Quantum.

SANtricity 11.30 Maintaining a Storage Array Using CLI

StorNext QD7000

Firmware 8.30.xx.xx

SANtricity 11.30 Maintaining a Storage Array Using CLI, 6-68580-01 Rev A, April 2017 Product of USA.

Quantum Corporation provides this publication "as is" without warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability or fitness for a particular purpose. Quantum Corporation may revise this publication from time to time without notice.

COPYRIGHT STATEMENT

TRADEMARK STATEMENT

Preface

> **Note:** The 8.30.xx.xx firmware (Lehigh) is used in the QD7000 (E5600, Titan RAID controller, only). Refer to the NetApp to Quantum Naming Decoder section for additional information.

This section provides the following information:

- Audience
- Prerequisites
- NetApp to Quantum Naming Decoder
- Product Safety Statements
- Contacts
- Comments
- Quantum Global Services

Audience

This manual is intended for storage customers and technicians.

Prerequisites

Prerequisites for installing and using this product include knowledge of:

- Servers and computer networks

- Network administration

- Storage system installation and configuration

- Storage area network (SAN) management and direct attach storage (DAS)

- Fibre Channel (FC) and Ethernet protocols

NetApp to Quantum Naming Decoder

Use Table 1 to correlate the NetApp product nomenclature to the equivalent Quantum-storage naming conventions.

Table 1 Product Nomenclature

E-Series NetApp Product	Quantum-Storage	Description
Controller-Drive Tray	Base System	Quantum uses Base System when referring to a drive tray with the RAID controllers.
Drive Tray	Expansion Unit	Quantum uses Expansion Unit when referring to a drive tray with the environmental services modules (ESMs).
E5600 (Code Name: Titan)	RAID Controller	Four 16Gb/s FC SFP+ host ports
E5500 (Code Name: Soyuz)	RAID Controller	Four 16Gb/s FC SFP+ host ports
E5400 (Code Name: Pikes Peak)	RAID Controller	Four 8Gb/s FC SFP+ host ports
DE6600 (Code Name: Wembley)	4U 60-drive enclosure	Sixty 3.5 inch disk drives

E-Series NetApp Product	Quantum-Storage	Description
E5560 or E5660 (DE6600 4U drive enclosure with E5500 or E5600 RAID controllers)	Quantum StorNext QD7000	
E5460 (DE6600 4U drive enclosure with E5400 RAID controllers)	Quantum StorNext QD6000	
E5424 (DE5600 24-drive 2U drive enclosure (Code Name: Camden with E5400 RAID controllers)	Quantum StorNext QS2400	
E5412 (DE1600 12-drive 2U drive enclosure (Code Name: Ebbets with E5400 RAID controllers)	Quantum StorNext QS1200	

Product Safety Statements

Quantum will not be held liable for damage arising from unauthorized use of the product. The user assumes all risk in this aspect.

This unit is engineered and manufactured to meet all safety and regulatory requirements. Be aware that improper use may result in bodily injury, damage to the equipment, or interference with other equipment.

WARNING: Before operating this product, read all instructions and warnings in this document and in the system, safety, and regulatory guide.

警告 在使用本产品之前，请先阅读本文档及系统、安全和法规信息指南中所有的说明和警告信息。

警告 操作本產品前，請先閱讀本文件及系統、安全與法規資訊指南中的指示與警告說明。

ADVERSAL Læs alle instruktioner og advarsler i dette dokument og i *Vejledning om system-sikkerheds- og lovgivningsoplysninger*, før produktet betjenes.

AVERTISSEMENT Avant d'utiliser ce produit, lisez la totalité des instructions et avertissements de ce document et du *Guide d'informations sur le système, la sécurité et la réglementation*.

HINWEIS Lesen Sie vor der Verwendung dieses Produkts alle Anweisungen und Warnhinweise in diesem Dokument und im System, Safety, and Regulatory Information Guide (Info-Handbuch: System, Sicherheit und Richtlinien).

אזהרה לפני ההפעלה של מוצר זה, קרא את כל ההוראות והאזהרות הכלולות במסמך זה וכן במדריך *מידע בנושאי מערכת, בטיחות ותקינה*

| 警告 | この製品を使用する前に、本文書、および『システム、安全、規制に関する情報ガイド』に記載しているすべての警告と指示をお読みください。 |

| 경고 | 이 제품을 작동하기 전에 이 문서 및 시스템, 안전, 및 규제 정보 안내서에 수록된 모든 지침과 경고 표지를 숙지하십시오. |

| ПРЕДУПРЕЖДЕНИЕ | Перед началом эксплуатации данного устройства ознакомьтесь во всеми инструкциями и предупреждениями, приведенными в данном документе и в *Справочном руководстве по устройству, технике безопасности и действующим нормативам*. |

| ADVERTENCIA | Antes de utilizar este producto, lea todas las instrucciones y advertencias en este documento y en la Guia informativa sobre sistema, seguridad y normas. |

| VARNING | Läs alla anvisningar och varningar i detta dokument och i *System, säkerhet och krav från myndigheter - Informationshandbok* innan denna produkt tas i bruk. |

Contacts

For information about contacting Quantum, including Quantum office locations, go to:

http://www.quantum.com/aboutus/contactus/index.aspx

Comments

To provide comments or feedback about this document, or about other Quantum technical publications, send e-mail to:

doc-comments@quantum.com

Quantum Global Services

Accelerate service issue resolution with these exclusive Quantum StorageCare services:

- **Service and Support Website** - Register products, license software, browse Quantum Learning courses, check backup software and operating system support, and locate manuals, FAQs, firmware downloads, product updates and more in one convenient location. Benefit today at:

  http://www.quantum.com/serviceandsupport/get-help/index.aspx#contact-support

- **eSupport** - Submit online service requests, update contact information, add attachments, and receive status updates via email. Online Service accounts are free from Quantum. That account can also be used to access Quantum's Knowledge Base, a comprehensive repository of product support information. Get started at:

  http://www.quantum.com/customercenter/

For further assistance, or if training is desired, contact the Quantum Customer Support Center:

North America	1-800-284-5101 (toll free)
	+1-720-249-5700
EMEA	+800-7826-8888 (toll free)
	+49-6131-324-185
APAC	+800-7826-8887 (toll free)
	+603-7953-3010

For worldwide support:

http://www.quantum.com/serviceandsupport/get-help/index.aspx#contact-support

Maintaining a Storage Array using the Command Line Interface

NetApp®

Contents

Deciding whether to use this guide

This guide describes how to maintain an E-Series storage array using the command line interface (CLI). Using the CLI, you can run commands from an operating system prompt or from a script engine window in the SANtricity Storage Manager Enterprise Management Window (EMW).

Use this guide if you want to maintain a storage array using the CLI.

This guide is based on the following assumptions:

- Your storage system has been successfully installed.

- SANtricity Storage Manager has been successfully installed.

Where to Find the Latest Information About the Product

You can find information about the latest version of the product, including new features and fixed issues, and a link to the latest documentation at the following address: *NetApp E-Series and EF-Series Systems Documentation Center*.

Related information

NetApp E-Series and EF-Series Systems Documentation Center
SANtricity 11.30 Command Line Interface and Script Commands Programming Guide
SANtricity 11.30 Configuring a Storage Array Using the Command Line Interface

About the command line interface

The command line interface (CLI) is a software application that provides a way for installers, developers, and engineers to configure and monitor storage arrays. Using the CLI, you can run commands from an operating system prompt, such as the DOS `C:` prompt, a Linux operating system path, or a Solaris operating system path.

Each command performs a specific action for managing a storage array or returning information about the status of a storage array. You can enter individual commands, or you can run script files when you need to perform operations more than once. For example, you can run script files when you want to install the same configuration on several storage arrays. The CLI enables you to load a script file from a disk and run the script file. The CLI provides a way to run storage management commands on more than one network storage array. You can use the CLI both in installation sites and in development environments.

The CLI gives you direct access to a script engine that is a utility in the SANtricity Storage Manager software (also referred to as the storage management software). The script engine runs commands that configure and manage the storage arrays. The script engine reads the commands, or runs a script file, from the command line and performs the operations instructed by the commands.

> **Note:** You can also access the script engine by using the Enterprise Management Window in the storage management software. If you access the script engine by using the Enterprise Management Window, you can edit or run script commands on only one storage array in the script window. You can open a script window for each storage array in your configuration and run commands in each window. By using the CLI, you can run commands on more than one storage array from a single command line.

You can use the command line interface to perform these actions:

- Directly access the script engine and run script commands

- Create script command batch files to be run on multiple storage arrays when you need to install the same configuration on different storage arrays

- Run script commands on an in-band managed storage array, an out-of-band managed storage array, or a combination of both

- Show configuration information about the network storage arrays

- Add storage arrays to and remove storage arrays from the management domain

- Perform automatic discovery of all of the storage arrays that are attached to the local subnet

- Add or delete Simple Network Management Protocol (SNMP) trap destinations and email alert notifications

- Specify the mail server and sender email address or SNMP server for alert notifications

- Show the alert notification settings for storage arrays that are currently configured in the Enterprise Management Window

- Direct the output to a standard command line display or to a named file

Structure of a CLI command

The CLI commands are in the form of a command wrapper and elements embedded into the wrapper. A CLI command consists of these elements:

- A command wrapper identified by the term `SMcli`

- The storage array identifier

- Terminals that define the operation to be performed

- Script commands

The CLI command wrapper is a shell that identifies storage array controllers, embeds operational terminals, embeds script commands, and passes these values to the script engine.

All CLI commands have the following structure:

```
SMcli storageArray terminal script-commands;
```

- `SMcli` invokes the command line interface.

- `storageArray` is the name or the IP address of the storage array.

- `terminal` are CLI values that define the environment and the purpose for the command.

- `script-commands` are one or more script commands or the name of a script file that contains script commands. (The script commands configure and manage the storage array.)

If you enter an incomplete or inaccurate `SMcli` string that does not have the correct syntax, parameter names, options, or terminals, the script engine returns usage information.

Interactive mode

If you enter `SMcli` and a storage array name but do not specify CLI parameters, script commands, or a script file, the command line interface runs in interactive mode. Interactive mode lets you run individual commands without prefixing the commands with `SMcli`.

In interactive mode, you can enter a single command, view the results, and enter the next command without typing the complete `SMcli` string. Interactive mode is useful for determining configuration errors and quickly testing configuration changes.

To end an interactive mode session, type the operating system-specific command for terminating a program, such as **Control-C** on the UNIX operating system or the Windows operating system. Typing the termination command (**Control-C**) while in interactive mode turns off interactive mode and returns operation of the command prompt to an input mode that requires you to type the complete `SMcli` string.

Command line parameters

Parameter	Definition
host-name-or-IP-address	Specifies either the host name or the Internet Protocol (IP) address (*xxx.xxx.xxx.xxx*) of an in-band managed storage array or an out-of-band managed storage array. • If you are managing a storage array by using a host through in-band storage management, you must use the -n parameter or the -w parameter if more than one storage array is connected to the host. • If you are managing a storage array by using out-of-band storage management through the Ethernet connection on each controller, you must specify the *host-name-or-IP-address* of the controllers. • If you have previously configured a storage array in the Enterprise Management Window, you can specify the storage array by its user-supplied name by using the -n parameter. • If you have previously configured a storage array in the Enterprise Management Window, you can specify the storage array by its World Wide Identifier (WWID) by using the -w parameter.
-A	Adds a storage array to the configuration file. If you do not follow the -A parameter with a *host-name-or-IP-address*, auto-discovery scans the local subnet for storage arrays.
-a	Adds a Simple Network Management Protocol (SNMP) trap destination or an email address alert destination. • When you add an SNMP trap destination, the SNMP community is automatically defined as the community name for the trap, and the host is the IP address or Domain Name Server (DNS) host name of the system to which the trap should be sent. • When you add an email address for an alert destination, the email-address is the email address to which you want the alert message to be sent. **Restriction:** This command line option is obsolete for the E2800 storage array. Use the RESTful API, SANtricity System Manager, or see *Using curl commands with the E2800 storage array* on page 20.
-c	Indicates that you are entering one or more script commands to run on the specified storage array. End each command with a semicolon (;). You cannot place more than one -c parameter on the same command line. You can include more than one script command after the -c parameter.
-d	Shows the contents of the script configuration file. The file content has this format: *storage-system-name host-name1 host-name2*
-e	Runs the commands without performing a syntax check first.
-F (uppercase)	Specifies the email address from which all alerts will be sent.

Parameter	Definition
`-f` (lowercase)	Specifies a file name that contains script commands that you want to run on the specified storage array. The `-f` parameter is similar to the `-c` parameter in that both parameters are intended for running script commands. The `-c` parameter runs individual script commands. The `-f` parameter runs a file of script commands. By default, any errors that are encountered when running the script commands in a file are ignored, and the file continues to run. To override this behavior, use the `set session errorAction=stop` command in the script file.
`-g`	Specifies an ASCII file that contains email sender contact information that will be included in all email alert notifications. The CLI assumes that the ASCII file is text only, without delimiters or any expected format. Do not use the `-g` parameter if a `userdata.txt` file exists.
`-h`	Specifies the host name that is running the SNMP agent to which the storage array is connected. Use the `-h` parameter with these parameters: • `-a` • `-x`
`-I` (uppercase)	Specifies the type of information to be included in the email alert notifications. You can select these values: • `eventOnly` – Only the event information is included in the email. • `profile` – The event and array profile information is included in the email. You can specify the frequency for the email deliveries using the `-q` parameter.
`-i` (lowercase)	Shows the IP address of the known storage arrays. Use the `-i` parameter with the `-d` parameter. The file content has this format: `storage-system-name IP-address1 IPaddress2`
`-m`	Specifies the host name or the IP address of the email server from which email alert notifications are sent. **Restriction:** This command line option is obsolete for the E2800 storage array. Use the RESTful API, SANtricity System Manager, or see *Using curl commands with the E2800 storage array* on page 20.
`-n`	Specifies the name of the storage array on which you want to run the script commands. This name is optional when you use a `host-name-or-IP-address`. If you are using the in-band method for managing the storage array, you must use the `-n` parameter if more than one storage array is connected to the host at the specified address. The storage array name is required when the `host-name-or-IP-address` is not used. The name of the storage array that is configured for use in the Enterprise Management Window (that is, the name is listed in the configuration file) must not be a duplicate name of any other configured storage array.

Parameter	Definition
-o	Specifies a file name for all output text that is a result of running the script commands. Use the -o parameter with these parameters: • -c • -f If you do not specify an output file, the output text goes to standard output (stdout). All output from commands that are not script commands is sent to stdout, regardless of whether this parameter is set.
-p	Defines the password for the storage array on which you want to run commands. A password is not necessary if a password has not been set on the storage array. If you set a monitor password for the storage array, the use of the -p parameter is mandatory. Users cannot run any of the non-destructive commands such as the show commands.
-q	Specifies the frequency that you want to receive event notifications and the type of information returned in the event notifications. An email alert notification containing at least the basic event information is always generated for every critical event. These values are valid for the -q parameter: • everyEvent – Information is returned with every email alert notification. • 2 – Information is returned no more than once every two hours. • 4 – Information is returned no more than once every four hours. • 8 – Information is returned no more than once every eight hours. • 12 – Information is returned no more than once every 12 hours. • 24 – Information is returned no more than once every 24 hours. Using the -I parameter you can specify the type of information in the email alert notifications. • If you set the -I parameter to eventOnly , the only valid value for the -q parameter is everyEvent . • If you set the -I parameter to either the profile value or the supportBundle value, this information is included with the emails with the frequency specified by the -q parameter.
-quick	Reduces the amount of time that is required to run a single-line operation. An example of a single-line operation is the recreate snapshot volume command. This parameter reduces time by not running background processes for the duration of the command. Do not use this parameter for operations that involve more than one single-line operation. Extensive use of this command can overrun the controller with more commands than the controller can process, which causes operational failure. Also, status updates and configuration updates that are collected usually from background processes will not be available to the CLI. This parameter causes operations that depend on background information to fail.

Parameter	Definition
-R (uppercase)	Defines the user role for the password. The roles can be either: • admin – The user has privilege to change the storage array configuration. • monitor – The user has privilege to view the storage array configuration, but cannot make changes. The -R parameter is valid only when used with the –p parameter, which specifies that you define a password for a storage array. The -R parameter is required only if the dual password feature is enabled on the storage array. The -R parameter is not necessary under these conditions: • The dual password feature is not enabled on the storage array. • Only one admin role is set and the monitor role is not set for the storage array.
-S (uppercase)	Suppresses informational messages describing the command progress that appear when you run script commands. (Suppressing informational messages is also called silent mode.) This parameter suppresses these messages: • Performing syntax check • Syntax check complete • Executing script • Script execution complete • SMcli completed successfully
-s (lowercase)	Shows the alert settings in the configuration file when used with the -d parameter. **Restriction:** This command line option is obsolete for the E2800 storage array. Use the RESTful API, SANtricity System Manager, or see *Using curl commands with the E2800 storage array* on page 20.
-v	Shows the current global status of the known devices in a configuration file when used with the -d parameter.
-w	Specifies the WWID of the storage array. This parameter is an alternate to the -n parameter. Use the -w parameter with the -d parameter to show the WWIDs of the known storage arrays. The file content has this format: *storage-system-name world-wide-ID IP-address1 IP-address2*
-X (uppercase)	Deletes a storage array from a configuration.

Parameter	Definition
-x (lowercase)	Removes an SNMP trap destination or an email address alert destination. The *community* is the SNMP community name for the trap, and the *host* is the IP address or DNS host name of the system to which you want the trap sent. **Restriction:** This command line option is obsolete for the E2800 storage array. Use the RESTful API, SANtricity System Manager, or see *Using curl commands with the E2800 storage array* on page 20.
-?	Shows usage information about the CLI commands.

CLI command wrapper syntax

General syntax forms of the CLI command wrappers are listed in this section. The conventions used in the CLI command wrapper syntax are listed in the following table.

Convention	Definition
a | b	Alternative ("a" or "b")
italicized-words	Needs user input to fulfill a parameter (a response to a variable)
[...] (square brackets)	Zero or one occurrence (square brackets are also used as a delimiter for some command parameters)
{ ... } (curly braces)	Zero or more occurrences
(a | b | c)	Choose only one of the alternatives

Attention: To run all of the CLI commands you must have administrator privileges. Some CLI commands will run without administrator privileges. Many commands, however, will not run. If the CLI command does not run because you do not have correct privileges, the CLI returns an exit code of 12.

