal-name
```

```
THAW USER { user-spec | ( user-spec [ , user-spec ] ... ) }

   [ [,] WHERE expression ]
```

```
VOLUME [ $volume           ]
       [ $volume.subvolume ]
       [         subvolume ]
```

```
? [ string   ]
  [ "string" ]
  [ linenum  ]
  [ -linenum ]
```

```
!  [ string   ]
   [ "string" ]
   [ linenum  ]
   [ -linenum ]
```

Note. The TEXT-DESCRIPTION attribute is supported only on systems running G06.27 and later G-series RVUs and H06.06 and later H-series RVUs.

Note. The attributes, AUDIT-PRIV-LOGON and PRIV-LOGON, are supported only on systems running H06.11 and later H-series RVUs and G06.32 and later G-series RVUs.

SAFECOM Command Syntax

Index

A

ACCESS authorities
- for all objects 4-2
- for devices and subdevices 4-4
- for disk volumes 4-3
- for OBJECTTYPE records 5-6

Access control lists
- device and subdevice priority 9-17
- for terminals 7-1
- process and subprocess priority 9-17
- testing 8-1
- volume, subvolume and disk-file priority 9-19

ADD ALIAS command 2-41
ADD DEVICE command 4-1, 4-4
ADD GROUP command 3-2
ADD OBJECTTYPE command 5-6
ADD SECURITY-GROUP command 6-4
ADD TERMINAL command 7-3
ADD USER command 2-12
ADD VOLUME command 4-1, 4-3
Adding a device to the Safeguard database 4-4
Adding a subdevice to the Safeguard database 4-4
Adding a volume to the Safeguard database 4-3
Adding an OBJECTTYPE record 5-7
Adding users 2-6, 2-12
ADDUSER program 2-28, 10-9
Administrative group 2-5, 3-1, 3-3, 3-4
Alias
- adding 2-40
- altering 2-41
- as an object type 5-8
- defined 2-40

ALTER ALIAS command 2-41
ALTER DEVICE command 4-1
ALTER GROUP command 3-3
ALTER OBJECTTYPE command 5-6
ALTER SAFEGUARD command 2-2, 6-2, 9-1, 10-4
ALTER SECURITY-GROUP command 6-7
ALTER TERMINAL command 7-4
ALTER USER command 2-12
ALTER VOLUME command 4-1
Attributes
- OBJECTTYPE 5-4
- user security 2-7

AUDIT 9-27, 9-33
Audit attributes
- for an OBJECTTYPE 5-10
- for user security 2-7

Audit files 10-14
Auditing
- for a user ID 2-25
- for an OBJECTTYPE 5-10

AUDIT-AUTHENTICATE-FAIL attribute 2-7, 9-22
AUDIT-AUTHENTICATE-PASS attribute 2-7, 9-22
AUDIT-CLIENT-GUARDIAN 9-4
AUDIT-CLIENT-OSS 9-4
AUDIT-CLIENT-SERVICE attribute 9-26
AUDIT-DISKFILE-PRIV-LOGON 9-25
AUDIT-EXCLUDE-FIELD 9-27
AUDIT-EXCLUDE-VALUE 9-33
AUDIT-MANAGE-FAIL attribute 2-7
AUDIT-MANAGE-PASS attribute 2-7
AUDIT-USER-ACTION-FAIL attribute 2-7
AUDIT-USER-ACTION-PASS attribute 2-7
authorization record 6-5, 6-6
Automatic group deletion 3-6
AUTO-DELETE flag 3-6

B

BLINDLOGON attribute 9-35

C

Changing a password 2-21, 2-22
Changing the owner of a user record 2-16, 2-17
CIIN file 10-7, 10-8
CI-CPU attribute 2-7
CI-LIB attribute 2-7
CI-NAME attribute 2-7
CI-PARAM-TEXT attribute 2-7
CI-PRI attribute 2-7
CI-PROG attribute 2-7
CI-PROG Safeguard attribute 9-33
CI-SWAP attribute 2-7
CMON attribute 9-35
Command interpreter specification
  for a terminal 7-2
  for a user 2-37
  for Safeguard configuration 9-33
  precedence 7-2
Command syntax (SAFECOM) A-3
Commands