Examples

The following examples demonstrate the command line parameters described in *Command line parameters* on page 8.

```
SMcli host-name-or-IP-address [host-name-or-IP-address] [-c "command;
{command2};"] [-n storage-system-name | -w wwID] [-o outputfile] [-p
password] [-R (admin | monitor)] [-e] [-S] [-quick]
```

```
SMcli host-name-or-IP-address [hostname-or-IP-address] [-f scriptfile] [-
n storage-system-name | -w wwID] [-o outputfile] [-p password] [-R
(admin | monitor)] [-e] [-S] [-quick]
```

```
SMcli (-n  storage-system-name | -w wwID) [-c "command; {command2};"] [-
o outputfile] [-p password] [-R (admin | monitor)] [-e] [-S] [-quick]
```

```
SMcli (-n storage-system-name  -w wwID) [-f scriptfile] [-o outputfile]
[-R (admin | monitor)] [-p password] [-e] [-S] [-quick]
```

```
SMcli -a email: email-address [host-name-or-IP-address1 [host-name-or-IP-
address2]] [-n storage-system-name | -w wwID | -h host-name] [-I
information-to-include] [-q frequency] [-S]
```

Restriction: The -a command line option is not supported for the E2800 storage array. Use the RESTful API for script support and the SANtricity System Manager for a graphical user interface.

```
SMcli -x email: email-address [host-name-or-IP-address1 [host-name-or-IP-
address2]] [-n storage-system-name | -w wwID | -h host-name] [-S]
```

Restriction: The -x command line option is not supported for the E2800 storage array. Use the RESTful API for script support and the SANtricity System Manager for a graphical user interface.

```
SMcli (-a | -x) trap: community, host-name-or-IP-address [host-name-or-
IP-address1 [host-name-or-IP-address2]] [-n storage-system-name | -w
wwID | -h host-name] [-S]
```

Restriction: The -a and -x command line options are not supported for the E2800 storage array. Use the RESTful API for script support and the SANtricity System Manager for a graphical user interface.

```
SMcli -d [-w] [-i] [-s] [-v] [-S]
```

Restriction: The -s command line option is not supported for the E2800 storage array. Use the RESTful API for script support and the SANtricity System Manager for a graphical user interface.

```
SMcli -m host-name-or-IP-address -F email-address [-g contactInfoFile] [-
S]
```

```
SMcli -A [host-name-or-IP-address [host-name-or-IP-address]] [-S]
```

```
SMcli -X (-n storage-system-name | -w wwID | -h host-name)
```

```
SMcli -?
```

Naming conventions

- Names can have a maximum of 30 characters.

- You can use any combination of alphanumeric characters, hyphens, and underscores for the names of the following components:

  ◦ Storage arrays

  ◦ Host groups

  ◦ Hosts

  ◦ Volume groups

  ◦ Volumes

  ◦ HBA host ports

- You must use unique names. If you do not use unique names, the controller firmware returns an error.

- If the name contains more than one word, hyphens, or underscores, enclose the name in double quotation marks (" "). In some usages, you must also surround the name with square brackets ([ ]). The description of each parameter indicates whether you need to enclose a parameter in double quotation marks, square brackets, or both.

- The name character string cannot contain a new line.

- On Windows operating systems, you must enclose the name between two backslashes (\\) in addition to other delimiters. For example, the following name is used in a command that runs under a Windows operating system:

  ```
  [\"Engineering\"]
  ```

- For a UNIX operating system and, when used in a script file, the name appears as in the following example:

  ```
  ["Engineering"]
  ```

- When you enter a World Wide Identifier (WWID) of an HBA host port, some usages require that you surround the WWID with double quotation marks. In other uses, you must surround the WWID with angle brackets (<>). The description of the WWID parameter indicates whether you need to enclose the WWID in double quotation marks or angle brackets.

Entering numerical names

When the storage management software automatically configures a storage array, the storage management software assigns names that consist of numerical characters. Names that consist only of numerical characters are valid names. Numerical character names, however, must be treated differently than names that start with alphabetic characters.

- Names that are only numbers, such as 1 or 2

- Names that start with a number, such as 1Disk or 32Volume

- `[\"1\"]`

- `[\"1Disk\"]`

> **Note:** If you have any doubt as to the validity of a name, use both double quotation marks and square brackets. Using both makes sure that the name works and does not cause any processing issues.

Formatting CLI commands

Double quotation marks (" ") that are used as part of a name or label require special consideration when you run the CLI commands and the script commands on a Windows operating system.

When double quotation marks (" ") are part of a name or value, you must insert a backslash (\) before each double quotation mark character. For example:

```
-c "set storageArray userLabel=\"Engineering\";"
```

In this example, "Engineering" is the storage array name. A second example is:

```
-n \"My\"_Array
```

In this example, "My"_Array is the name of the storage array.

You cannot use double quotation marks (" ") as part of a character string (also called string literal) within a script command. For example, you cannot enter the following string to set the storage array name to "Finance" Array:

```
-c "set storageArray userLabel=\"\"Finance\"Array\";"
```

In the Linux operating system and the Solaris operating system, the delimiters around names or labels are single quotation marks (' '). The UNIX versions of the previous examples are as follows:

```
-c 'set storageArray userLabel="Engineering";'
```

```
-n "My"_Array
```

In a Windows operating system, if you do not use double quotation marks (" ") around a name, you must insert a caret (^) before each special script character. Special characters are ^, |, <, and >.

Insert a caret before each special script character when used with the terminals -n, -o, -f, and -p. For example, to specify storage array CLI>CLIENT, enter this string:

```
-n CLI^>CLIENT
```

Insert one caret (^) before each special script character when used within a string literal in a script command. For example, to change the name of a storage array to FINANCE_|_PAYROLL, enter the following string:

```
-c "set storageArray userLabel=\"FINANCE_^|_PAYROLL\";"
```

Formatting rules for script commands

Syntax unique to a specific script command is explained in the Notes section at the end of each script command description.

Case sensitivity – The script commands are not case sensitive. You can type the script commands in lowercase, uppercase, or mixed case. (In the following command descriptions, mixed case is used as an aid to reading the command names and understanding the purpose of the command.)

Spaces – You must enter spaces in the script commands as they are shown in the command descriptions.

Square brackets – Square brackets are used in two ways:

*   As part of the command syntax.

*   To indicate that the parameters are optional. The description of each parameter tells you if you need to enclose a parameter value in square brackets.

Parentheses – Parentheses shown in the command syntax enclose specific choices for a parameter. That is, if you want to use the parameter, you must enter one of the values enclosed in parentheses. Generally, you do not include parentheses in a script command; however, in some instances, when you enter lists, you must enclose the list in parentheses. Such a list might be a list of tray ID values and slot ID values. The description of each parameter tells you if you need to enclose a parameter value in parentheses.

Vertical bars – Vertical bars in a script command indicate "or" and separate the valid values for the parameter. For example, the syntax for the `raidLevel` parameter in the command description appears as follows:

```
raidLevel=(0 | 1 | 3 | 5 | 6)
```

To use the `raidLevel` parameter to set RAID level 5, enter this value:

```
raidLevel=5
```

Drive locations – The CLI commands that identify drive locations support both high-capacity drive trays and low-capacity drive trays. A high-capacity drive tray has drawers that hold the drives. The drawers slide out of the drive tray to provide access to the drives. A low-capacity drive tray does not have drawers. For a high-capacity drive tray, you must specify the identifier (ID) of the drive tray, the ID of the drawer, and the ID of the slot in which a drive resides. For a low-capacity drive tray, you need only specify the ID of the drive tray and the ID of the slot in which a drive resides. For a low-capacity drive tray, an alternative method for identifying a location for a drive is to specify the ID of the drive tray, set the ID of the drawer to 0, and specify the ID of the slot in which a drive resides. Separate the ID values with a comma. If you enter more than one set of ID values, separate each set of values with a space. Enclose the set of values in parentheses. For example:

```
(1,1 1,2 1,3 1,4 2,1 2,2 2,3 2,4)
```

or, for a high-capacity drive tray, this example:

```
(1,1,1 1,2,2 1,3,3 1,4,4 2,1,1 2,2,2 2,3,3 2,4,4)
```

Italicized terms – Italicized terms in the command indicate a value or information that you need to provide. For example, when you encounter the italicized term:

```
numberOfDrives
```

Replace the italicized term with a value for the number of drives that you want to include with the script command.

Semicolon – Script commands must end with a semicolon (`;`). You can enter more than one script command on the command line or in a script file. For example, a semicolon is used to separate each script command in the following script file.

```
create volume drives=(0,2 0,3 1,4 1,5 2,6 2,7) raidLevel=5
userLabel="v1" capacity=2gb owner=a;
create volume volumeGroup=2 userLabel="v2" capacity=1gb owner=b;
create volume volumeGroup=2 userLabel="v3" capacity=1gb owner=a;
```

```
create volume drives=(0,4 0,5 1,6 1,7 2,8 2,9) raidLevel=5
userLabel="v4" capacity=2gb owner=b;
create volume volumeGroup=3 userLabel="v5" capacity=1gb owner=a;
create volume volumeGroup=3 userLabel="v6" capacity=1gb owner=b;
```

Formatting CLI commands in Windows PowerShell

The Windows PowerShell is an interactive and scripting shell that provides access to command-line tools. The Windows PowerShell improves upon the Windows Command Prompt with a more robust set of commands and scripting capability. You can run all of the CLI and script commands in the Windows PowerShell; however, the Windows PowerShell has some unique formatting requirements. The requirements are these:

- Begin all SMcli commands with a period and a forward slash (./)

- SMcli wrapper must be identified as an executable command with the .exe extension (SMcli.exe)

- Enclose the script command in single quotation marks (' ')

- Double quotation marks that are part of a name, file path, or value must have a backslash before each double quotation mark character (\")

The following is an example of a CLI command to create a storage array name in the Windows Powershell. Note the use of the single quotation marks as delimiters for the script command and the backslash double quotation marks around the storage array name, identified as `userLabel` in the command syntax.

```
PS C:\...\StorageManager\client> ./SMcli.exe 123.45.67.88 123.45.67.89
-c 'set storageArray userLabel=\"Engineering\";'
```

The following is an example of a CLI command to enable a premium feature in the Windows Powershell. Note the use of the backslash double quotation marks before the file path to the premium feature key.

```
PS C:\...\StorageManager\client> ./SMcli.exe 123.45.67.88 123.45.67.89
-c 'enable storageArray feature file=\"C:\licenseKey.bin\";'
```

In the previous examples both upper case letters and lower case letters are used. This use is to help make clear how the commands are formatted. The Windows PowerShell is, however, not case sensitive and does not require the use specific cases.

Usage examples

This example shows how to change the name of a storage array. The original name of the storage array is Payroll_Array. The new name is Finance_Array.

Windows operating system:

```
SMcli ICTSANT -n "Payroll_Array" -c "set storageArray userLabel=
\"Finance_Array\";"
```

UNIX operating system:

```
SMcli ICTSANT -n 'Payroll_Array' -c 'set storageArray
userLabel="Finance_Array";'
```

This example shows how to delete an existing volume and create a new volume on a storage array. The existing volume name is Stocks_<_Bonds. The new volume name is Finance. The controller host names are finance1 and finance2. The storage array is protected, requiring the password TestArray.

Windows operating system:

```
SMcli finance1 finance2 -c "set session password=\"TestArray\";
delete volume [\"Stocks_^<_Bonds\"];
create volume driveCount[3] RAIDLEVEL=3 capacity=10GB userLabel=\"Finance
\";
show storageArray healthStatus;"
```

UNIX operating system:

```
SMcli finance1 finance2 -c 'set session password="TestArray";
delete volume ["Stocks_<Bonds"];
create volume driveCount[3] RAIDLEVEL=3 capacity=10GB
userLabel="Finance";
show storageArray healthStatus;'
```

This example shows how to run commands in a script file named `scriptfile.scr` on a storage array named Example. The `-e` terminal causes the file to run without checking the syntax. Running a script file without checking the syntax lets the file run more quickly; however, the file might not run correctly because the syntax for a command might be incorrect.

```
SMcli -n Example -f scriptfile.scr -e
```

This example shows how to run commands in a script file named `scriptfile.scr` on a storage array named Example. In this example, the storage array is protected by the password MyArray. Output, as a result of commands in the script file, goes to file `output.txt` .

Windows operating system:

```
SMcli -n Example -f scriptfile.scr -p "My_Array" -o output.txt
```

UNIX operating system:

```
SMcli -n Example -f scriptfile.scr -p 'My_Array' -o output.txt
```

This example shows how to show all of the storage arrays in the current configuration. The command in this example returns the host name of each storage array.

```
SMcli -d
```

If you want to know the IP address of each storage array in the configuration, add the -i terminal to the command.

```
SMcli -d -i
```

Exit status

This table lists the exit statuses that might be returned and the meaning of each status.

Status Value	Meaning
0	The command terminated without an error.
1	The command terminated with an error. Information about the error also appears.
2	The script file does not exist.
3	An error occurred while opening an output file.
4	A storage array was not at the specified address.
5	Addresses specify different storage arrays.
6	A storage array name does not exist for the host agent that is connected.
7	The storage array name was not at the specified address.
8	The storage array name was not unique.
9	The storage array name was not in the configuration file.
10	A management class does not exist for the storage array.
11	A storage array was not found in the configuration file.
12	An internal error occurred. This exit status indicates that you do not have privileges for running a CLI command from the command line. You must have administrator privileges to run all of the CLI commands from a command line.
13	Invalid script syntax was found.
14	The controller was unable to communicate with the storage array.
15	A duplicate argument was entered.
16	An execution error occurred.
17	A host was not at the specified address.
18	The WWID was not in the configuration file.
19	The WWID was not at the address.
20	An unknown IP address was specified.
21	The Event Monitor configuration file was corrupted.
22	The storage array was unable to communicate with the Event Monitor.
23	The controller was unable to write alert settings.
24	The wrong organizer node was specified.
25	The command was not available.

Status Value	Meaning
26	The device was not in the configuration file.
27	An error occurred while updating the configuration file.
28	An unknown host error occurred.
29	The sender contact information file was not found.
30	The sender contact information file could not be read.
31	The `userdata.txt` file exists.
32	An invalid `-I` value in the email alert notification was specified.
33	An invalid `-f` value in the email alert notification was specified.
34	The `-r` option is not supported anymore.
35	Invalid alert severity specified.
36	The operation needs either the Administrator or Monitor password to be set.
37	The operation cannot be completed because an invalid Monitor password was entered.
38	The operation cannot be completed because an invalid Administrator password was entered.
39	The password provided is exceeding the character limit.
40	The `-R` monitor is not supported for this array. Use a valid role and retry the operation.
41	An error occurred while writing to or reading from the AutoSupport configuration file. Please retry this operation again.
42	Host address or mail server address is incorrect.

Using curl commands with the E2800 storage array

The E2800 storage array contains an embedded web server. You can use the curl tool on the command line for getting or sending files from the array, using the URL. In cases where an SMcli command is not supported on the E2800 storage array, the curl command line is a convenient alternative.

Curl is available for download from *curl: The curl project*.

> **Note:** Because the E2800 storage array uses the TLSv2 protocol, curl defaults to this during protocol negotiations. If the version of curl being used does not support this protocol, the communication will fail.

> **Important:** These commands operate on a single E2800 storage array. For the E2700, E5600 and EF560, an enabled Event Monitor on a management station performs alert and AutoSupport configuration tasks for every storage array in the management domain.

Useful curl command line options

The following table describes a few useful options when sending commands to the E2800 storage array. For a complete list of curl options, see the curl documentation online using the link in the Related references section.

Note: If you want the parameter to contain spaces, you must enclose the entire parameter within double quotes ("). Within those quotes, you specify a quote as \".

Note: You must specify options and their arguments on the same line.

Note: Wherever you see angle brackets "< >" in the examples or descriptions below, substitute the appropriate value and do not include the brackets.

Option	Description
`d`	This option is used to pass data to the E2800 storage array. **Note:** POST data must be URL-encoded.

`H`	There are several types of headers used in HTTP requests. Two types of headers used for managing the E2800 storage array are:
	
	• Accept header - the HTTP client uses this header to tell the E2800 storage array what content types are acceptable.
	
	• Content-Type header - the storage array sends a response that signifies the content type of the returned data.
	
	When you send data to the E2800 storage array (for example in a POST request), you need both headers. In this case the Content-Type header signifies the content type of the posted data.
	
	To tell the E2800 storage array which type of content to return, set the Content-Type header using the `-H` option. For example, to receive JSON data:
	
	`-H "Accept: application/json"`
	
	To tell the E2800 storage array which type of content to return, set the Content-Type header using the `-H` option. For example, to receive JSON data:
	
	`-H "Content-Type: application/json"`
	
	Note: Because curl allows multiple -H options on a request, you can specify more than one header in different formats.

`k`	The `k` option is required to allow curl to run against a system that has a self-signed certificate and to suppress certificate verification. **Note:** For systems that have a real CA signed certificate, this option is not required because curl performs certificate authentication against a provider.
`O`	This option allows results to be saved to a file. The file is saved to the current working directory with the same filename as the filename on the server.
`o`	This option allows results to be saved to a file with a custom filename. Follow the option with the filename of your choice.
`u`	Specify the user name and password as arguments to this option, as shown in the example below: `-u <rw:password>` **Note:** The user name for all E-Series storage arrays is rw.
`v`	This option is the verbose option, useful for debugging.

Option	Description
x	This option changes the HTTP request method. Available methods for HTTP 1.1 are: • GET • HEAD • POST • PUT • DELETE • CONNECT • OPTIONS • TRACE

Retrieve alert configuration information

The following example command gets alert configuration data from the E2800 controller.

```
curl  -k -u <rw:password> -H "Accept: application/json" "https://<array
IP address>/devmgr/v2/storage-systems/1/device-alerts"
```

Post new alert configuration data

The following example command passes alert configuration data to the E2800 controller as part of
the request body. Note the use of the -d option and the JSON data sent using keyalue pairs.

```
curl  -k -u <rw:password> -H "Accept: application/json" -H "Content-
Type: application/json" -d @- "https://<array IP address>/devmgr/v2/
storage-systems/1/device-alerts" <<EOF
 {
   "alertingEnabled": true,
   "emailServerAddress": "smtp.corp.mydomain.com",
   "emailSenderAddress": "test@mydomain.com",
   "sendAdditionalContactInformation": true,
   "additionalContactInformation": "Additional information",
   "recipientEmailAddresses": [
     "mytestaddr@somedomain.com"
   ]
 }
 EOF
```

Perform a test against the email recipients configured for alerts

The following example command tests the alert configuration by sending email to configured
recipients.

```
curl -X POST  -k -u <rw:password> -H "Accept: application/json" -H
"Content-Type: application/json"  "https://<array IP address>/devmgr/v2/
storage-systems/1/device-alerts/alert-email-test"
```

Retrieve AutoSupport configuration data

The following example command gets AutoSupport configuration data from the E2800 storage array in JSON format.

```
curl  -k -u <rw:password> -H "Accept: application/json" "https://<array
IP address>/devmgr/v2/device-asup"
```

Post new AutoSupport configuration data

The following example command posts new AutoSupport configuration data to the E2800 storage array in JSON format.

```
curl  -k -u <rw:password> -H "Accept: application/json" -H "Content-
Type: application/json" -d @- "https://<array IP address>/devmgr/v2/
device-asup" <<EOF
 {
   "asupCapable": true,
   "onDemandCapable": true,
   "asupEnabled": true,
   "onDemandEnabled": true,
   "remoteDiagsEnabled": true,
   "delivery": {
     "method": "https",
     "routingType": "direct",
     "proxyHost": null,
     "proxyPort": 0,
     "proxyrw": null,
     "proxyPassword": null,
     "proxyScript": null,
     "mailRelayServer": null,
     "mailSenderAddress": null
   },
   "destinationAddress": "https://testbed.netapp.com/put/AsupPut/",
   "schedule": {
     "dailyMinTime": 0,
     "dailyMaxTime": 1439,
     "weeklyMinTime": 0,
     "weeklyMaxTime": 1439,
     "daysOfWeek": []
   }
 }
 EOF
```

Retrieve information about the web server

The following example command retrieves information about the web server running on an E2800 controller.

```
curl -k -u <rw:password> -H "Accept: application/json" "https://<array
IP address>/devmgr/utils/about"
```

Restart the web server on an E2800 controller

The following example command restarts the web server on an E2800 controller.

```
curl -X POST  -k -u <rw:password> "https://<array IP address>/devmgr/v2/
restart"
```

Retrieve the available AutoSupport log file list

The following example command gets the list of available AutoSupport log files from the E2800 controller.

```
curl -k -u <rw:password> -H "Accept: application/json" "https://<array
IP address>/devmgr/v2/device-asup/logs"
```

Retrieve a specific AutoSupport log file downloaded to the local host

The following example command downloads an AutoSupport log file from the E2800 controller to the local host.

```
curl -k -u <rw:password> -O "https://<array IP address>/devmgr/v2/device-
asup/logs/ASUPMessages.0"
```

Override to allow HTTP traffic to the E2800 storage array

The following example command overrides the HTTPS default to allow HTTP traffic.

> **Note:** Run this command on both controllers.

> **Note:** If the value in the payload is set to true all traffic will be redirected to HTTPS. If the value is false, HTTP traffic is allowed to the web server.

> **Note:** Restart the web server after running this command.

> **Note:** This command may return an error with an error code of 204. This is a no content reply which can be ignored.

```
curl -k -u <rw:password> -H "Accept: application/json" -H "Content-Type:
application/json" -d @- "https://<array IP address>/devmgr/v2/override"
<<EOF
{
"key":"redirect.http",
"value":"true"
}
EOF
```

Retrieve the syslog configuration

The following example command gets the syslog configuration from the E2800 storage array.

```
curl -k -u <rw:password> -H "Accept: application/json" "https://<array
IP address>/devmgr/v2/storage-systems/1/device-alerts/alert-syslog"
```

Post a new syslog configuration

The following example command posts a new syslog configuration to the E2800 storage array in JSON format.

> **Note:** To indicate to curl to read the format from `stdin` write@- .

```
curl -k -u <rw:password> -H "Accept: application/json" -H "Content-Type:
application/json" -d @- "https://<array IP address>/devmgr/v2/storage-
systems/1/device-alerts/alert-syslog" <<EOF
 {
  "syslogReceivers":
  [
    {
```

```
        "serverName":"testserver.yourdomain.com",
        "portNumber":514
      }
  ],
      "defaultFacility":3,
      "defaultTag":"StorageArray"
 }
 EOF
```

Test the syslog configuration

The following example command sends a test alert to the configured syslog server addresses.

```
curl -X POST  -k -u <rw:password> -H "Accept: application/json" -H
"Content-Type: application/json"  "https://<array IP address>/devmgr/v2/
storage-systems/1/device-alerts/alert-syslog-test"
```

Get a list of failure log entries with details turned on

The following example command gets a list of failure log entries from the E2800 storage array with
all available details.

```
curl -k -u <rw:password> -H "Accept: application/json" "https://<array
IP address>/devmgr/v2/storage-systems/1/failures?details=true"
```

About the script commands

You can use the script commands to configure and manage a storage array. The script commands are distinct from the command line interface (CLI) command wrappers. You can enter individual script commands, or you can run a file of script commands. When you enter an individual script command, you embed the script command in a CLI command wrapper. When you run a file of script commands, you embed the file name in the CLI command wrapper. The script commands are processed by a script engine that performs the following functions:

- Verifies the command syntax

- Interprets the commands

- Converts the commands to the appropriate protocol-compliant commands

- Passes the commands to the storage array

At the storage array, the storage array controllers run the script commands.

The script engine and the script commands support the storage array configuration and management operations that are listed in the following table.

Operation	Activities
General storage array configuration	Resetting a configuration to defaults, labeling, checking the health status, setting the time of day, clearing the Event Log, and setting the media scan rate
Volume configuration and volume group configuration	Creating, deleting, and setting the reconstruction priority control, labeling, setting drive composition when creating volumes, setting the segment size, and setting the media scan control
Drive configuration	Assigning hot spares
Controller configuration	Defining volume ownership, changing mode settings, defining network settings, and setting host channel IDs
Firmware management	Downloading controller firmware, the environmental services module (ESM) firmware, and the drive firmware
NVSRAM configuration	Downloading and modifying the user configuration region at the bit level and the byte level, showing nonvolatile static random access memory (NVSRAM) values
Cache configuration	Controlling all cache parameters, both at the storage array level and the individual volume level
Product identification	Retrieving the tray profile display data
Battery management	Setting the battery installation date

Structure of a script command

All script commands have the following structure:

```
command operand-data (statement-data)
```

- *command* identifies the action to be performed.

- *operand-data* represents the objects associated with a storage array that you want to configure or manage.

- *statement-data* provides the information needed to perform the command.

The syntax for *operand-data* has the following structure:

```
(object-type | all object-types | [qualifier] (object-type [identifier]
(object-type [identifier] | object-types [identifier-list]))
```

An object can be identified in four ways:

- Object type – Use when the command is not referencing a specific object.

- `all` parameter prefix – Use when the command is referencing all of the objects of the specified type in the storage array (for example, `allVolumes`).

- Square brackets – Use when performing a command on a specific object to identify the object (for example, `volume [engineering]`).

- A list of identifiers – Use to specify a subset of objects. Enclose the object identifiers in square brackets (for example, `volumes [sales engineering marketing]`).

A qualifier is required if you want to include additional information to describe the objects.

The object type and the identifiers that are associated with each object type are listed in this table.

Object Type	Identifier
`controller`	a or b
`drive`	For enclosures with drawers, use tray ID, drawer ID, and slot ID. Alternatively, just tray ID and slot ID.
`replacementDrive`	For enclosures with drawers, use tray ID, drawer ID, and slot ID. Alternatively, just tray ID and slot ID.
`driveChannel`	Drive channel identifier
`host`	User label
`hostChannel`	Host channel identifier
`hostGroup`	User label
`hostPort`	User label
`iscsiInitiator`	User label or iSCSI Qualified Name (IQN)
`iscsiTarget`	User label or IQN
`storageArray`	Not applicable
`tray`	Tray ID
`volume`	Volume user label or volume World Wide Identifier (WWID) (`set` command only)
`volumeCopy`	Target volume user label and, optionally, the source volume user label
`volumeGroup`	User label Valid characters are alphanumeric, a hyphen, and an underscore.

Statement data is in the form of:

- Parameter = value (such as `raidLevel=5`)

- Parameter-name (such as `batteryInstallDate`)

- Operation-name (such as `redundancyCheck`)

A user-defined entry (such as user label) is called a variable. In the syntax, it is shown in italic (such as `trayID` or `volumeGroupName`).

Synopsis of the script commands

Because you can use the script commands to define and manage the different aspects of a storage array (such as host topology, drive configuration, controller configuration, volume definitions, and volume group definitions), the actual number of commands is extensive. The commands, however, fall into general categories that are reused when you apply the commands to configure or maintain a storage array. The following table lists the general form of the script commands and a definition of each command.

Syntax	Description
`accept object` `{statement-data}`	Performs the pending operation.
`activate object` `{statement-data}`	Sets up the environment so that an operation can take place or performs the operation if the environment is already set up correctly.
`autoConfigure storageArray` `{statement-data}`	Automatically creates a configuration that is based on the parameters that are specified in the command.
`check object` `{statement-data}`	Starts an operation to report on errors in the object, which is a synchronous operation.
`clear object` `{statement-data}`	Discards the contents of some attributes of an object. This operation is destructive and cannot be reversed.
`create object` `{statement-data}`	Creates an object of the specified type.
`deactivate object` `{statement-data}`	Removes the environment for an operation.
`delete object`	Deletes a previously created object.
`diagnose object` `{statement-data}`	Runs a test and shows the results.

Syntax	Description
`disable object {statement-data}`	Prevents a feature from operating.
`download object {statement-data}`	Transfers data to the storage array or to the hardware that is associated with the storage array.
`enable object {statement-data}`	Sets a feature to operate.
`load object {statement-data}`	Transfers data to the storage array or to the hardware that is associated with the storage array. This command is functionally similar to the `download` command.
`recopy object {statement-data}`	Restarts a volume copy operation by using an existing volume copy pair. You can change the parameters before the operation is restarted.
`recover object {statement-data}`	Re-creates an object from saved configuration data and the statement parameters. (This command is similar to the `create` command.)
`remove object {statement-data}`	Removes a relationship between objects.
`repair object {statement-data}`	Repairs errors found by the `check` command.
`replace object {statement-data}`	The specified object replaces an existing object in the storage array.
`reset object {statement-data}`	Returns the hardware or an object to an initial state.
`resume object`	Starts a suspended operation. The operation starts where it left off when it was suspended.
`revive object`	Forces the object from the Failed state to the Optimal state. Use this command only as part of an error recovery procedure.
`save object {statement-data}`	Writes information about the object to a file.

Syntax	Description
`set object` `{statement-data}`	Changes object attributes. All changes are completed when the command returns.
`show object` `{statement-data}`	Shows information about the object.
`start object` `{statement-data}`	Starts an asynchronous operation. You can stop some operations after they have started. You can query the progress of some operations.
`stop object` `{statement-data}`	Stops an asynchronous operation.
`suspend object` `{statement-data}`	Stops an operation. You can then restart the suspended operation, and it continues from the point where it was suspended.
`validate object` `{statement-data}`	Validates a security key.

Recurring syntax elements

Recurring syntax elements are a general category of parameters and options that you can use in the script commands. The following table lists the conventions used in the recurring syntax elements.

Convention	Definition
`a	b`	Alternative ("a" or "b")
`italicized-words`	Needs user input to fulfill a parameter (a response to a variable)
`[ ... ]` (square brackets)	Zero or one occurrence (square brackets are also used as a delimiter for some command parameters)
`{ ... }` (curly braces)	Zero or more occurrences
`(a | b | c)`	Choose only one of the alternatives

The following table lists the recurring syntax parameters and the values that you can use with the recurring syntax parameters.

Recurring Syntax	Syntax Value
`autoconfigure-vols-attr-` `value-list`	`autoconfigure-vols-attr-value-pair` `{autoconfigure-vols-attr-value-pair}`

Recurring Syntax	Syntax Value

autoconfigure-vols-attr-value-pair	`driveType=`*drive-type* `|`
	`driveMediaType=`*drive-media-type* `|`
	`raidLevel=`*raid-level* `|`
	`volumeGroupWidth=`*integer-literal* `|`
	`volumeGroupCount=`*integer-literal* `|`
	`volumesPerGroupCount=`*integer-literal*[6]`|`
	`hotSpareCount=`*integer-literal* `|`
	`segmentSize=`*segment-size-spec* `|`
	`cacheReadPrefetch=(TRUE | FALSE)`
	`securityType=(none | capable | enabled)`[7]`|`
	`dataAssurance=(none | enabled)`[5]

boolean	`(TRUE | FALSE)`

cache-flush-modifier-setting	`immediate, 0, .25, .5, .75, 1, 1.5, 2,`
	`5, 10, 20, 60, 120, 300, 1200, 3600,`
	`infinite`

capacity-spec	*integer-literal* `[KB	MB	GB	TB	Bytes]`

count-based-repository-spec	`repositoryRAIDLevel =`*repository-raid-level* `repositoryDriveCount=`*integer-literal*
	`[repositoryVolumeGroupUserLabel =`*user-label*`] [driveType=`*drive-type*[4]
	`] [trayLossProtect=(TRUE | FALSE)`[1]`] |`
	`[drawerLossProtect=(TRUE | FALSE)`[2]`] |`
	`[dataAssurance=(none | enabled)`[5]`] |`

create-raid-vol-attr-value-list	*create-raid-volume-attribute-value-pair* {*create-raid-volume-attribute-value-pair*}

create-raid-volume-attribute-value-pair	`capacity=`*capacity-spec* `| owner=(a | b) |`
	`cacheReadPrefetch=(TRUE | FALSE) |`
	`segmentSize=`*integer-literal* `|`
	`usageHint=`*usage-hint-spec*

create-volume-copy-attr-value-list	*create-volume-copy-attr-value-pair* {*create-volume-copy-attr-value-pair*}

create-volume-copy-attr-value-pair	`copyPriority=(highest | high | medium |`
	`low | lowest) |`
	`targetReadOnlyEnabled=(TRUE | FALSE) |`
	`copyType=(offline | online) |`
	`repositoryPercentOfBase=(20 | 40 | 60 |`
	`120 | default) |`
	`repositoryGroupPreference=(sameAsSource`
	`| otherThanSource | default)`

drive-media-type	`(HDD | SSD | unknown | allMedia)` *HDD* means hard disk drive. *SSD* means solid state disk.

Recurring Syntax	Syntax Value
drive-spec	*trayID,slotID* or *trayID,drawerID,slotID* A drive is defined as two or three integer literal values separated by a comma. Low-density trays require two values. High-density trays, those trays that have drawers, require three values.
drive-spec-list	*drive-spec drive-spec*
drive-type	(fibre | SATA | SAS) **Note:** Only SAS drives are supported for firmware versions 7.86 and later.
error-action	(stop | continue)

ethernet-port-options	```enableIPv4=(TRUE | FALSE) |```
	```enableIPv6=(TRUE | FALSE) |```
	```IPv6LocalAddress=ipv6-address |```
	```IPv6RoutableAddress=ipv6-address |```
	```IPv6RouterAddress=ipv6-address |```
	```IPv4Address=ip-address |```
	```IPv4ConfigurationMethod= (static | dhcp)```
	```| IPv4GatewayIP=ip-address |```
	```IPv4SubnetMask=ip-address |```
	```duplexMode=(TRUE | FALSE) |```
	```portSpeed=(autoNegotiate | 10 | 100 |```
	```1000)```

feature-identifier	**Note:** Most features in SANtricity 11.30 are enabled by default. ```driveSecurity```
filename	*string-literal*
gid	*string-literal*
hex-literal	A literal in the range of 0x00 - 0xFF.
host-card-identifier	(1 | 2 | 3 | 4)
host-type	*string-literal* | *integer-literal*

instance-based- repository- spec	```(repositoryRAIDLevel =repository-raid-```
	```level repositoryDrives= (drive-spec-```
	```list)```
	```[repositoryVolumeGroupUserLabel =user-```
	```label] [trayLossProtect=(TRUE | FALSE)[1]])```
	```|```
	```[drawerLossProtect=(TRUE | FALSE)[2]]) |```
	```(repositoryVolumeGroup=user-label```
	```[freeCapacityArea=integer-literal[3]])```
	
	Specify the repositoryRAIDLevel parameter with the repositoryDrives parameter. Do not specify the RAID level or the drives with the volume group. Do not set a value for the trayLossProtect parameter when you specify a volume group.

Recurring Syntax	Syntax Value
`ip-address`	`(0-255).(0-255).(0-255).(0-255)`
`ipv6-address`	`(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF): (0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF)` You must enter all 32 hexadecimal characters.
`iscsi-host-port`	`(1 | 2 | 3 | 4)` The host port number might be 2, 3, or 4 depending on the type of controller you are using.
`iscsi-host-port-options`	`IPv4Address=ip-address |` `IPv6LocalAddress=ipv6-address |` `IPv6RoutableAddress=ipv6-address |` `IPv6RouterAddress=ipv6-address |` `enableIPv4=(TRUE | FALSE) |` `enableIPv6=(TRUE | FALSE) |` `enableIPv4Priority=(TRUE | FALSE) |` `enableIPv6Priority=(TRUE | FALSE) |` `IPv4ConfigurationMethod=(static | dhcp)` `|` `IPv6ConfigurationMethod= (static | auto)` `| IPv4GatewayIP=ip-address |` `IPv6HopLimit=integer |` `IPv6NdDetectDuplicateAddress=integer |` `IPv6NdReachableTime=time-interval |` `IPv6NdRetransmitTime=time-interval |` `IPv6NdTimeOut=time-interval |` `IPv4Priority=integer |` `IPv6Priority=integer	IPv4SubnetMask=ip-` `address |` `IPv4VlanId=integer	IPv6VlanId=integer` `|` `maxFramePayload=integer |` `tcpListeningPort=tcp-port-id |` `portSpeed=(autoNegotiate | 1 | 10)`
`iscsiSession`	`[session-identifier]`
`nvsram-offset`	`hex-literal`

`nvsramBitSetting`	`nvsram-mask, nvsram-value = 0xhexadecimal,`
	`0xhexadecimal|integer-literal`
	The `0xhexadecimal` value is typically a value from 0x00 to 0xFF.

`nvsramByteSetting`	`nvsram-value = 0xhexadecimal|integer-literal` The `0xhexadecimal` value is typically a value from 0x00 to 0xFF.
`portID`	`(0-127)`
`raid-level`	`(0 | 1 | 3 | 5 | 6)`

Recurring Syntax	Syntax Value
`recover-raid-volume-attr-value-list`	`recover-raid-volume-attr-value-pair {recover-raid-volume-attr-value-pair}`
`recover-raid-volume-attr-value-pair`	`owner=(a|b) |cacheReadPrefetch=(TRUE | FALSE)	dataAssurance=(none | enabled)`
`repository-raid-level`	`(1 | 3 | 5 | 6)`
`repository-spec`	`instance-based-repository-spec|count-based-repository-spec`
`segment-size-spec`	`integer-literal` - all capacities are in base-2.
`serial-number`	`string-literal`
`slotID`	`(1-24)`
`test-devices`	`controller=(a|b) esms=(esm-spec-list)drives=(drive-spec-list)`
`test-devices-list`	`test-devices {test-devices}`
`time-zone-spec`	`(GMT+HH:MM | GMT-HH:MM) [dayLightSaving=HH:MM]`
`trayID-list`	`trayID {trayID}`
`usage-hint-spec`	`usageHint=(multiMedia | database | fileSystem)` The usage hint, or expected I/O characteristics, of the volume are used by the controller to indicate an appropriate default volume segment size and dynamic cache read prefetch. For file system and database, a 128 KB segment size is used. For multimedia, a 256 KB segment size is used. All three usage hints enable dynamic cache read prefetch.
`user-label`	`string-literal` Valid characters are alphanumeric, the dash, and the underscore.
`user-label-list`	`user-label {user-label}`
`volumeGroup-number`	`integer-literal`
`wwID`	`string-literal`

[1]For tray loss protection to work, your configuration must adhere to the following guidelines:

Level	Criteria for Tray Loss Protection	Minimum number of trays required
Disk Pool	The disk pool contains no more than two drives in a single tray.	6
RAID 6	The volume group contains no more than two drives in a single tray.	3
RAID 3 or RAID 5	Each drive in the volume group is located in a separate tray.	3
RAID 1	Each drive in a RAID 1 pair must be located in a separate tray.	2
RAID 0	Cannot achieve Tray Loss Protection.	Not applicable

[2]For drawer loss protection to work, your configuration must adhere to the following guidelines:

Level	Criteria for Drawer Loss Protection	Minimum number of drawers required
Disk Pool	The disk pool includes drives from all five drawers and there are an equal number of drives in each drawer. A 60-drive tray can achieve Drawer Loss Protection when the disk pool contains 15, 20, 25, 30, 35, 40, 45, 50, 55, or 60 drives.	5
RAID 6	The volume group contains no more than two drives in a single drawer.	3
RAID 3 or RAID 5	Each drive in the volume group is located in a separate drawer.	3
RAID 1	Each drive in a mirrored pair must be located in a separate drawer.	2
RAID 0	Cannot achieve Drawer Loss Protection.	Not applicable

If you have a storage array configuration in which a volume group spans several trays, you must make sure that the setting for drawer loss protection works with the setting for tray loss protection. You can have drawer loss protection without tray loss protection. You cannot have tray loss protection without drawer loss protection. If the `trayLossProtect` parameter and the `drawerLossProtect` parameter are not set to the same value, the storage array returns an error message and a storage array configuration will not be created.

[3] To determine if a free capacity area exists, run the `show volumeGroup` command.

[4] The default drive (drive type) is `SAS` .

The `driveType` parameter is not required if only one type of drive is in the storage array. If you use the `driveType` parameter, you also must use the `hotSpareCount` parameter and the `volumeGroupWidth` parameter.

[5] The `dataAssurance` parameter relates to the Data Assurance (DA) feature.

The Data Assurance (DA) feature increases data integrity across the entire storage system. DA enables the storage array to check for errors that might occur when data is moved between the hosts and the drives. When this feature is enabled, the storage array appends error-checking codes (also known as cyclic redundancy checks or CRCs) to each block of data in the volume. After a data block is moved, the storage array uses these CRC codes to determine if any errors occurred during transmission. Potentially corrupted data is neither written to disk nor returned to the host.

If you want to use the DA feature, start with a volume group or disk pool that includes only drives that support DA. Then, create DA-capable volumes. Finally, map these DA-capable volumes to the host using an I/O interface that is capable of DA. I/O interfaces that are capable of DA include Fibre Channel, SAS, and iSER InfiniBand (iSCSI Extensions for RDMA/IB). DA is not supported by iSCSI over TCP/IP, or by the SRP InfiniBand.

> **Note:** When all the required hardware and the I/O interface is DA-capable, you can set the `dataAssurance` parameter to `enabled` and then use DA with certain operations. For example, you can create a volume group that includes DA-capable drives, and then create a volume within that volume group that is DA-enabled. Other operations that use a DA-enabled volume have options to support the DA feature.

[6] The `volumesPerGroupCount` parameter is the number of equal-capacity volumes per volume group.

[7] The `securityType` parameter enables you to specify the security setting for a volume group that you are creating. All of the volumes are also set to the security setting that you choose. Available options for setting the security setting include:

- `none` – The volume group is not secure.

- `capable` – The volume group is security capable, but security has not been enabled.

- `enabled` – The volume group is security enabled.

> **Note:** A storage array security key must already be created for the storage array if you want to set `securityType=enabled`. (To create a storage array security key, use the `create storageArray securityKey` command.)

Usage guidelines

This list provides guidelines for writing script commands on the command line:

- You must end all commands with a semicolon (;).

- You can enter more than one command on a line, but you must separate each command with a semicolon (;).

- You must separate each base command and its associated primary parameters and secondary parameters with a space.

- The script engine is not case sensitive. You can enter commands by using uppercase letters, lowercase letters, or mixed-case letters.

- Add comments to your scripts to make it easier for you and future users to understand the purpose of the script commands. (For information about how to add comments, see *Adding comments to a script file* on page 36.)

> **Note:** While the CLI commands and the script commands are not case sensitive, user labels (such as for volumes, hosts, or host ports) are case sensitive. If you try to map to an object that is identified by a user label, you must enter the user label exactly as it is defined, or the CLI commands and the script commands will fail.

Adding comments to a script file

The script engine looks for certain characters or a command to show comments. You can add comments to a script file in three ways:

- Add text after two forward slashes (//) as a comment until an end-of-line character is reached. If the script engine does not find an end-of-line character in the script after processing a comment, an error message appears, and the script operation is terminated. This error usually occurs when a comment is placed at the end of a script and you have forgotten to press the **Enter** key.