  for device security 4-1
  for network users 2-28
  for OBJECTTYPE control 5-6
  for Safeguard management 10-4
  for user security 2-12
  for volume security 4-1
Components of Safeguard 10-1
Configuration attributes
  default values 9-2
  for client auditing 9-26
  for default command interpreter 9-33
  for device control 9-16
  for disk file control 9-18
  for dynamic process updates 9-37
  for logon dialog 9-35
  for node specific subjects 9-37
  for password control 9-6
  for persistence 9-37
  for process control 9-17
  for systemwide auditing of all objects 9-25
  for systemwide device auditing 9-22
  for systemwide disk-file auditing 9-24
  for systemwide process auditing 9-23
  for systemwide user auditing 9-21
  for terminal exclusive access 9-36
  for user authentication 9-5
  for warning mode 9-36
  for $CMON communication 9-34
CONFTEXT file 10-5, 10-7
Console messages 10-14
Controlling aliases as an object type 5-8
Controlling an entire object type 5-7
Controlling groups as an object type 5-8
Controlling logon dialog 7-2
Controlling users as an object type 5-8
Controlling who can add an object type 5-9

D

Database 10-1
Default protection for disk files 2-7
  default access control list 2-35
  default auditing 2-36
  default ownership 2-36
  eliminating 2-37
  Guardian 2-38
  specifying 2-34
DEFAULT-PROTECTION attribute 2-7, 2-34
Defining security groups 6-1
Defining user groups 2-5, 3-1, 3-3, 3-4
DELETE DEVICE command 4-1
DELETE GROUP command 3-6
DELETE OBJECTTYPE command 5-6
DELETE TERMINAL command 7-5
DELETE USER command 2-12, 2-26
DELETE VOLUME command 4-1
Deleting user groups 2-26
Deleting users 2-26

DELUSER program 2-28, 10-9
DETAIL option
of INFO OBJECTTYPE command 5-11
of INFO USER command 2-14
Device security commands 4-1
Devices
adding to the Safeguard database 4-4
valid ACCESS authorities 4-4
DYNAMIC-PROC-UPDATE 9-5, 9-27

E

Effective group ID 2-34
Establishing a network of users 2-27
Establishing a user community 2-5
Establishing network users 2-30
Exclusive access at Safeguard terminals 7-1, 9-36
Expiration date for users 2-17

F

FREEZE DEVICE command 4-1
FREEZE OBJECTTYPE command 5-6
FREEZE SECURITY-GROUP command 6-10
FREEZE TERMINAL command 7-5
FREEZE USER command 2-12, 2-25
FREEZE VOLUME command 4-1

G

Grace period 2-23, 7-2, 9-6
GRANT option 8-2, 9-36
Group
administrative 2-5, 3-1, 3-3, 3-4
as an object type 5-8
commands 3-1, 3-3, 3-4
file-sharing 2-5, 3-1, 3-3, 3-4
list 2-34
security 6-1
Guardian
default security 2-7, 2-38, 8-2
default volume 2-7, 2-38
fallback option 8-2, 9-36
GUARDIAN DEFAULT SECURITY attribute 2-7, 2-39
GUARDIAN DEFAULT VOLUME attribute 2-7, 2-39

I

Identifying network users 2-28
INFO DEVICE command 4-1
INFO GROUP command 3-2
INFO OBJECTTYPE
command 5-6
DETAIL option 5-11
INFO SAFEGUARD command 10-4
INFO SECURITY-GROUP command 6-4
INFO TERMINAL command 7-3
INFO USER
CI option 2-38
command 2-12
DEFAULT-PROTECTION option 2-35
GENERAL option 2-20
INFO VOLUME command 4-1
INITIAL-DIRECTORY attribute 2-11
INITIAL-PROGRAM attribute 2-11
INITIAL-PROGTYPE attribute 2-11

L

Login name 2-34
Logon dialog 7-2, 9-35

M

Managing network users
with SAFECOM commands 2-28
with standard security 2-28
Managing Security Groups 6-1 6-1
Managing the audit files 10-14
Monitoring Safeguard status 10-14

N

NAMED process protection records 9-18
NAMELOGON attribute 9-35
Network users
- aliases as 2-33
- defined 2-27
- establishing 2-30
- granting access to objects 2-29
- identifying 2-28
- managing with SAFECOM commands 2-28
- managing with standard security 2-28
- remote passwords for 2-29

O

Object database 10-1
OBJECTTYPE attributes 5-4
OBJECTTYPE auditing 5-10
OBJECTTYPE commands 5-6
OBJECTTYPE DEVICE 5-5
OBJECTTYPE DISKFILE 5-5
OBJECTTYPE OBJECTTYPE 5-5, 5-9
OBJECTTYPE PROCESS 5-5
OBJECTTYPE SUBDEVICE 5-5
OBJECTTYPE SUBPROCESS 5-5, 5-8
OBJECTTYPE SUBVOLUME 5-5
OBJECTTYPE USER 5-5, 5-8
OBJECTTYPE VOLUME 5-5
operations 6-1
OSIMAGE file 10-5, 10-7
OWNER attribute for user authentication record 2-7
OWNER authority 6-10, 6-11
Owner of user record
- capabilities of 2-7
- changing 2-16, 2-17

OWNER-LIST attribute 2-7

P

PASSWORD 9-4, 9-7
Password 9-7
- changing at logon 7-2
- changing with PASSWORD program 2-20
- compatibility mode 9-6
- encryption 9-6
- expiration 2-21, 9-7
- expiration grace 2-23, 9-6
- immediate expiration 2-23
- maximum length 9-7
- minimum length 9-6
- permission to change 9-7
- requiring periodic change 2-20

PASSWORD attribute 2-7
PASSWORD program 2-21
PASSWORD-ALGORITHM 9-4
PASSWORD-COMPATIBILITY-MODE 9-6
PASSWORD-DETAIL-DETAIL 9-14
PASSWORD-EXPIRES attribute 2-7, 2-23
PASSWORD-EXPIRES date 2-21, 9-15
PASSWORD-EXPIRY-GRACE attribute 2-7, 2-23, 9-7
PASSWORD-LOWERCASE-REQUIRED 9-8
PASSWORD-MAXIMUM-LENGTH 9-7
PASSWORD-MAY-CHANGE attribute 2-21, 9-15
PASSWORD-MIN-QUALITY-REQUIRED 9-9
PASSWORD-MUST-CHANGE attribute 2-7, 2-20
PASSWORD-NUMERIC-REQUIRED 9-8
PASSWORD-SPACES-ALLOWED 9-14
PASSWORD-SPECIALCHAR-REQUIRED 9-9
PASSWORD-UPPERCASE-REQUIRED 9-7
Planning for security 1-3
Policy, security 1-2

Previous RVU, returning to 10-11
PRIMARY-GROUP attribute 2-11
Process stop modes 8-3
Protecting an entire object type 5-7
Protecting an object 4-2

R

Remote passwords
converting to Safeguard protection 10-9
for network users 2-29
REMOTEPASSWORD attribute 2-7
RESET DEVICE command 4-1
RESET OBJECTTYPE command 5-6
RESET USER command 2-12
RESET VOLUME command 4-1
RESET-BINARY-DESCRIPTION attribute 2-9
Returning to a previous RVU 10-11
RPASSWRD program 2-28, 10-9

S

SAFECOM 10-1
command syntax A-3
commands for network users 2-28
Safeguard management commands 10-4
securing the SAFECOM program file 10-12
security commands 4-1
syntax elements A-1
user security commands 2-12
Safeguard
components 10-1
management commands 10-4
Safeguard console messages 10-14
Safeguard Helper Process 10-2
Safeguard management commands 10-4
Safeguard software, updating to a new product version 10-10
Safeguard status, monitoring 10-14
Securing an entire object type 5-7
Securing devices 4-4
Securing disk volumes 4-3
Securing subdevices 4-4
Securing terminals 7-1
Securing the Safeguard software, guidelines for 10-12
Security administrator 1-3
Security group, creating 6-1
Security Manager Process (SMP)
defined 10-1
starting the SMP 10-7
Security Monitor (SMON) 10-1, 10-2
Security officer 1-3
Security planning 1-3
Security policy
importance of 1-2
types of 1-3
SECURITY-ADMINISTRATOR group 6-1
SECURITY-OSS-ADMINISTRATOR group 6-1
SECUTITY-OSS-ADMINISTRATOR
privileges over the normal users 6-1
Sensitive objects 10-12
SET DEVICE command 4-1
SET OBJECTTYPE command 5-6
SET USER command 2-12
SET VOLUME command 4-1
SHOW DEVICE command 4-1
SHOW OBJECTTYPE command 5-6
SHOW USER command 2-12
SHOW VOLUME command 4-1
SMON (Security Monitor) 10-1
SMON (Security Monitor) Process 10-2
SMP (Security Manager Process)
defined 10-1
starting the SMP 10-7
Special considerations
for devices 4-4
for subdevices 4-4

for volumes 4-3
Specifying auditing for a user ID 2-25
Standard security programs 2-28
START SAFEGUARD command 10-4
Starting the SMP 10-7
STATIC-FAILED-LOGON-RESET 2-11
STOP SAFEGUARD command 6-2, 10-4
Subdevices
adding to the Safeguard database 4-4
valid ACCESS authorities 4-4
Subject database 10-1
Super ID
denial of authorities 5-8, 5-9
restricting authority 1-2
undeniable 10-5
Systemwide auditing
for all objects 9-25
for devices 9-22
for disk files 9-24
for processes 9-23
for subvolumes 9-24
for user-authentication 9-21
for volumes 9-24
supplementing individual settings 9-21
SYSTEM-OPERATOR group 6-1

T

Temporary access to system 2-17
TERMINAL commands 2-2, 7-1, 7-2
TERMINAL-EXCLUSIVE-ACCESS attribute 9-36
THAW DEVICE command 4-1
THAW OBJECTTYPE command 5-6
THAW SECURITY-GROUP command 6-10
THAW TERMINAL command 7-5
THAW USER command 2-12, 2-25
THAW VOLUME command 4-1

U

UNNAMED process protection records 9-18
Updating the Safeguard software 10-10
User alias
adding 2-40
altering 2-41
defined 2-40
User community, establishing 2-5
User groups
administrative 2-5, 3-1, 3-3, 3-4
defining 2-5
deleting 2-26
file sharing 3-1, 3-3, 3-4
User name at logon 9-35
User Security Attributes note 2-7
User security commands 2-12
Users
adding 2-6
as an object type 5-8
deleting 2-26
establishing network users 2-28
freezing 2-24
removing deleted 2-26
setting password expiration 2-21
temporary 2-17
thawing 2-24
transferring ownership 2-17
USER-EXPIRES
attribute 2-7
date 2-17
USER-EXPIRES date 2-17

V

Volume security commands 4-1
Volumes
adding to the Safeguard database 4-3
valid ACCESS authorities 4-3

W

Warning mode 8-1
WARNING-MODE Safeguard attribute 9-36

Special Characters

$CMON 9-34
$SYSTEM.SAFE.A000000n 10-8
$SYSTEM.SAFE.CONFIG 10-8
$SYSTEM.SAFE.CONFIGA 10-8
$SYSTEM.SAFE.CONFIGP 10-8
$SYSTEM.SAFE.GUARD 10-8
$SYSTEM.SAFE.LUSERID 10-8
$SYSTEM.SAFE.LUSERIDG 10-8
$SYSTEM.SAFE.OTHER 10-8
$SYSTEM.SAFE.PATGUARD 10-8
$SYSTEM.SAFE.SPTGUARD 10-9
$ZSMP
process name for SMP 10-7
securing the Security Manager Process 10-12