```
// Deletes the existing configuration.
set storageArray resetConfiguration=true;
```

- Add text between /* and */ as a comment. If the script engine does not find both a starting comment notation and an ending comment notation, an error message appears, and the script operation is terminated.

```
/* Deletes the existing configuration */
set storageArray resetConfiguration=true;
```

- Use the show statement to embed comments in a script file that you want to appear while the script file is running. Enclose the text that you want to appear by using double quotation marks (" ").

```
show "Deletes the existing configuration";
set storageArray resetConfiguration=true;
```

Storage management

The SANtricity 11.30 release introduces a new kind of controller, the E2800. This section provides information on storage management for three different environments: an environment containing a single E2800, an environment containing multiple E2800s, and an environment containing one or more E2800s, in addition to one or more E2700s, E5600s, and/or EF560s.

Managing an E2800 storage array

What's new in an E2800 storage array

The E2800 is a new entry-level 12 GB SAS system with SANtricity 11.30 software. This system features the following new capabilities:

- Embedded web services

- Embedded SANtricity System Manager, with an easy to use graphical user interface

- The ability to view SSL information, get a Certificate Signing Request (CSR), and import a new certificate

- The ability to store and present up to 30 days of performance data, including I/O latency, IOPS, and throughput

- The ability to do application/workload tagging

- Easier alert management, including an embedded SNMP agent and MIB

- The ability to view, configure and test syslog receivers

- Embedded AutoSupport functionality

The new E2800 storage array can be managed using SANtricity 11.30, embedded directly on the controller, along with embedded web services. If you have a single E2800 and have no need to do mirroring or use a command line interface, a web browser is all you need to manage your system.

E2800 with embedded software, web services,
Event Monitor, and AutoSupport

Network

SANtricity System Manager in a
web browser

Managing multiple E2800 storage arrays

Install SANtricity Storage Manager Enterprise Management Window for domain management, for mirroring operations between two storage arrays, and for using the SMcli or the Script Editor.

You also can use the RESTful API (Java or Python client libraries in the SDK or curl commands) to perform management for the E2800 storage array.

The following graphic shows the management components you can use to manage one or more E2800 storage arrays:

Script
Editor

Enterprise Management Window

SANtricity System
Manager in a web browser

C:\

SMcli

Network

{REST}

Web services proxy

Management station
with SANtricity 11.30

E2800 arrays with embedded
software, web services, Event
Monitor, and AutoSupport
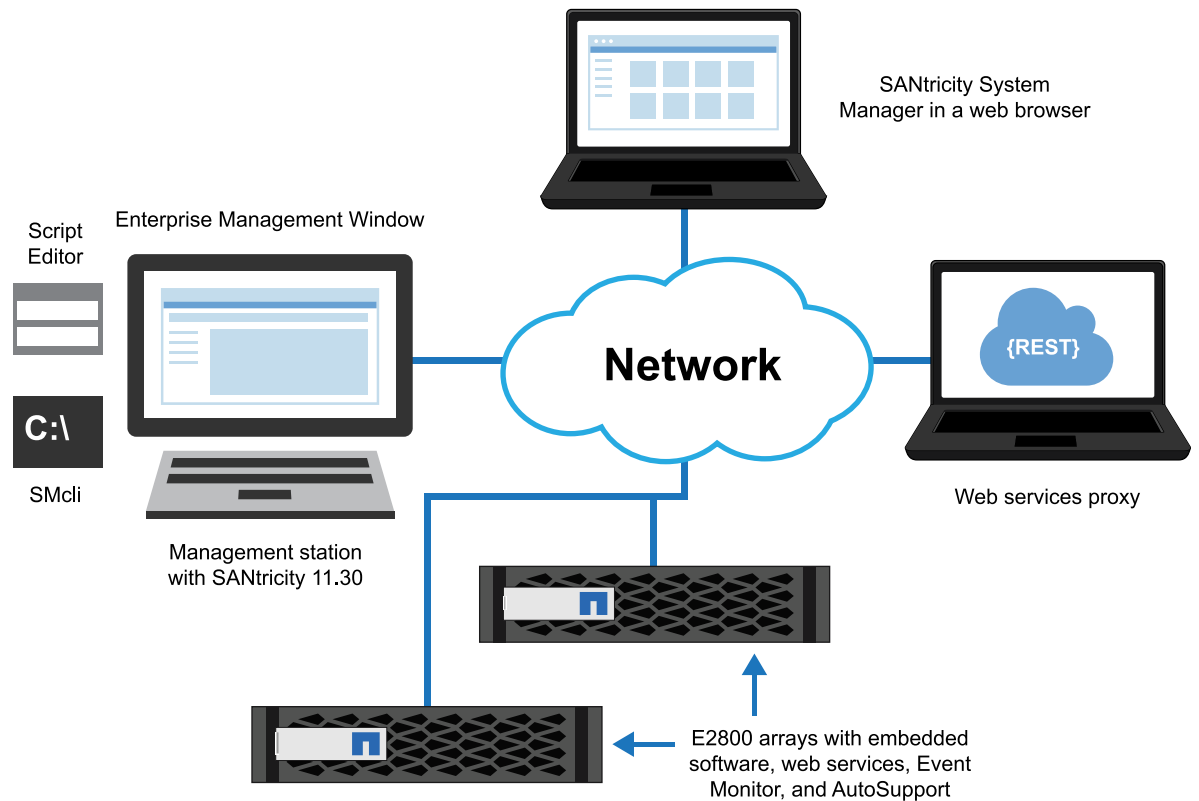
Management of mixed storage arrays

The Enterprise Management Window (EMW) on the management station discovers E2800, E2700,
E5600, and EF560 storage arrays in your domain. A mixed environment is easy to manage, because
the EMW launches the appropriate software for managing each individual array. For the E2800
storage array, the EMW launches SANtricity System Manager in a web browser. For the E2700,
E5600, or EF560 storage arrays, the EMW launches an Array Management Window.

E2800

SANtricity System Manager

E2700/E5600/
EF560

Enterprise Management Window

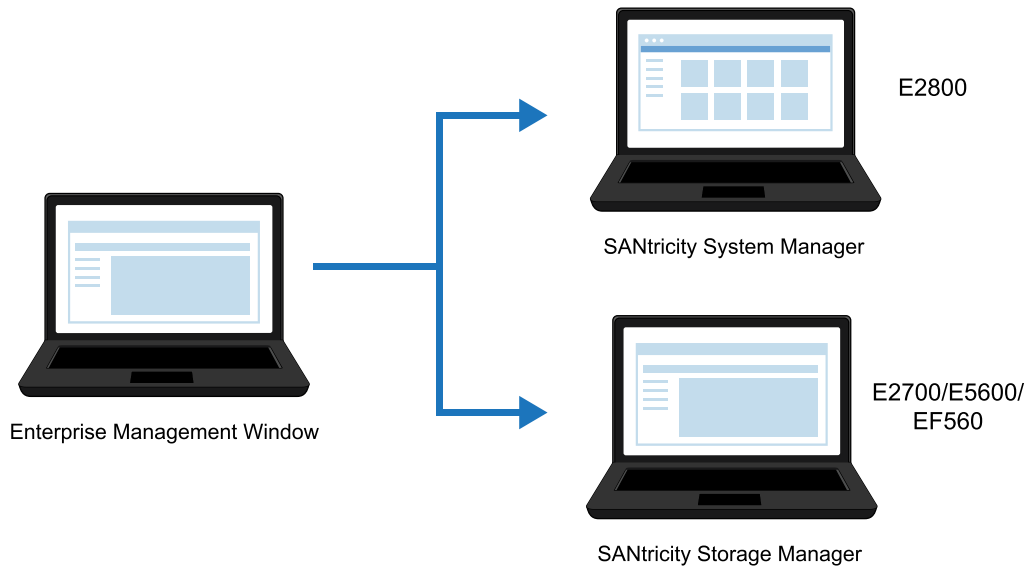SANtricity Storage Manager

In addition to using the Enterprise Management Window for domain management, you also use it for mirroring operations between two storage arrays, and for using the SMcli or the Script Editor.

For managing individual arrays, you can use the graphical user interfaces (GUID) described earlier, the SMcli or Script Editor (with some exceptions noted below), or the RESTful API.

The following graphic shows the management components in a mixed environment:

Array Management
Window - only for E2700,
E5600, and EF560

SANtricity System
Manager in a web browser

Only for E2800s

Enterprise Management Window

Script
Editor

C:

SMcli

Event
Monitor

*Only for E2700,
E5600, and EF560*

Management station
with SANtricity 11.30

Network

{REST}

Web services proxy

E2800 arrays with embedded
software, web services, Event
Monitor, and AutoSupport

E2700, E5600,
and/or EF560

Event Management

Critical events indicate a problem with the storage array. If you resolve the critical event immediately, you might prevent loss of data access.

When a critical event occurs, it is logged in the Event Log. All critical events are sent to the SNMP management console or to the email recipient that you have configured to receive alerts.

When you receive a critical event, refer to the Recovery Guru procedure for a detailed description of the critical event. Complete the Recovery Guru procedure to correct the critical event. To correct certain critical events, you might need to contact technical support.

There are differences in the way events and SNMP trap messages are managed between the new E2800 and the E2700, E5600 and EF560 storage arrays. These differences are described below. Because of the differences, there are different management methods available.

- For the E2800 storage array, you can configure event and SNMP capabilities and settings with the SANtricity System Manager, or you can use the RESTful API. You also can write scripts directly accessing the embedded web services on the controller, using the RESTful API.

- For the E2700, E5600 and EF560 storage arrays, you can configure event and SNMP capabilities via the SANtricity Storage Manager, or you can use the SMcli or the script editor in the Enterprise Management Window. You must run the Event Monitor on at least one management station on the network.

About SNMP monitoring of the storage array

The following terminology definitions are useful in understanding the SNMP capabilities of the E-Series storage arrays:

Term	Description
Storage array	This is a managed device that can be managed and/or monitored through SNMP. • With the E2700, E5600, and EF560 storage arrays, the Event Monitor enables an SNMP management application to provide monitoring capabilities for the array. • With the E2800 storage array, an SNMP agent is embedded on the controller, allowing an SNMP management application to provide simple monitoring capabilities for the array. The embedded SNMP agent is compliant with the SNMP V2C standard and RFC 1213 (MIB-II).

Term	Description
SNMP agent	Software that generates outgoing trap messages whenever critical MEL events are logged on the array. • With the E2700, E5600, and EF560 storage arrays, the Event Monitor contains the SNMP agent, and only generates outgoing trap messages. • With the E2800 storage array, an SNMP agent is embedded on the controller, allowing the array to provide basic query support and also handle GET requests from an SNMP management application.
Management Information Base (MIB)	A Management Information Base describes the structure of the data for a specific managed device. • With the E2700, E5600, and EF560 storage array, the Event Monitor on the storage management station contains the MIB file. • With the E2800 storage array, the MIB file is embedded on the controller.
Trap message	This is an asynchronous notification from the SNMP agent to the SNMP manager that a specific event has occurred on the managed device.
Network Management System	This is an SNMP management application acting as an administration and monitoring console for SNMP managed devices, including E-Series storage arrays.
SNMP community	An SNMP community comprises both managed devices and SNMP management applications. The SNMP community string is included in the data packets that are passed over the network. • The community string is included in all trap notifications that are passed from an SNMP agent to an SNMP management application. • The SNMP agent on the E2800 storage array rejects queries from an SNMP management system that do not contain the appropriate community string.

Term	Description
SNMP trap destination	SNMP TRAP messages will only be sent to destinations you specify. Each trap destination must have a unique IP address. The embedded SNMP Agent supports configuration of both IPV4 and IPV6 destination addresses. A previously registered community must also be specified by the user when configuring a trap destination so that the embedded SNMP agent can place the correct community name in each outgoing trap message.

Event management for E2800 storage arrays
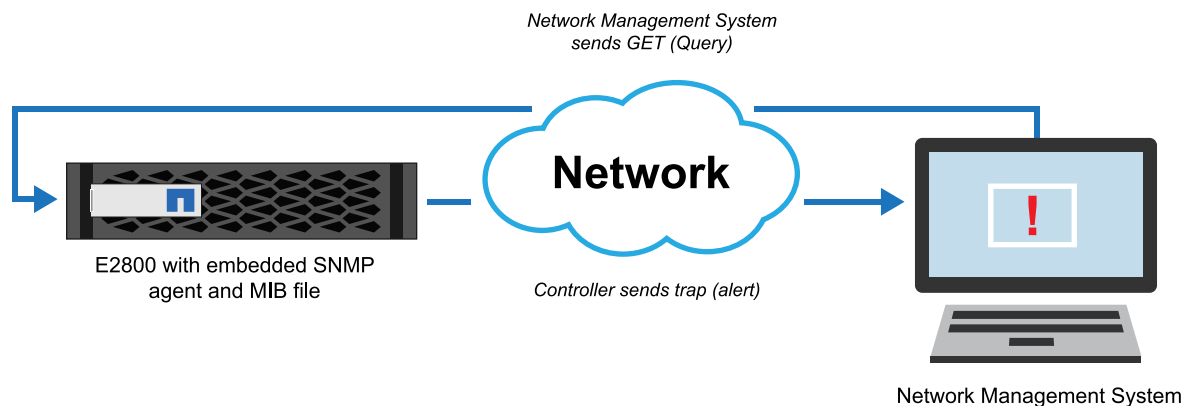
RESTful API sample code in Python and Java client libraries for new E2800 capabilities

Sample programs that perform the following tasks are available in the Python and Java client libraries, part of the REST SDK:
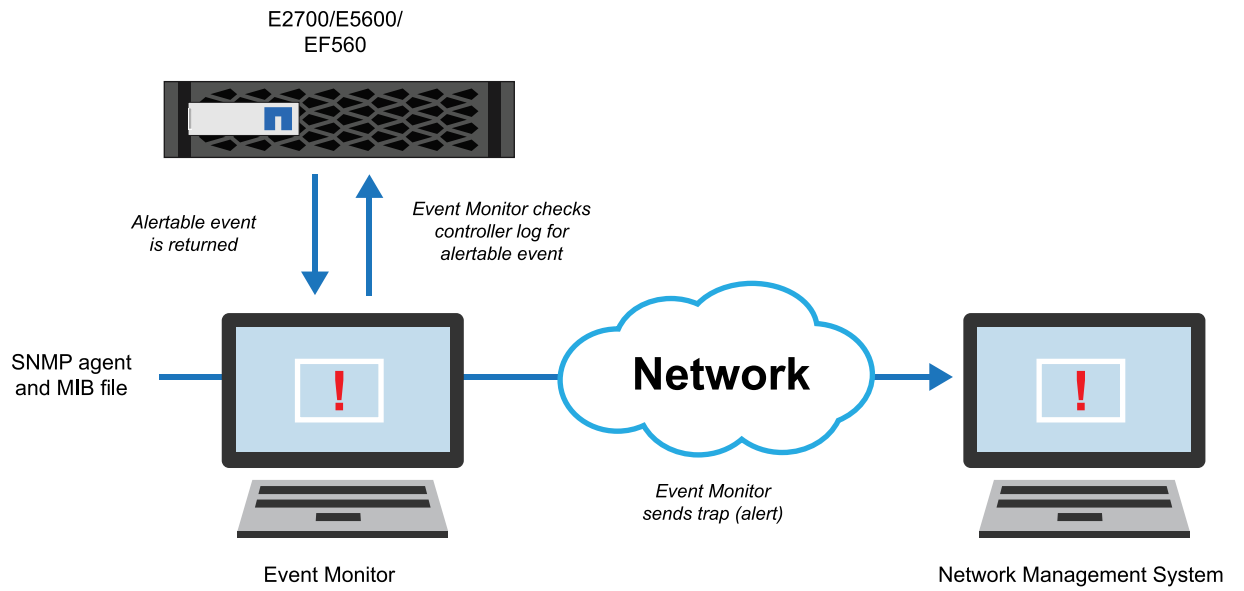
- View SSL configuration, get a Certificate Signing Request (CSR), and import a new certificate

- View, configure and test syslog receivers

- View historical data and rolled up statistics for I/O latency, IOPS, and throughput.

- Create volume sets based on application types (application tagging)

- Collect and view support bundle artifacts specific to the E2800 storage array

Event management for the E2700, E5600, and EF560

Event management for the E2700, E5600, and EF560 storage arrays

With the E2700, E5600, and EF560 storage arrays, the Event Monitor on the storage management station periodically queries the storage array for alertable events. The SNMP agent running in the

Event Monitor then generates a trap message for each event and forwards to the SNMP management application.

E2700/E5600/
EF560

*Event Monitor checks
controller log for
alertable event*

*Alertable event
is returned*

SNMP agent
and MIB file

!

Network

!

*Event Monitor
sends trap (alert)*

Event Monitor

Network Management System

AutoSupport Overview

AutoSupport (ASUP) is a feature that enables storage arrays to automatically collect support data into a customer support bundle and send the data to NetApp technical support. Technical support can then perform remote troubleshooting and problem analysis with the storage management software.

About AutoSupport and AutoSupport OnDemand

ASUP collects support data to report configuration, subsystem status, and exceptions in near-real time. ASUP OnDemand (AOD) is an optional extension to ASUP that introduces two-way communication between the E-Series array and the ASUP backend. ASUP RemoteDiagnostics provides the ability for NetApp technical support to remotely ask the given E-Series array to create a new ASUP message, or request retransmission of a particular ASUP message.

Important: Under no circumstances is business data from the storage array ever sent to NetApp support.

ASUP messages typically include a collection of system log files, configuration data (formatted XML and unstructured command output), state data (subsystem up/down, capacity used), performance metrics, and system inventory data. All of the data gathered is collected into a single compressed archive file format (7z).

The storage array uses the Internet to send ASUP messages to the ASUP backend. The ASUP backend provides near-real time access to the messages by technical support. ASUP requires compliance to the following transport protocol-specific requirements:

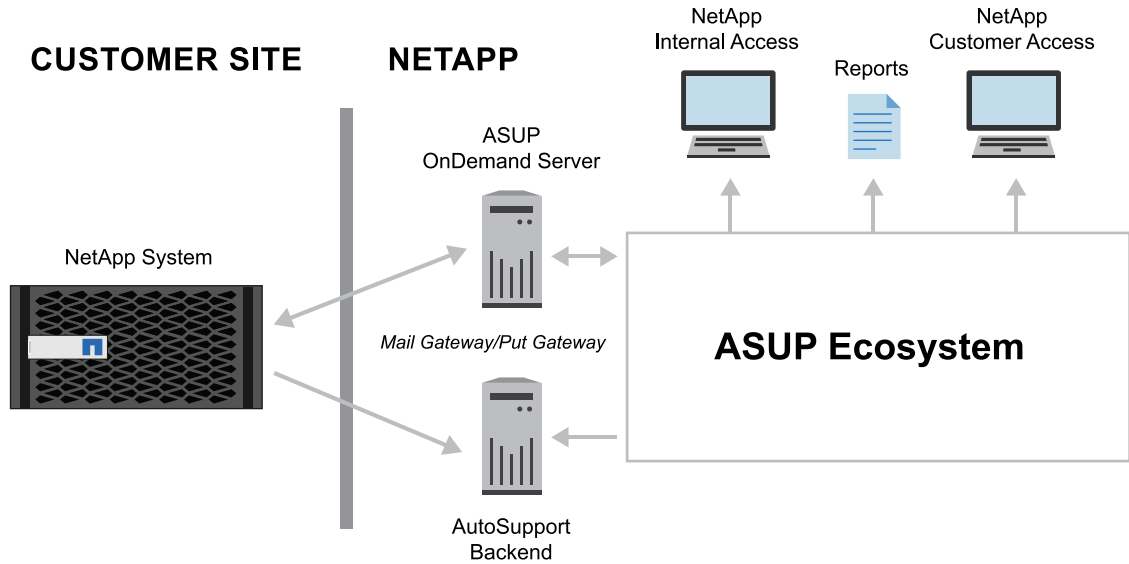• HTTP or HTTPS upload

• SMTP notifications

Methods for collecting support data

With the implementation of ASUP and AOD, you have three possible methods for collecting support data in a storage array:

• Standard ASUP collection - Data is automatically collected and sent to technical support, triggered by a daily or weekly schedule, or by an event occurrence that is specified to automatically send a support bundle.

• ASUP OnDemand - With AOD enabled, before an E-Series system can send any standard ASUP message to the ASUP backend, it must first obtain permission from the ASUP OnDemand Server (AODS). ASUP OnDemand also introduces the ability for a given E-Series array to check in periodically with the AODS to see if there are any pending actions for the particular E-Series array.

• Legacy support bundle collection (E2700, E5600, or EF560 storage arrays only) - Collection of legacy support bundle data is user-configurable at intervals you schedule. You can then manually send the support bundles to technical support.

For the E2700, E5600, or EF560 storage arrays only, ASUP operations and legacy support bundle operations are mutually exclusive on a given storage array. When you turn on ASUP, with or without AOD, you automatically disable legacy support bundle collection. If you want to run legacy support bundle collection, you must turn off ASUP. Because ASUP speeds up troubleshooting and problem analysis, ASUP is the preferred data collection method to use if available on the storage array.

AutoSupport components

AutoSupport component	Description
NetApp System	The NetApp System is an array on the customer site. The system is managed by the SANtricity software on a management station or host, and the Persistent Monitor runs on the management station or host. **Note:** The NetApp System stores an AutoSupport log, located in the ASUPLog directory on the array. The log provides information about status, history of transmission, and any errors encountered during the collection and delivery of the AutoSupport messages.
ASUP OnDemand Server (AODS)	With AOD enabled, before an E-Series system can send any standard ASUP message to the ASUP backend, it must first obtain permission from the AODS. **Important:** Under no circumstances does the AODS initiate communication with an E-Series array. The flow of communication always begins with the E-Series array, and a reply is sent back from the AODS. The reply might contain actions requested by technical support, such as to retransmit a missing ASUP message.
Mail Gateway/Put Gateway	The HTTP(s) Put/Post Gateway moves the messages into the ASUP Ecosystem.
ASUP Ecosystem	The ASUP Ecosystem provides near real-time access to messages by NetApp technical support. Additionally, there is a portal within the ecosystem that NetApp customers can use to access their ASUP data. Finally, the ASUP Ecosystem provides the ability to generate reports used by various groups within NetApp.

AutoSupport data processing

The backend processing of AutoSupport data is used in several ways:

- Routing messages based on message content
- Support case generation

- Proactive health-check and system level risk detection

- Service entitlement

- Subscription-based service fulfillment

- Persisting the support information associated with the ASUP message

- Data analysis to improve the E-Series products

### AutoSupport configuration in E-Series storage arrays

The E2800 storage array includes embedded ASUP functionality, rather than relying on an externally installed Event Monitor to configure and monitor AutoSupport (as does the E2700, E5600, and EF560 storage arrays). Because of the differences in the E2800 storage array, there are different management methods for configuring AutoSupport across the two kinds of arrays:

- For the E2800 storage array, you can use SANtricity System Manager or the RESTful API to configure AutoSupport. There are sample programs in the Java and Python client libraries which are part of the REST API SDK.

- For the E2700, E5600, and EF560 storage arrays, you can use the SANtricity Storage Manager application or SMcli.

AutoSupport for E2800 storage arrays

With the E2800 storage array, the controller contains embedded AutoSupport functionality.
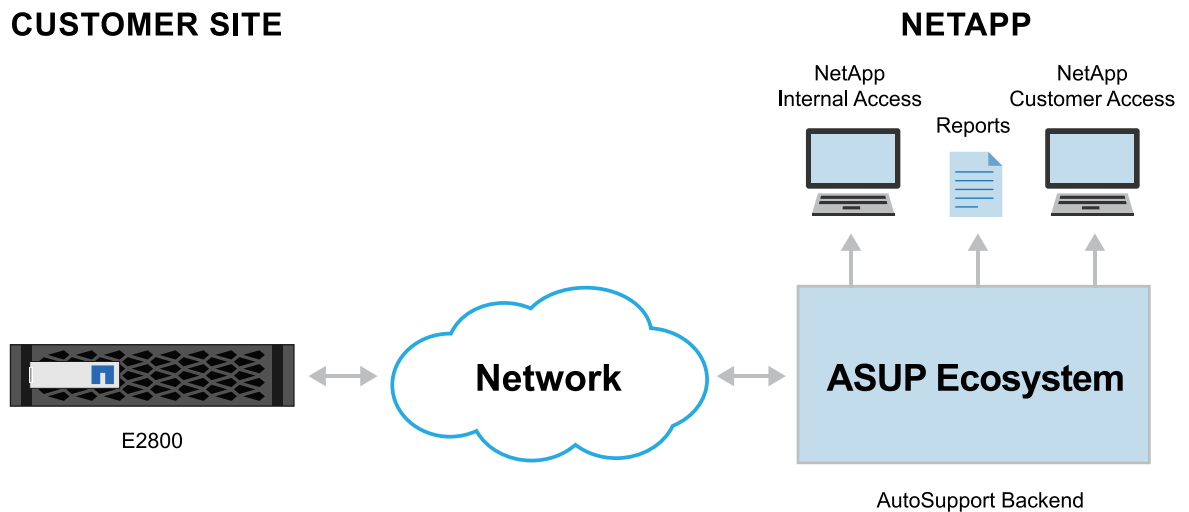

cURL statements for AutoSupport configuration

This document provides cURL statements as an alternative to SMcli for the E2800. These statements connect and communicate with the embedded web services on the E2800 to accomplish these tasks, using the RESTful API.

- View and update AutoSupport configuration

- View and update collection schedules

- View AutoSupport log files

- Validate AutoSupport configuration

AutoSupport for E2700, E5600, and EF560 storage arrays

With the E2700, E5600, or EF560 storage arrays, the Event Monitor on a storage management station on your network must be running to use AutoSupport.
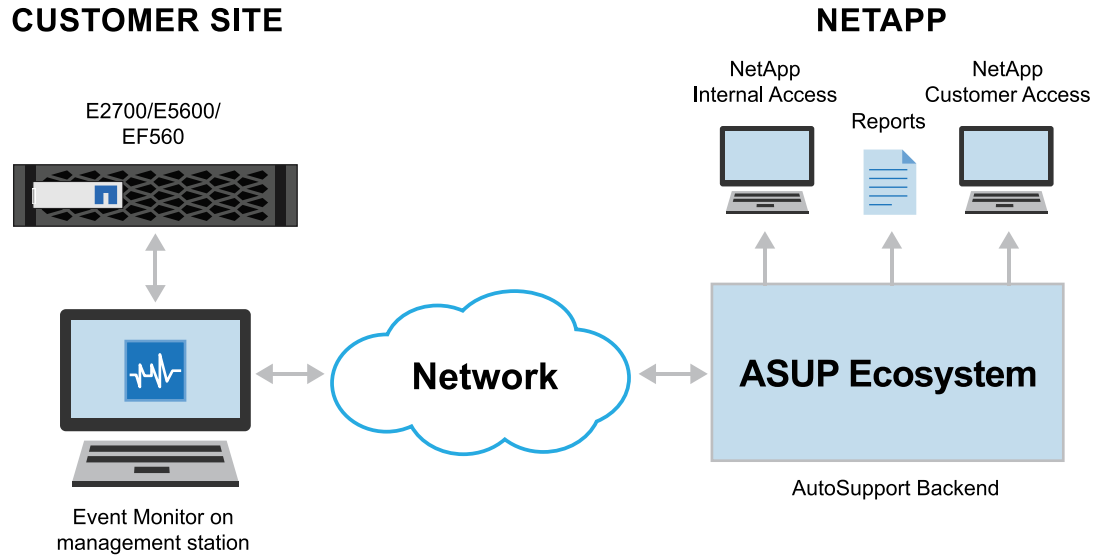
Management Methods in SANtricity 11.30

SANtricity 11.30 introduces a new controller with embedded management software, web services, and Event Monitoring and AutoSupport. Previous controllers such as the E2700, E5600, and EF560 do not have this embedded functionality. Because you may have a mixed environment, with both the new E2800 storage array and older storage arrays, there are a variety of options to manage this mixed environment.

Use the table below to determine, by task, which management component provides the functionality you need.

Management Methods

The following high level summary describes management methods for different functionality:

Task	E2700/E5600/EF560, all other E-Series arrays except the E2800	E2800
Array Management GUI	SANtricity AMW - install on management station	SANtricity System Manager - On-Box
Multi-system management	SANtricity EMW (part of SANtricity Storage Manager) or Web Service proxy	SANtricity EMW (part of SANtricity Storage Manager) or Web Service proxy
Email Alerts, ASUP	Configure and dispatch from Event Monitor (part of SANtricity Storage Manager) on management station	Configure and dispatch on-box
SNMP	On management station, traps only	Embedded, adds GETs
REST APIs	Web Services proxy only	Embedded or Web Services proxy
CLI	On management station, all GUI operations +	On management station, no monitor configuration
In-band management	Full GUI and CLI, no REST	CLI and most EMW functions
Remote login	Telnet	SSH

Management Use Cases

The following more detailed table is included for reference. The following list includes the acronyms used in the table:

- System Manager - SANtricity System Manager (embedded management software for an E2800 storage array)

- EMW - Enterprise Management Window (used for domain management)

- AMW - Array Management Window (storage management software for an E2700, E5600, or EF560)

- SMcli - SANtricity command line interface

- EMW script editor CLI - script editor in Enterprise Management Window

- REST - REST API endpoints, available in web proxy application for all arrays and natively for the E2800.

Task	E2700/E5600/EF560, all other E-Series arrays except the E2800	E2800
Manage and discover		
Discover an array in your management domain	• EMW	• EMW
Add or remove an array from your management domain	• EMW • SMcli*	• EMW • SMcli*
Launch SANtricity System Manager	• N/A	• EMW
Launch Array Management Window (AMW)	• EMW	• N/A
AutoSupport and legacy support bundle collection		
Enable/disable AutoSupport, AutoSupport OnDemand, and AutoSupport Remote Diagnostics features	• EMW • SMcli*	• System Manager • REST
Show AutoSupport logs for all or a select storage array	• EMW • SMcli*	• System Manager • REST
Enable or disable legacy support bundle collection for a select storage array	• EMW • SMcli*	• N/A
Specify legacy support bundle collection schedule	• EMW • SMcli*	• N/A
Configuration and status		
Display information (other than alert settings) about configured arrays	• AMW • SMcli* • EMW script editor CLI	• System Manager • SMcli* • EMW script editor CLI • REST
Show IP address of each array	• AMW • SMcli* • EMW script editor CLI	• System Manager • SMcli* • EMW script editor CLI • REST

Task	E2700/E5600/EF560, all other E-Series arrays except the E2800	E2800
Show WWN of each array	• AMW • SMcli* • EMW script editor CLI	• System Manager • SMcli* • EMW script editor CLI • REST
Show status of each array	• AMW • SMcli • EMW script editor CLI	• System Manager • EMW (domain level status) • SMcli* • EMW script editor CLI • REST
Set up remote volume mirroring groups and pairs	• AMW • SMcli* • EMW script editor CLI	• System Manager (requires launch from EMW to correctly set up mirroring relationships) • SMcli* • EMW script editor CLI • REST
Array-level configuration, provisioning, and tuning	• AMW • SMcli* • EMW script editor CLI	• System Manager • REST
Alert and SNMP Configuration		
Show global alert settings	• EMW • SMcli* • EMW script editor CLI	• System Manager • REST
Specify email server and other configuration for global alert settings	• EMW • SMcli* • EMW script editor CLI	• System Manager • REST
Remove an email from configuration for a specific array	• EMW • SMcli • EMW script editor CLI	• System Manager • REST

Task	E2700/E5600/EF560, all other E-Series arrays except the E2800	E2800
Add or remove SNMP trap information for a specific array	• EMW • SMcli*	• System Manager • REST
Send a test email based on global alert settings	• EMW • SMcli*	• System Manager • REST
New features for E2800 only - see SANtricity System Manager online help for descriptions		
Certificate handling - view SSL information, get a Certificate Signing Request (CSR), import a new certificate	• N/A	• EMW • System Manager • REST
More convenient syslog configuration for E2800	• N/A	• System Manager • REST
Save up to 30 days of historical statistical I/O data for E2800	• N/A	• System Manager • REST
Perform application tagging of volumes, snapshots for E2800	• N/A	• System Manager • REST

* Requires installation of SANtricity Storage Manager

Terminology differences in SANtricity 11.30

SANtricity 11.30 introduces, with the E2800 storage array, an embedded storage management software with new terminology. These terminology differences are noted in the following table.

Software terms

The following table defines some of the most commonly used software terms. Note that different terms might apply to different hardware models.

Note that while new terminology is visible in SANtricity System Manager, some documents might still use the old terminology.

Term used for E2800 storage arrays / SANtricity System Manager	Term used for E2700 and E5600 storage arrays, and EF560 flash arrays / SANtricity Storage Manager	Definition
Allocated capacity	Configured, Used, or Physical capacity	The actual physical capacity allocated from the drives in a pool or volume group that is used for volumes and copy services operations.
Allocated capacity limit	Maximum capacity expansion	The quota or cap on how large the allocated physical capacity (repository) for a thin volume can grow.
Assign, unassign	Map, unmap	Associating a Logical Unit Number (LUN) with a host.
Default cluster	Default group	A system-defined entity that is only used if you want all the hosts connected to your storage array to have shared access to all of the volumes. Normally, you will define individual hosts and then assign specific volumes to each of those hosts for restricted access. Refer to the SANtricity System Manager online help for more details.
Event log	Major event log (MEL)	The Event log is a detailed record of events that occur in the storage array. You can use the Event log as a supplementary diagnostic tool to the Recovery Guru for tracing storage array events.

Term used for E2800 storage arrays / SANtricity System Manager	Term used for E2700 and E5600 storage arrays, and EF560 flash arrays / SANtricity Storage Manager	Definition
Host cluster	Host group	A host group or host cluster is a collection of hosts that share access to storage array volumes. Creating host groups lets administrators use the combined processing power of the host group computers to access the volumes. Because host groups also permit multiple hosts to access a single volume, the hosts manage this shared access.
Management client	Storage management station	For the E2800, any system that has a supported browser that can point to the storage array URL and issue management commands. If the system is performing I/O, it is a host system. For the E2700 and E5600 storage arrays and the EF560 flash array, this is a system running the SANtricity Storage Manager application. This could also be a host system if it is performing I/O.
Objects	Elements or Components	Any managed item upon which a performance metric is being collected.
Pool	Disk Pool	Dynamic Disk Pool
Reported capacity	Volume capacity, Virtual capacity	The capacity that is reported up to the host and can accessed by the host.
Reserved capacity	Repository capacity	The physically allocated capacity that is used for any copy service operations or objects.
Reserved capacity threshold	Repository warning threshold	A generic term, used in context within the OLH.
SANtricity OS $n.n0$	Controller firmware $n.n0$	The version of bundled software code that runs on the controller.
Snapshot consistency group	Consistency group	A set of volumes that will have a snapshot image created at the same time for purposes of application consistency.
Thick volume	Standard volume	Describes any non-thin volume.
Unassigned capacity	Unconfigured capacity	Capacity that has not been configured yet into a pool or volume group.

Term used for E2800 storage arrays / SANtricity System Manager	Term used for E2700 and E5600 storage arrays, and EF560 flash arrays / SANtricity Storage Manager	Definition
Written capacity	Used capacity, Consumed capacity	The amount of capacity that has actually be written of the reserved capacity allocated for thin volumes and copy service objects or operations.

Maintaining a storage array

About this task

Maintenance covers a broad spectrum of activity with the goal of keeping a storage array operational and available to all hosts. This chapter provides descriptions of commands you can use to perform storage array maintenance. The commands are organized into four sections:

- Routine maintenance

- Performance tuning

- Troubleshooting and diagnostics

- Recovery operations

The organization is not a rigid approach, and you can use the commands as appropriate for your storage array. The commands listed in this chapter do not cover the entire array of commands you can use for maintenance. Diagnostic or maintenance capabilities are provided using a variety of commands, particularly those starting with `set`.

Routine maintenance

Routine maintenance involves those tasks that you perform periodically in conjunction with E-Series AutoSupport (ASUP) to help make sure that the storage array is running at peak performance or for early detection of conditions that might cause future problems.

Running a media scan

Media scan provides a way of detecting and repairing drive media errors before they are found during a normal read from or write to the drives. Any media scan errors that are detected are reported to the Event Log. The Event Log provides an early indication of an impending drive failure and reduces the possibility of encountering a media error during host operations. A media scan is performed as a background operation and scans all data and redundancy information in defined user volumes.

> **Note:** If media scan is disabled at the storage array level, this operation has no effect.

A media scan runs on each of the volumes in the storage array that have these conditions:

- Optimal status

- No modification operations in progress

- Media scan enabled

Errors that are detected during a scan of a user volume are reported to the Major Event Log (MEL).

> **Note:** Media scan is enabled for all user volumes by default. A media scan is unnecessary on an EF-Series storage array or on any storage array using solid state drives (SSD). Disable media scan by setting the `mediaScanEnabled` parameter of the `set volume` command to `FALSE`.

> **Attention:** If the scan determines that parity is in error, and automatic parity repair is enabled, attempts to repair the parity to a consistent state could make a data inconsistency permanent in parity. Automatic parity repair is disabled by default for all E-Series products.

If the controller firmware supports repairs, the errors are managed as follows:

- **Redundancy mismatches** – Redundancy errors are also known as data/parity mismatch errors. Redundancy errors are found when both the data on a stripe and the redundancy block can be

read, but the redundancy information calculated from the data (the parity) does not match the redundancy information on the media. The redundancy process first retries individual read operations to every drive in the volume to determine if the read response data for the drives has changed. After all of the drives have been read again, the redundancy information is checked for consistency. If the redundancy check is consistent, an informational MEL event is logged. If the check finds that inconsistencies still exist, a critical MEL event is logged. The redundancy inconsistency is then addressed based upon the NVSRAM setting: either the parity repair capability is disabled and no further action is taken, or the data is presumed to be correct and the redundancy information is recalculated and written to the appropriate sectors on the drives.

- **Recovered read error** – The drive could not read the requested data on its first attempt. When a recovered read error occurs, the data is written, with verification, to the same sectors on the drive. If the write-verify operation to the same sectors fails, the failing sectors are reassigned and the data is written, with verification, to replacement sectors on the drive. If the write-verify operation to the replacement sectors fails, the drive is failed. The error is reported to the Event Log.

- **Unrecovered media error** – The drive could not read the requested data on its first try or on any subsequent retries. The result of this action is that for volumes with redundancy protection, the data is reconstructed, rewritten to the drive, and verified, and the error is reported to the Event Log. For volumes without redundancy protection, the error is not corrected, but it is reported to the Event Log.

- **Unfixable error** – The data could not be read, and parity information or redundancy information could not be used to regenerate it. For example, redundancy information cannot be used to reconstruct data on a degraded volume. The result of this action is that the error is reported to the Event Log.

Two commands define media scan properties:

- `set volume`

- `set storageArray`

The `set volume` command enables a media scan for a volume. The command has this form:

```
set (allVolumes | volume [volumeName] |
volumes [volumeName1 ... volumeNameN] |
volume <"wwID">)
mediaScanEnabled=(TRUE | FALSE)
```

When using this command, the volume names must be enclosed in square brackets. If a volume name also has special characters or numbers, you must enclose the volume name in double quotation marks. When using the WWID, enclose the WWID in double quotation marks (" ") inside angle brackets (< >). Do not include colons in the WWID.

The `set storageArray` command defines how frequently a media scan is run on a storage array. The command has this form:

```
set storageArray mediaScanRate=(disabled | 1-30)
```

The `mediaScanRate` values define the number of days over which the media scan runs. Valid values are `disabled` , which turns off the media scan; or `1` day to `30` days, where `1` day is the fastest scan rate, and `30` days is the slowest. A value other than what is shown will not allow the media scan to function. Be aware that a syntax error is not returned if an invalid value is entered, but the command fails.

Running a redundancy check

During a redundancy check, all of the data blocks in a volume are scanned, and depending on the RAID level, deteriorated data is corrected. Redundancy checks are performed when media scans are run if redundancy checking is enabled for a volume.

Note: If media scan is disabled at the storage array level, this operation has no effect.

Note: Running a redundancy check while you are doing a media scan is highly recommended. If you do a media scan without a redundancy check, you incur the following risks:

- For volumes without protection information, any corrupted data will be returned with host read requests.

- Data and redundancy mismatches will never be reported or repaired.

- Redundancy blocks are never read except during drive reconstruction, so an unreadably redundancy block will be undetected until it is needed to reconstruct missing data, and the result may be data loss.

For a description about how to set up and run media scans, see *Running a media scan* on page 56.

Correction is performed as follows:

- For RAID level 3, RAID level 5, or RAID level 6 volumes, redundancy is checked and repaired.

- For RAID level 1 volumes, the data is compared between the mirrored drives and data inconsistencies are repaired.

- RAID level 0 volumes have no redundancy.

Before you can run a redundancy check, you must enable redundancy checking by using the `set volume` command. The command has this form:

```
set (allVolumes | volume [volumeName] |
volumes [volumeName1 ... volumeNameN] |
volume <"wwID">)
redundancyCheckEnabled=(TRUE | FALSE)
```

When using this command, the volume names must be enclosed in square brackets. If a volume name also has special characters or numbers, you must enclose the volume name in double quotation marks. When using the WWID, enclose the WWID in double quotation marks (" ") inside angle brackets (< >). Do not include colons in the WWID.

Resetting a controller

Important: If you are not using a multi-path driver to manage controller access to volumes, resetting the controller means the controller is no longer available for I/O operations until the reset is complete. If a host is using volumes that are owned by the controller being reset, the I/O that is directed to the controller is rejected. Before resetting the controller, either make sure that the volumes that are owned by the controller are not in use, or make sure that a multi-path driver is installed on all of the hosts that are using these volumes.

Resetting a controller is the same as rebooting the controller processors. To reset a controller, use this command:

```
reset controller [(a | b)]
```

Enabling a controller data transfer

At times, a controller might become quiescent while running diagnostics. If this condition occurs, the controller might become unresponsive. To revive a controller that has become quiescent while running diagnostics, use this command:

```
enable controller [(a | b)] dataTransfer
```

Removing persistent reservations

Persistent reservations preserve volume registrations, and they prevent hosts, other than the host holding a reservation for the volume, from accessing the volume.

You must remove persistent reservations before you make these changes to your configuration:

* Change or delete LUN mappings on a volume holding a reservation.

* Delete volume groups or volumes that have any reservations.

To determine which volumes have reservations, use this command:

```
show (allVolumes | volume [volumeName] |
volumes [volumeName1 ... volumeNameN]) reservations
```

To clear persistent volume reservations, use this command:

```
clear (allVolumes | volume [volumeName] |
volumes [volumeName1 ... volumeNameN]) reservations
```

Synchronizing the controller clocks

To synchronize the clocks on both controllers in a storage array with the clock of the management station or host running the instance of SANtricity Storage Manager, use this command:

```
set storageArray time
```

Locating drives

At times, you might need to locate a specific drive. In very large storage array configurations, this task can sometimes be awkward. If you need to locate a specific drive, you can do so by turning on the indicator light on the front of the drive. To locate a drive, use this command:

```
start drive [trayID,drawerID,slotID] locate
```

Note: The drawerID parameter is only required for high capacity drive trays.

To turn off the indicator light after locating the drive, use this command:

```
stop drive locate
```

Note: Since the example did not specify a drive ID, all indicators for all drives with the indicator on are turned off. Use a tray ID, drawer ID if needed, and slot ID to turn off the indicator for a particular drive.

The drive parameter supports both high-capacity drive trays and low-capacity drive trays. A high-capacity drive tray has drawers that hold the drives. The drawers slide out of the drive tray to provide access to the drives. A low-capacity drive tray does not have drawers. For a high-capacity drive tray,

you must specify the identifier (ID) of the drive tray, the ID of the drawer, and the ID of the slot in which a drive resides. For a low-capacity drive tray, you need only specify the ID of the drive tray and the ID of the slot in which a drive resides. For a low-capacity drive tray, an alternative method for identifying a location for a drive is to specify the ID of the drive tray, set the ID of the drawer to `0` , and specify the ID of the slot in which a drive resides.

Performance tuning

Over time, as a storage array exchanges data between the hosts and the drives, its performance can degrade. You can monitor the performance of a storage array and make adjustments to the operational settings on the storage array to help improve performance.

Monitoring the performance

You can monitor the performance of a storage array by using the `save storageArray performanceStats` command. This command saves performance information to a file that you can review to help determine how well the storage array is running. The following table lists the performance information that is saved to the file.

Type of Information	Description
Devices	These devices are included in the file: • **Controllers** – The controller in slot A or slot B and a list of the volumes that are owned by the controller • **Volumes** – A list of the volume names • **Storage array totals** – A list of the totals for both controllers in an active/active controller pair, regardless if one, both, or neither are selected for monitoring
Total I/Os	The number of total I/Os performed since the storage array was started
Read Percentage	The percentage of total I/Os that are read operations (calculate the write percentage by subtracting the read percentage from 100 percent)
Cache Hit Percentage	The percentage of reads that are fulfilled by data from the cache rather than requiring an actual read from a drive
Current KB per second	The current transfer rate in kilobytes per second (current means that the number of kilobytes per second since the last time that the polling interval elapsed, causing an update to occur)
Maximum KB per second	The highest data transfer value that is achieved in the current kilobyte-per-second statistic block
Current I/O per second (IOPS)	The current number of I/Os per second (current means the number of I/Os per second since the last time that the polling interval elapsed, causing an update to occur)
Maximum I/O per second	The highest number of I/Os achieved in the current I/O-per-second statistic block

The command takes this form:

```
save storageArray performanceStats file="filename"
```

In this command, `filename` is the name of the file in which you want to save the performance statistics. You can use any file name that your operating system can support. The default file type is `.csv`. The performance information is saved as a comma-delimited file.

Before you use the `save storageArray performanceStats` command, run these commands to specify how often statistics are collected.

*   `set session performanceMonitorInterval`

*   `set session performanceMonitorIterations`

The `performanceMonitorInterval` parameter defines the frequency of gathering performance data. Enter an integer value for the polling interval, in seconds, for which you want to capture data. The range of values is 3 to 3600 seconds. The default value is 5 seconds. Use this parameter when you want to change the frequency for gathering performance data.

The `performanceMonitorIterations` defines the number of samples to capture. Enter an integer value. The range of values for samples captured is 1 to 3600. The default value is 5. Use this parameter when you want to redefine the number of samples to capture.

The polling interval and the number of iterations that you specify remain in effect until you end the session. After you end the session, the polling interval and the number of iterations return to the default values.

Changing the RAID levels

When you create a volume group, you can define the RAID level for the volumes in that volume group. You can change the RAID level later to improve performance or provide more secure protection for your data.

> **Note:** RAID levels do not apply to dynamic disk pools (DDP), which use RAID level 6 only.

> **Note:** This operation may take days or weeks to complete, and once launched, cannot be terminated except by deleting the impacted volume group, along with all associated volumes and data. Some maintenance operations, such as upgrading controller firmware or NVSRAM, are prohibited until this operation completes.

To change the RAID level, use this command:

```
set volumeGroup [volumeGroupNumber]
raidLevel=(0 | 1 | 3 | 5 | 6)
```

In this command, `volumeGroupNumber` is the number of the volume group for which you want to change the RAID level.

Alternatively, you can use the command with a volume group name instead of a volume group number, as shown:

```
set volumeGroup [volumeGroupName]
raidLevel=(0|1|3|5|6)
```

Changing the segment size

When you create a new volume, you can define the segment size for that volume. In addition, you can change the segment size later to optimize performance.

In a multiuser database or file system storage environment, set your segment size to minimize the number of drives that are needed to satisfy an I/O request. Use larger values for the segment size. Using a single drive for a single request leaves other drives available to simultaneously service other

requests. If the volume is in a single-user large I/O environment, performance is maximized when a single I/O request is serviced with a single data stripe; use smaller values for the segment size.

> **Note:** This operation may take days or weeks to complete, and once launched, cannot be terminated except by deleting the impacted volume group, along with all associated volumes and data. Some maintenance operations, such as upgrading controller firmware or NVSRAM, are prohibited until this operation completes.

To change the segment size, use this command:

```
set volume ([volumeName] | <"wwID">) segmentSize=segmentSizeValue
```

In this command, `segmentSizeValue` is the new segment size that you want to set, in KB. Valid segment size values are 8 , 16 , 32 , 64 , 128 , 256 , and 512 . In addition, a 4 KB option is available, but only for SSD volume groups.

> **Note:** You are restricted to change the segment size up or down one level from the current size each time you use the command. For example, if the current segment size is 64 KB, you can only change it to 32 KB or 128 KB with the first command. You can then change the segment size up or down one more level if desired, one level each time you use the command.

You can identify the volume by name or by WWID. The volume names must be enclosed in square brackets. If a volume name also has special characters or numbers, you must enclose the volume name in double quotation marks. When using the WWID, enclose the WWID in double quotation marks (" ") inside angle brackets (< >). Do not include colons in the WWID. (For usage information, refer to the `set volume` command in the *Command Line Interface and Script Commands Programming Guide* .)

Changing the cache parameters

There are two kinds of cache used with E-Series storage:

- Controller cache – A physical memory dedicated to I/O operations between controller and hosts and between controller and disk drives.

- SSD cache – a feature that provides a way to improve read-only performance. SSD cache is a set of Solid-State Disk (SSD) drives that you logically group together in your storage array to implement a read cache for end-user volumes.

The script command set provides two commands that you can use to change cache parameter settings:

- `set storageArray`

- `set volume`

The `set storageArray` command lets you change the controller cache block size. The `set volume` command lets you change these controller and SSD cache settings:

- The controller cache without batteries enabled or disabled

- The controller mirror cache enabled or disabled

- The controller read cache enabled or disabled

- The controller write cache enabled or disabled

- The controller cache read Prefetch enabled or disabled

- The SSD cache enabled or disabled

Defragmenting a volume group

When you defragment a volume group, you consolidate the free capacity in the volume group into one contiguous area. Defragmentation does not change the way in which the data is stored on the volumes.

> **Note:** This operation may take days or weeks to complete, and once launched, cannot be terminated except by deleting the impacted volume group, along with all associated volumes and data. Some maintenance operations, such as upgrading controller firmware or NVSRAM, are prohibited until this operation completes.

As an example, consider a volume group with five volumes. If you delete volume 1 and volume 3, your volume group is configured as follows:

```
space, volume 2, space, volume 4, volume 5, original unused space
```

When you defragment this volume group, the space (free capacity) is consolidated into one contiguous location after the volumes. After being defragmented, the volume group appears as follows:

```
volume 2, volume 4, volume 5, consolidated unused space
```

To defragment a volume group, use this command:

```
start volumeGroup [volumeGroupNumber] defragment
```

In this command, *volumeGroupNumber* is the identifier for the volume group.

Troubleshooting and diagnostics

If a storage array exhibits abnormal operation or failures, you can use the commands that are described in this section to help determine the cause of the problem.

Detailed error reporting

Data collected from an error encountered by the CLI is written to a file. Detailed error reporting under the CLI works as follows:

- If the CLI must abnormally end running CLI commands and script commands, error data is collected and saved before the CLI finishes.

- The CLI saves the error data by writing the data to a standard file name.

- The CLI automatically saves the data to a file. Special command line options are not required to save the error data.

- You are not required to perform any action to save the error data to a file.

- The CLI does not have any provision to avoid over-writing an existing version of the file that contains error data.

For error processing, errors appear as two types:

- Terminal errors or syntax errors that you might enter

- Exceptions that occur as a result of an operational error

When the CLI encounters either type of error, the CLI writes information that describes the error directly to the command line and sets a return code. Depending on the return code, the CLI also might write additional information about which terminal caused the error. The CLI also writes information about what it was expecting in the command syntax to help you identify any syntax errors that you might have entered.

When an exception occurs while a command is running, the CLI captures the error. At the end of processing the command (after the command processing information has been written to the command line), the CLI automatically saves the error information to a file.

The name of the file to which error information is saved is `excprpt.txt`. The CLI tries to place the `excprpt.txt` file in the directory that is specified by the system property `devmgr.datadir`. If for any reason the CLI cannot place the file in the directory specified by `devmgr.datadir`, the CLI saves the `excprpt.txt` file in the same directory from which the CLI is running. You cannot change the file name or the location. The `excprpt.txt` file is overwritten every time that an exception occurs. If you want to save the information in the `excprpt.txt` file, you must copy the information to a new file or a new directory.

Collecting all support data

- AutoSupport (ASUP)

- `save storageArray supportData` command

The following table lists the type of support data that you can collect. For the commands that you can use to collect support bundle data, refer to *Command Line Interface and Script Commands Programming Guide* .

Type of data	File name and description

Automatic Load Balancing statistics	`alb-statistics-A.txt`
	
	`alb-statistics-B.txt`
	
	This file, one per controller, specify various statistics related to the Automatic Load Balancing feature that allow further analysis via offline analysis tool.

Core dump metadata	`all-coredump.xml` This file contains core dump metadata for the array. **Note:** Starting with the 8.25 release, core-dump-info.xml has been renamed to all-coredump.xml.
AutoSupport status	`autosupport.xml` This file specifies the current status of ASUP for the given array.
The AutoSupport history file	`autosupport-history.xml` This file specifies all ASUP messages, both standard and AutoSupport OnDemand messages, for the given array. Each entry in the table corresponds to an ASUP message currently sitting in one of the repositories for a given array.
AutoSupport logs	`asup-transmission-logs.txt` This file contains all the transmitted AutoSupport messages.
Web Server Client Info (E2800 only)	`client-info.txt` This file specifies various Web Services client information

Type of data	File name and description
Drive cabling topology	`connection.txt` This file contains information about the connections between the drive tray environmental services module (ESM) and the controller pair.
Cumulative statistics bundles	`cumulative-drive-vol-stats.xml` This file contains cumulative statistics for the controllers.
Misbehaving drive log	`controller-drive-error-event-log.txt` This file contains the storage array controller drive-side error and event log messages.
Controller health image metadata	`core-dump-info.xml` This file contains DPL controller health image information. **Note:** Starting with the 8.25 release, core-dump-info.xml has been renamed to all-coredump.xml.
DOM 0 journal files for controller B (E2800 only)	`dom0-complete-journal-A.7z` This file, one per controller, contains detailed Linux system logs. This includes logs for NetApp Linux packages and standard Linux utilities and services.
DOM 0 journal files for controller B (E2800 only)	`dom0-complete-journal-B.7z` This file, one per controller, contains detailed Linux system logs. This includes logs for NetApp Linux packages and standard Linux utilities and services.
Miscellaneous DOM 0 log files for controller A (E2800 only)	`dom0-misc-logs-A.7z` This file, one per controller, contains system logs that cannot be contained with the Journal. This includes the serial log for the RAID Application and debug logs for the Hypervisor.
Miscellaneous DOM 0 log files for controller B (E2800 only)	`dom0-misc-logs-B.7z` This file, one per controller, contains system logs that cannot be contained with the Journal. This includes the serial log for the RAID Application and debug logs for the Hypervisor.
Drive command aging timeout values	`drive-command-aging-timeout.txt` This file contains the default values and current values for the command aging timeout field for every drive.
Drive health logs	`drive-health-data.bin` This file contains various drive information related to the drive's health. **Note:** This file is a binary file and will require an offline parser to convert to human readable format.
Drive performance analyzer data	`drive-performance-log.txt` This file contains drive performance data that helps you identify any drives that are performing below expectations.
Enterprise Management Window configuration	`emwdata_v04.bin` This file contains the EMW configuration data store file. **Note:** In support bundles for the E2800, this file isn't present.

Type of data	File name and description
Tray event logs	`expansion-tray-log.txt` ESM event logs.
Failed repository analysis	`failed-repository-analysis.txt` This file contains the failed repository analysis information.
Features of the storage array	`feature-bundle.txt` This file contains a list of the number of volumes, drives, and drive trays allowed in the storage array and a list of the features available and their limits.
Firmware inventory	`firmware-inventory.txt` This file contains a list of all of the firmware versions for all of the components in the storage array.
Infiniband interface statistics (Infiniband only)	`ib-statistics.csv` This file contains the Infiniband interface statistics.
I/O path statistics	`io-path-statistics.7z` This file contains raw performance data for each controller that can be used to analyze application performance issues.
IOC dump info for the host interface chip	`ioc-dump-info.txt` This file contains IOC dump information for the host interface chip.
IOC dump logs for the host interface chip	`ioc-dump.gz` This file contains the log dump from the host interface chip on the controller. The file is compressed in gz format. The zip file is saved as a file inside of the Customer Support Bundle.
iSCSI connections (iSCSI only)	`iscsi-session-connections.txt` This file contains a list of all of the current iSCSI sessions.
iSCSI statistics (iSCSI only)	`iscsi-statistics.csv` This file contains statistics for the Ethernet media access control (MAC), Ethernet Transmission Control Protocol (TCP)/Internet Protocol (IP), and iSCSI target.
iSER interface statistics (iSER only)	`iser-statistics.csv` This file contains the statistics for the host interface card that runs iSER.
Major event log	`major-event-log.txt` This file contains a detailed list of events that occur on the storage array. The list is stored in reserved areas on the disks in the storage array. The list records configuration events and component failures in the storage array.
Manifest file	`manifest.xml` This file contains a table that describes the files included in the archive file and the collected status of each of those files.

Type of data	File name and description
Storage management software runtime information	`msw-runtime-info.txt` This file contains the storage management software application runtime information. Contains the JRE version currently used by the storage management software.
NVSRAM data	`nvsram-data.txt` This controller file specifies the default settings for the controllers.
Object bundle	`object-Bundle` This bundle contains a detailed description of the status of your storage array and its components, which was valid at the time that the file was generated.

Summary performance statistics	`perf-stat-daily-summary-a.csv`
	`perf-stat-daily-summary-b.csv`
	
	This file contains various controller performance statistics, one file per controller.

Persistent reservations and registrations	`persistent-reservations.txt` This file contains a detailed list of volumes on the storage array with persistent reservations and registrations.
Storage management software user preferences	`pref-01.bin` This file contains the user preference persistent data store. **Note:** In support bundles for the E2800, this file isn't present.
Recovery Guru procedures	`recovery-guru-procedures.html` This file contains a detailed list of all of the recovery guru topics that are issued in response to problems detected on the storage array.
Recovery profile	`recovery-profile.csv` This file contains a detailed description of the latest recovery profile record and historical data.
SAS PHY error logs	`sas-phy-error-logs.csv` This file contains the error information for SAS PHY.
State capture data	`state-capture-data.txt` This file contains a detailed description of the current state of your storage array.
Storage array configuration	`storage-array-configuration.cfg` This file contains a detailed description of the logical configuration on your storage array.
Storage array profile	`storage-array-profile.txt` This file contains a description of all of the components and properties of a storage array.
Trace buffer contents	`trace-buffers.7z` This file contains the contents of the controllers' trace buffers that are used to record debug information.

Type of data	File name and description
Tray capture data	`tray-component-state-capture.7z` If your tray contains drawers, the diagnostic data is archived in this zipped file. The Zip file contains a separate text file for each tray that contains drawers. The Zip file is saved as a file inside of the Customer Support Bundle.
Unreadable sectors	`unreadable-sectors.txt` This file contains a detailed list of all of the unreadable sectors that have been logged to the storage array.

Web Services Trace Log (E2800 only)	`web-server-trace-log-A.7z`
	`web-server-trace-log-B.7z`
	
	This file, one per controller, contains Web Services trace buffers that are used to record debug information.

Workload capture analytics log file	`wlc-analytics-a.lz4`
	`wlc-analytics-b.lz4`
	
	This file, one per controller, contains computed key workload characteristics such as LBA histogram, read/write ratio and I/O throughput across all active volumes.

X-header data file	`x-header-data.txt` This AutoSupport message header consists of plain text key-value pairs; which include information about the array and message type.

Collecting drive data

To gather information about all of the drives in a storage array, use the `save allDrives` command. This command collects sense data and saves the data to a file. The sense data consists of statistical information that is maintained by each of the drives in the storage array. When you have collected the file, send it to your Technical Support Representative.

Diagnosing a controller

The `diagnose controller` command provides these tests that help you make sure that a controller is functioning correctly:

- Loopback drive channel

- Read

- Write

- Data-loopback

- Pattern

The loopback drive channel identifies the drive channels on which you want to run the diagnostic tests. You can either choose to run the diagnostics on all channels or select a specific channel on which to run diagnostics. If you select a specific channel, valid values for the drive channels are `1`, `2`, `3`, `4`, `5`, `6`, `7`, or `8`.

The read test initiates a read command as it would be sent over an I/O data path. The read test compares data with a known, specific data pattern, and the read test checks for data integrity and errors. If the read command is unsuccessful or the data compared is not correct, the controller is considered to be in error and is placed offline.

The write test initiates a write command as it would be sent over an I/O data path to the diagnostics region on a specified drive. This diagnostics region is then read and compared to a specific data pattern. If the write fails or the data compared is not correct, the controller is considered to be in error, and it is failed and placed offline.

Run the data-loopback test only on controllers that have connections between the controller and the drives. The test passes data through each controller drive-side channel, out onto the loop, and back again. Enough data is transferred to determine error conditions on the channel. If the test fails on any channel, this status is saved so that it can be returned if all of the other tests pass.

For best results, run all three tests after you first install the storage array and any time that you that have made changes to the storage array or the components that are connected to the storage array (such as hubs, switches, and host adapters).

The test results contain a generic, overall status message and a set of specific test results. Each test result contains these items:

- Test (read, write, or data loopback)

- Port (read or write)

- Level (internal or external)

- Status (pass or fail)

Events are written to the Event Log when the diagnostics are started and when testing is completed. These events help you to evaluate whether diagnostics testing was successful or failed and the reason for the failure.

Running read link status diagnostics

> **Note:** This service operation and related commands are only for legacy Fibre Channel storage arrays.

Read link status (RLS) error counts refer to link errors that have been detected in the traffic flow of a Fibre Channel loop. The errors detected are represented as a count (32-bit field) of error occurrences that are accumulated over time. The counts provide coarse measure of the integrity of the components and devices on the loop. By analyzing the error counts that are retrieved, you can determine the components or devices within the Fibre Channel loop that might be experiencing problems communicating with the other devices on the loop. A high error count for a particular component or device indicates that it might be experiencing problems and should be given immediate attention.

Error counts are calculated from the current baseline. The baseline describes the error count values for each type of device in the Fibre Channel loop, either when the controller goes through its start-of-day sequence or when you reset the baseline. The baseline indicates the difference in error counts from the time the baseline was established to the time you request the read link status data.

The script command set provides two commands for running RLS diagnostics:

- `reset storageArray RLSBaseline` – Resets the RLS baseline for all devices by setting all of the counts to 0.

- `save storageArray RLSCounts` – Saves the RLS counters to a file that you can review later. The default file name is `readLinkStatus.csv`.

Run the `reset storageArray RLSBaseline` command before you run the `save storageArray RLSBaseline` command.

The following table lists the type of data contained by the file that is generated by the `save storageArray RLSBaseline` command.

Type of Data	Description
Devices	A list of all devices on the Fibre Channel loop. The devices appear in channel order. Within each channel, the devices are sorted according to the device position in the loop.
Baseline time	The date and time when the baseline was set.
Elapsed time	The time that has elapsed from when the baseline time was set to when the read link status was gathered.
Invalid transmission word (ITW)	The total number of ITW errors that were detected on the Fibre Channel loop from the baseline time to the current date and time. ITW might also be referred to as the Received Bad Character Count. ITW counts indicate that in decoding a read/write transmission, the mapping did not exist and the running disparity of the transmission word is invalid. This data is the key error count to be used when analyzing the error count data.
Link failure (LF)	The total number of LF errors that were detected on the Fibre Channel loop from the baseline time to the current date and time. An LF condition is either a link fault signal, a loss of signal, or a loss of synchronization condition. The LF signal indicates a failure with the media module laser operation.
Loss of synchronization (LOS)	The total number of LOS errors that were detected on the Fibre Channel loop from the baseline time to the current date and time. LOS errors indicate that the receiver cannot acquire symbol lock with the incoming data stream due to a degraded input signal. If this condition persists, the number of LOS errors increases.
Loss of signal (LOSG)	The total number of LOSG errors that were detected on the Fibre Channel loop from the baseline date to the current date and time. LOSG errors typically indicate a loss of signal from the transmitting node or the physical component within the Fibre Channel loop. Physical components where a loss of signal typically occurs include the gigabit interface converters (GBICs), the Small Form-factor Pluggable (SFP) transceivers, and the Fibre Channel fiber-optic cable.

Type of Data	Description
Primitive sequence protocol (PSP)	The total number of PSP errors that were detected on the Fibre Channel loop from the baseline date to the current date and time. PSP refers to the number of N_Port protocol errors that were detected and Link Reset Response (LRR) primitive sequences that were received while the link is up. An LRR is issued by another N_Port in response to a link reset. An N_Port is a Fibre Channel-defined port at the end of a link, such as a server or a workstation. Each port can act as an originator or a responder (or both) and contains a transmitter and receiver. Each port is given a unique name, called an N_Port or an N_Port identifier. If an N_Port is connected to a loop, it becomes an NL_Port. An NL_Port is a Fibre Channel controller ID in a hexadecimal number. The hexadecimal number varies depending on the topology: • For a private arbitrated loop, the ID is a 1-byte arbitrated loop physical address (ALPA). • For all other arbitrated loops, it appears as a single 24-bit hexadecimal number (a triplet of domain, area, and ALPA where each field is 1 byte). • For fabric and point-to-point, the ID is a 3-byte hexadecimal number used in the DID and SID (destination identifier and source identifier) fields of Fibre Channel frames.
Invalid cyclic redundancy check (ICRC)	The total number of ICRC errors that were detected on the Fibre Channel loop from the baseline date to the current date and time. An ICRC count indicates that a frame has been received with an invalid cyclic redundancy check value. A cyclic redundancy check reads the data, calculates the cyclic redundancy check character, and compares the calculated cyclic redundancy check character with a cyclic check character already present in the data. If they are equal, the new data is presumed to be the same as the old data. If the calculated characters and the old characters do not match, an error is posted, and the data is re-sent.

Interpreting the RLS Results

The way that you interpret the RLS results is based on the concept that the device immediately following the problematic component will have the largest number of invalid transition word (ITW) error counts. The process is to obtain the ITW count for every component and device on the loop, analyze the data in loop order, and identify any large increases in the ITW counts.

Important:

The current error counting standard for when to calculate the ITW error count is not well defined. Different vendor devices calculate at different rates. Analysis of the data must take this discrepancy into consideration.

Recovery operations

Recovery operations include repairing the storage array and returning it to an operational state. This might involve replacing a failed canister, a failed controller, a failed drive, restoring data, or changing the operational mode of the storage array. For information about when it is appropriate to replace a canister, see *Replacing canisters* on page 75.

Setting the controller operational mode

A controller has three operational modes:

- Online

- Offline

- Service

Placing a controller online sets it to the Optimal state and makes it active and available for I/O operations. Placing a controller offline makes it unavailable for I/O operations and moves its volume groups to the other controller if failover protection is enabled.

Taking a controller offline can seriously impact data integrity and storage array operation.

- If you do not use write cache mirroring, data in the cache of the controller you place offline is lost.

- If you take a controller offline and you have controller failover protection through a host multi-path driver, the other controller in the pair takes over. Volume groups and their associated volumes that were assigned to the offline controller are automatically reassigned to the remaining controller. If you do not have a multi-path driver installed on the application host and you take a controller offline while the application is using associated volumes, application errors will occur.

   Attention: Possible loss of data access – Placing a controller offline can cause loss of data.

Use Service mode to replace canisters, such as a controller. Placing a controller in Service mode makes it unavailable for I/O operations and moves its volume groups to the second controller without affecting the preferred path of the volume group. This action might significantly reduce performance. The volume groups are automatically transferred back to the preferred controller when it is placed back online.

If you change a controller to Service mode while an application is using the associated volumes on the controller, the change causes I/O errors unless a multi-path driver is installed on the host. Before you place a controller in Service mode, make sure that the volumes are not in use, or a multi-path driver is installed on all of the hosts that are using these volumes.

In addition, if you do not have a multi-path driver, you must make appropriate operating system-specific modifications to make sure that the volume groups moved are accessed on the new path when you change to Service mode.

   Important: Place a controller in Service mode only under the direction of a technical support Representative.

To change the operational mode of a controller, use this command:

```
set controller [(a | b)] availability=(online | offline | serviceMode)
```

Changing the controller volume ownership

You can change which controller is the owner of a volume by using the `set volume` command. The command takes this form:

```
set (allVolumes | volume [volumeName] | volumes [volumeName1 ...
volumeNameN] | volume <"wwID">) owner=(a | b)
```

You can identify the volume by name or by WWID. The volume names must be enclosed in square brackets. If a volume name also has special characters or numbers, you must enclose the volume name in double quotation marks. When using the WWID, enclose the WWID in double quotation

marks (" ") inside angle brackets (< >). Do not include colons in the WWID. (For usage information, refer to the `set volume` command in the *Command Line Interface and Script Commands Programming Guide* .)

Initializing a drive

> **Attention: Possible loss of data access** – When you initialize a drive, all data on the drive is lost.

You must initialize a drive when you have moved a drive that was previously part of a multidisk volume group from one storage array to another. If you do not move the entire set of drives, the volume group information and the volume information on the drives that you move are incomplete. Each drive that you move contains only part of the information that is defined for the volume and the volume group. To be able to reuse the drives to create a new volume group and volume, you must delete all of the old information from the drives by initializing the drive.

When you initialize a drive, all of the old volume group information and volume information are deleted, and the drive is returned to an unassigned state. Returning a drive to an unassigned state adds unconfigured capacity to a storage array. You can use this capacity to create additional volume groups and volumes.

To initialize a drive, use this command:

```
start drive [trayID,drawerID,slotID] initialize
```

In this command, `trayID`, `drawerID`, and `slotID` are the identifiers for the drive. These identifiers support both high-capacity drive trays and low-capacity drive trays. A high-capacity drive tray has drawers that hold the drives. The drawers slide out of the drive tray to provide access to the drives. A low-capacity drive tray does not have drawers. For a high-capacity drive tray, you must specify the identifier (ID) of the drive tray, the ID of the drawer, and the ID of the slot in which a drive resides. For a low-capacity drive tray, you need only specify the ID of the drive tray and the ID of the slot in which a drive resides. For a low-capacity drive tray, an alternative method for identifying a location for a drive is to specify the ID of the drive tray, set the ID of the drawer to `0` , and specify the ID of the slot in which a drive resides.

Reconstructing a drive

You use drive reconstruction after a drive replacement to return a Degraded volume group or disk pool to an Optimal status. The data that you reconstruct is the data as it would appear on the failed drive or drives.

> **Attention:** If your volume group or disk pool has failed, use this command only under the direction of technical support. You may use this command without technical support for a degraded volume group or disk pool.

RAID level	Number of failed drives to cause a Degraded status	Number of failed drives to cause a Failed status
0	1 drive, reconstruction not possible	1 drive, reconstruction not possible
1	1 drive	2 drives of the same mirrored pair
3 or 5	1 drive	2 drives
6	1 or 2 drives	3 drives
Disk pool	1 or 2 drives	3 drives in quick succession

To reconstruct a drive, use this command:

```
start drive [trayID,drawerID, slotID] reconstruct
```

In this command, `trayID`, `drawerID`, and `slotID` are the identifiers for the drive. These identifiers support both high-capacity drive trays and low-capacity drive trays. A high-capacity drive tray has drawers that hold the drives. The drawers slide out of the drive tray to provide access to the drives. A low-capacity drive tray does not have drawers. For a high-capacity drive tray, you must specify the identifier (ID) of the drive tray, the ID of the drawer, and the ID of the slot in which a drive resides. For a low-capacity drive tray, you need only specify the ID of the drive tray and the ID of the slot in which a drive resides. For a low-capacity drive tray, an alternative method for identifying a location for a drive is to specify the ID of the drive tray, set the ID of the drawer to 0 , and specify the ID of the slot in which a drive resides.

Initializing a volume

A volume is automatically initialized when you first create it. If the volume fails, you might be required to re-initialize the volume to correct the failure condition.

> **Attention: Loss of data access** – When you initialize a volume, all data on the volume becomes inaccessible to the host. Use this command only under the direction of technical support.

Consider these restrictions when you initialize a volume:

- You cannot cancel the operation after it begins.

- You cannot use this option if any modification operations are in progress on the volume or the volume group.

- You cannot change the cache parameters of the volume while the initialization operation is in progress.

To initialize a volume, use this command:

```
start volume [volumeName] initialize
```

In this command, `volumeName` is the identifier for the volume.

Redistributing volumes

When you redistribute volumes, you return the volumes to their preferred controller owners. The preferred controller ownership of a volume or a volume group is the controller of an active-active pair that is designated to own the volumes. The preferred owner for a volume is initially designated when the volume is created. If the preferred controller is being replaced or undergoing a firmware download, ownership of the volumes is automatically shifted to the other controller. That controller becomes the current owner of the volumes. This change is considered to be a routine ownership change and is reported in the Event Log.

To redistribute volumes to their preferred controllers, use this command:

```
reset storageArray volumeDistribution
```

> **Important:** If you run this command without a multi-path driver on the hosts, stop I/O activity to the volumes to prevent application errors.

Under some host operating systems, you must reconfigure the multi-path host driver. You might also need to make operating system modifications to recognize the new I/O path to the volume.

Replacing canisters

Storage array components such as the controller canisters and the power-fan canisters have a Service Action Allowed indicator light. This indicator light is a blue LED. The Service Action Allowed indicator light helps to make sure that you do not remove a canister before it is safe to do so.

> **Attention: Possible loss of data access** – Never remove a component that has a Service Action Required indicator light on unless the Service Action Allowed indicator light is on.

If a component fails and must be replaced, the Service Action Required indicator light on that canister comes on to indicate that service action is required, provided no data availability dependencies or other conditions exist that dictate the canister should not be removed. The Service Action Allowed indicator light automatically comes on or goes off when conditions change. In most cases, the Service Action Allowed indicator light comes on steadily when the Service Action Required indicator light comes on for the canister.

The ability to remove a canister depends on the data availability dependencies of the controller tray or the controller-drive tray. The Service Action Allowed indicator light does not come on if removing a canister jeopardizes data on the drive trays or current I/O activity. An example of limiting when you can remove a canister is when one controller canister has a Service Action Required indicator light on. You cannot remove the other controller canister (the Service Action Allowed indicator light does not come on), because doing so would jeopardize the data either on the drive trays or transitioning through the controllers.

A less obvious example is when the power supply for the controller canister in slot A has failed, and the controller canister in slot B has failed. Removing the controller canister in slot B before replacing the failed power-fan canister causes the controller canister in slot A to lose power, which results in a loss of data access. This action occurs because power distribution from each power-fan canister is through the controller canister that is physically connected to that power-fan canister.

So, in the preceding example, these actions occur:

*   The power-fan canister has both its Service Action Required indicator light and its Service Action Allowed indicator light on.

*   The controller canister in slot B has only its Service Action Required indicator light on, but its Service Action Allowed indicator light is off.

*   After the failed power-fan canister has been replaced, the Service Action Allowed indicator light comes on for the controller canister in slot B.

The following table shows when the Service Action Allowed indicator light does not come on for each canister (the indicator light is suppressed). An **X** in a table cell indicates that service is not allowed, therefore the Service Action Allowed light does not come on. For example, if the power supply in the power-fan canister in slot A has failed, then replacement of the controller canister in slot B, the interconnect-battery canister, or the power-fan canister in slot B is not allowed, which is indicated when the Service Action Allowed indicator light stays off for those canisters.

Description of Failure or Circumstance	Canister Description			
	Controller in Slot A	**Controller in Slot B**	**Power-Fan in Slot A**	**Power-Fan in Slot B**
The controller canister in slot A has failed or is locked down.		X		
The controller canister in slot B has failed or is locked down.	X			

Description of Failure or Circumstance	Canister Description			
	Controller in Slot A	Controller in Slot B	Power-Fan in Slot A	Power-Fan in Slot B
The controller canister in the slot A drive path is unavailable.		X		
The controller canister in the slot B drive path is unavailable.	X			
The controller canister in slot A has been removed.		X		
The controller canister in slot B has been removed.	X			
The power supply in the power-fan canister in slot A has failed.				X
A fan in the power-fan canister in slot A has failed or has no input power.				X
The power-fan canister in slot A has been removed.				X
The power supply in the power-fan canister in slot B has failed.			X	
A fan in the power-fan canister in slot B has failed or has no input power.			X	
The power-fan canister in slot B has been removed.			X	

About AutoSupport and AutoSupport OnDemand

ASUP collects support data to report configuration, subsystem status, and exceptions in near-real time. ASUP OnDemand (AOD) is an optional extension to ASUP that introduces two-way communication between the E-Series array and the ASUP backend. ASUP RemoteDiagnostics provides the ability for NetApp technical support to remotely ask the given E-Series array to create a new ASUP message, or request retransmission of a particular ASUP message.

> **Important:** Under no circumstances is business data from the storage array ever sent to NetApp support.

ASUP messages typically include a collection of system log files, configuration data (formatted XML and unstructured command output), state data (subsystem up/down, capacity used), performance metrics, and system inventory data. All of the data gathered is collected into a single compressed archive file format (7z).

The storage array uses the Internet to send ASUP messages to the ASUP backend. The ASUP backend provides near-real time access to the messages by technical support. ASUP requires compliance to the following transport protocol-specific requirements:

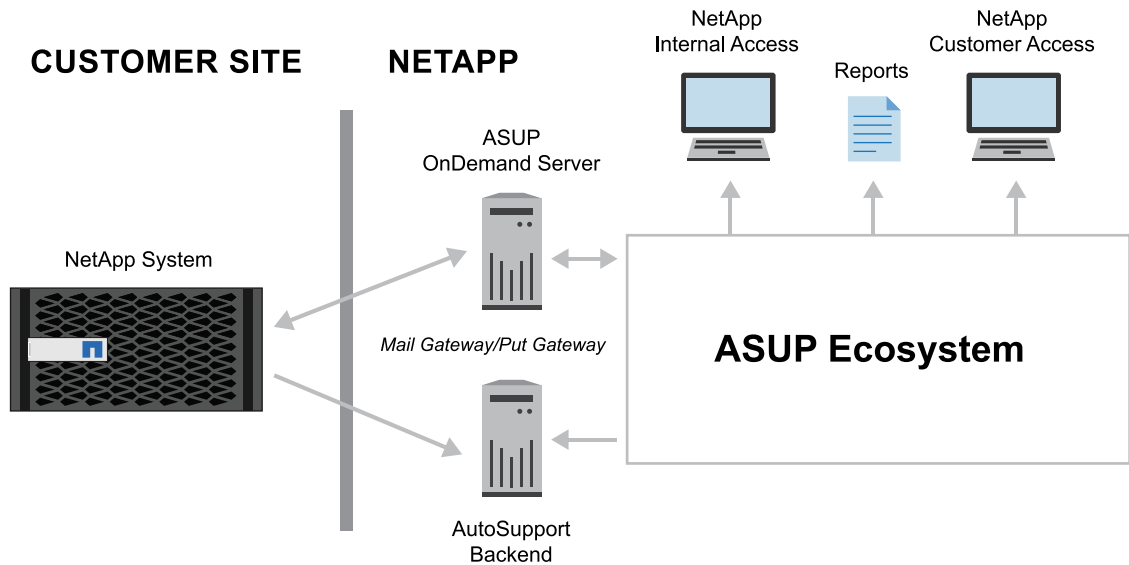- HTTP or HTTPS upload

- SMTP notifications

Methods for collecting support data

With the implementation of ASUP and AOD, you have three possible methods for collecting support data in a storage array:

*   Standard ASUP collection - Data is automatically collected and sent to technical support, triggered by a daily or weekly schedule, or by an event occurrence that is specified to automatically send a support bundle.

*   ASUP OnDemand - With AOD enabled, before an E-Series system can send any standard ASUP message to the ASUP backend, it must first obtain permission from the ASUP OnDemand Server (AODS). ASUP OnDemand also introduces the ability for a given E-Series array to check in periodically with the AODS to see if there are any pending actions for the particular E-Series array.

*   Legacy support bundle collection (E2700, E5600, or EF560 storage arrays only) - Collection of legacy support bundle data is user-configurable at intervals you schedule. You can then manually send the support bundles to technical support.

For the E2700, E5600, or EF560 storage arrays only, ASUP operations and legacy support bundle operations are mutually exclusive on a given storage array. When you turn on ASUP, with or without AOD, you automatically disable legacy support bundle collection. If you want to run legacy support bundle collection, you must turn off ASUP. Because ASUP speeds up troubleshooting and problem analysis, ASUP is the preferred data collection method to use if available on the storage array.

AutoSupport components

AutoSupport component	Description
NetApp System	The NetApp System is an array on the customer site. The system is managed by the SANtricity software on a management station or host, and the Persistent Monitor runs on the management station or host. **Note:** The NetApp System stores an AutoSupport log, located in the ASUPLog directory on the array. The log provides information about status, history of transmission, and any errors encountered during the collection and delivery of the AutoSupport messages.

AutoSupport component	Description
ASUP OnDemand Server (AODS)	With AOD enabled, before an E-Series system can send any standard ASUP message to the ASUP backend, it must first obtain permission from the AODS. **Important:** Under no circumstances does the AODS initiate communication with an E-Series array. The flow of communication always begins with the E-Series array, and a reply is sent back from the AODS. The reply might contain actions requested by technical support, such as to retransmit a missing ASUP message.
Mail Gateway/Put Gateway	The HTTP(s) Put/Post Gateway moves the messages into the ASUP Ecosystem.
ASUP Ecosystem	The ASUP Ecosystem provides near real-time access to messages by NetApp technical support. Additionally, there is a portal within the ecosystem that NetApp customers can use to access their ASUP data. Finally, the ASUP Ecosystem provides the ability to generate reports used by various groups within NetApp.

AutoSupport data processing

The backend processing of AutoSupport data is used in several ways:

- Routing messages based on message content

- Support case generation

- Proactive health-check and system level risk detection

- Service entitlement

- Subscription-based service fulfillment

- Persisting the support information associated with the ASUP message

- Data analysis to improve the E-Series products

AutoSupport configuration in E-Series storage arrays

The E2800 storage array includes embedded ASUP functionality, rather than relying on an externally installed Event Monitor to configure and monitor AutoSupport (as does the E2700, E5600, and EF560 storage arrays). Because of the differences in the E2800 storage array, there are different management methods for configuring AutoSupport across the two kinds of arrays:

- For the E2800 storage array, you can use SANtricity System Manager or the RESTful API to configure AutoSupport. There are sample programs in the Java and Python client libraries which are part of the REST API SDK.

- For the E2700, E5600, and EF560 storage arrays, you can use the SANtricity Storage Manager application or SMcli.

Standard AutoSupport Messaging

Standard AutoSupport (ASUP) messaging is used to communicate between an E-Series system and the NetApp E-Series ASUP Ecosystem. These messages are accompanied by storage array support bundles that include data tailored to the particular message type.

Standard ASUP message types

The following standard ASUP message types are used:

- Event ASUP Messages

  ◦ Triggered/originated whenever a MEL event set to trigger the automatic collection of a support bundle.

  ◦ Includes minimal system configuration information

  ◦ Includes detailed diagnostic/debug information

- Weekly ASUP Messages

  ◦ Includes configuration and system state information

- Daily ASUP Messages

  ◦ Provides current set of system event logs and performance data

  ◦ Places less burden on payload and transmission on the messages originating from Event ASUP messages

Scheduling of daily and weekly standard ASUP messages

Note the following with regard to daily and weekly message collection schedules:

- If ASUP is enabled, the management software sends both daily AutoSupport messages and weekly AutoSupport messages.

- The management software randomly selects a time of day for both daily and weekly messages and a day of the week for weekly messages. Management software makes all attempts to ensure that no two storage arrays within a management domain send scheduled AutoSupport messages at the same time.

- Using the `SMcli -autoSupportSchedule reset` command, you can specify a range (hour-level granularity) of preferred times to send both daily and weekly messages for the storage arrays in your management domain. For the weekly messages schedule, you use the command to select preferred days of the week.

Standard ASUP messaging process

When AutoSupport OnDemand is not enabled, the standard ASUP messaging process occurs as described in the following graphic and table:
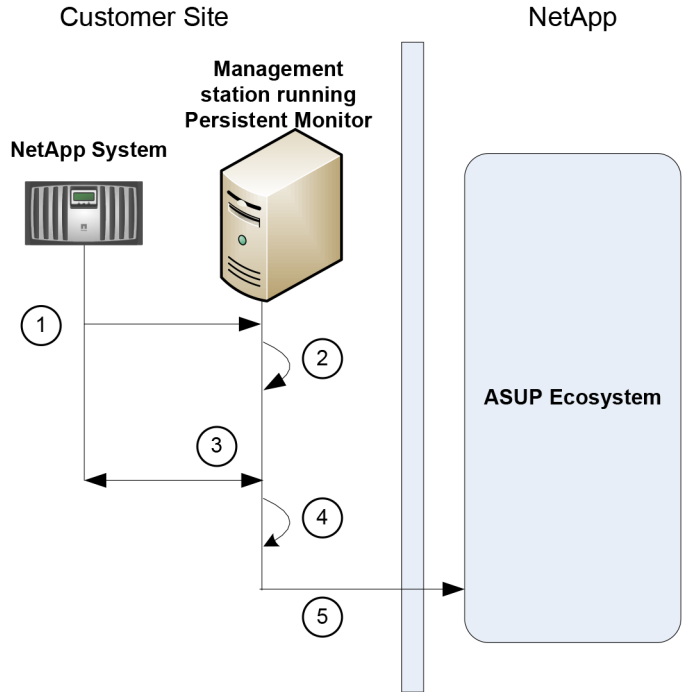
Step	Description
1	An E-Series array sends a report to the Persistent Monitor that a MEL with the "Collect Support Bundle" flag set to TRUE, OR -the Persistent Monitor determines it is time to send a daily or weekly ASUP message.
3	The Persistent Monitor requests a support bundle with the appropriate data subset, and the E-Series array returns the data.
4	The Persistent Monitor takes the collected support bundle and builds the ASUP message.
5	The Persistent Monitor initiates the dispatch of the Event, Weekly, or Daily ASUP message through the configured protocol (HTTPS, HTTP, or SMTP).

AutoSupport OnDemand Messaging

AutoSupport OnDemand (AOD) is an extension to the AutoSupport feature that allows the AutoSupport Ecosystem to determine when ASUP messages can be delivered from E-Series arrays in the field. The AOD components shift control of ASUP delivery to the backend servers and, with Remote Diagnostics, enable technical support to request the generation of new ASUP messages or the retransmission of missing ASUP messages during array troubleshooting.

Messaging with AutoSupport OnDemand

With AutoSupport OnDemand, two additional message types, beyond the standard message types, are introduced:

- AOD Trigger Query messages - The Persistent Monitor, on behalf of an E-Series system, must obtain permission from the AutoSupport OnDemand Server (AODS) before dispatching any ASUP messages. The AOD Trigger Query message is the mechanism by which the permission is obtained.

- AOD Poll messages - the SANtricity client periodically "checks in" with the ASUP OnDemand server. These messages are called AOD Poll messages.

AOD Trigger Query Messages

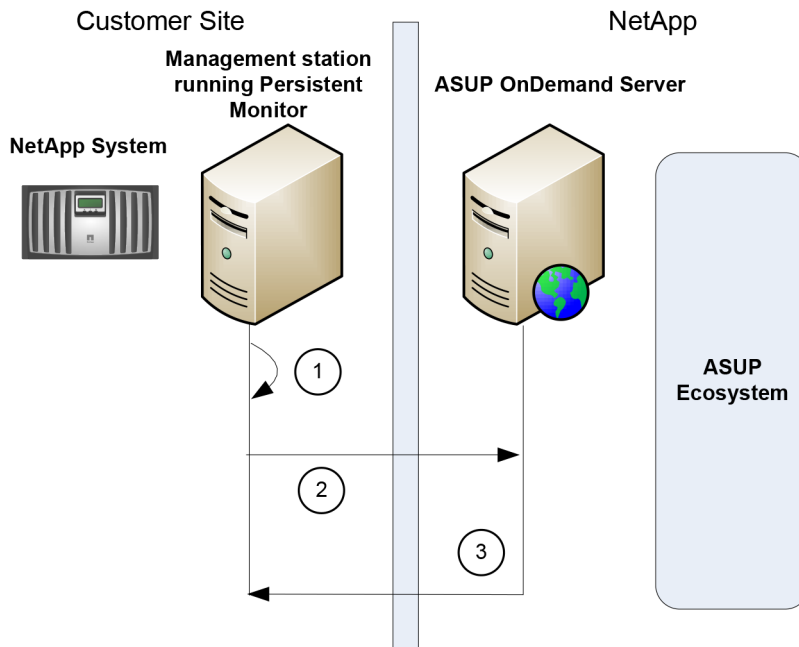The following graphic and table describe the AOD Trigger Query process.
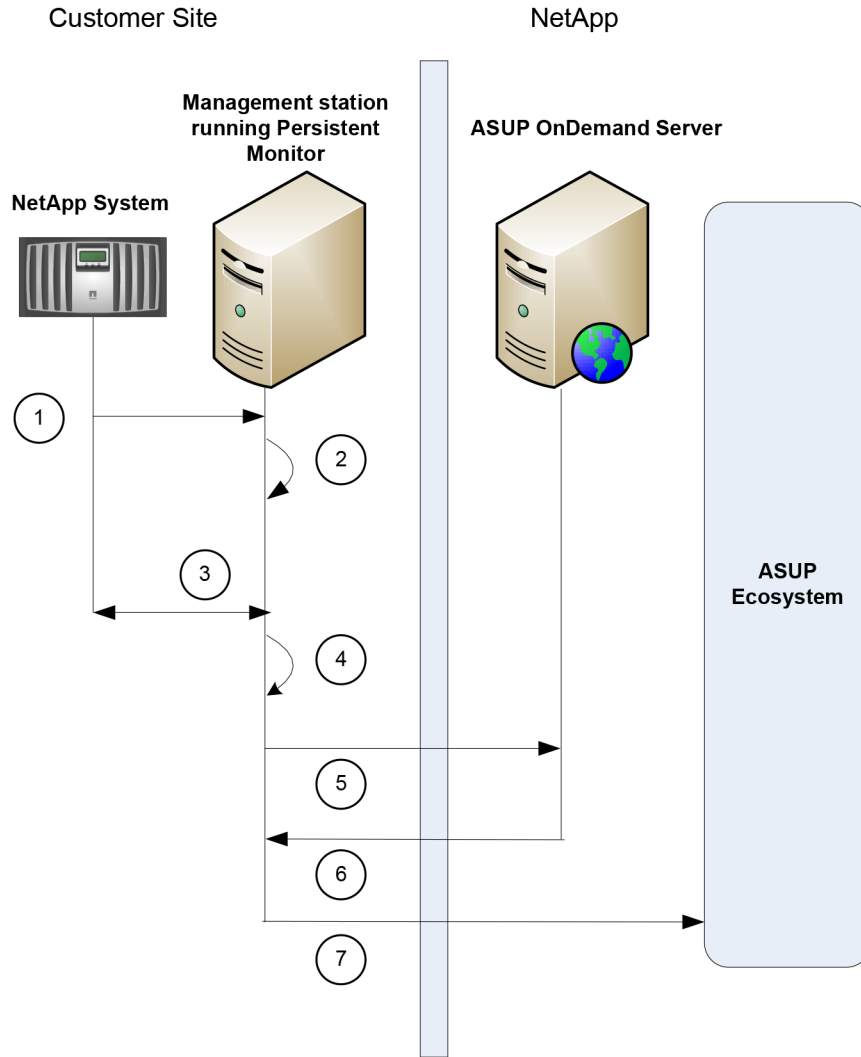
Step	Description
1	An E-Series array sends a report to the Persistent Monitor that a MEL with the "Collect Support Bundle" flag set to TRUE, OR - the Persistent Monitor determines it is time to send a daily or weekly ASUP message.
2	The Persistent Monitor establishes an interactive session with the storage array to collect a support bundle with the appropriate subsets of support data included.
3	The Persistent Monitor builds an ASUP message with the collected support bundle.
4	The Persistent Monitor generates a trigger query.
5	The Persistent Monitor dispatches it via HTTPS/POST to the URL for the AODS.

Step	Description

6	The AODS replies with:
	
	1. A `Deliver` option - values can be set to:
	
	   • **Proceed** - The ASUP message can be transmitted to the default HTTPS gateway server.
	
	   • **Decline** - The ASUP message cannot be sent. The declined ASUP message is stored in the Pending ASUP repository.
	
	   • **Proceed, to alternate URL** - The ASUP message can be transmitted to an alternate HTTPS gateway server within the ASUP ecosystem.
	
	2. Zero or more `Actions` (no action, or more than one action can be specified in a single request).
	
	See *AOD Action Options* on page 83 for more information.
	
	   Note: If the `Deliver` option is set to `Decline`, no `Action` is requested.
	
	   Note: If the Remote Diagnostics option is not enabled, no `Action` is requested.

7	If the `Deliver` option is set to `Proceed` or `Proceed with Alternate URL`, the Persistent Monitor dispatches an appropriate standard ASUP message via the default ASUP HTTPS/POST URL or the alternate URL, as appropriate. The ASUP message is processed and stored in the ASUP Ecosystem.

AOD Poll Messages

The Persistent Monitor, on behalf of the storage array, periodically "checks in" with the AODS. This is an opportunity for a response from the AODS that may include an action, such as a request to retransmit a missing ASUP message.

> **Note:** The polling interval is array-specific, and all arrays start out with the same default polling interval (of one hour). Only technical support can change the polling interval. Technical support may change the polling interval when doing triage or diagnostics of the array.

Step	Description
1	When the polling interval for a given array is reached, the Persistent Monitor generates a history file for the array. The history file contains a list of all available ASUP messages for the array, whether they reside in the Pending or the Delivered repository.
2	The Persistent Monitor generates a Poll AOD message and dispatches it via HTTPS/POST to the URL for the AODS.
3	AODS replies with a response, indicating zero or more `Actions` (no action, or more than one action can be specified in a single request). See *AOD Action Options* on page 83 for more information. **Note:** If the Remote Diagnostics option is not enabled, no `Action` is requested.

AOD Action Options

With AutoSupport OnDemand and AutoSupport remote diagnostics, a response from the AutoSupport OnDemand server to an E-Series array can include an action, such as a request for a new ASUP message to be collected and dispatched, a request for an available controller health image to be dispatched, a request to retransmit a missing ASUP message, or a request to change the polling rate of an array. Note that these action options are all requests from NetApp technical support that are queued up in the ASUP ecosystem, waiting for the next AOD Poll or AOD Trigger message from the array. The actions are sent as a response to the AOD Poll or AOD Trigger message.

Note: To specify these actions, both AutoSupport OnDemand and AutoSupport Remote Diagnostics must be enabled.

Action	Description
Invoke (all or test)	Informs the Persistent Monitor that a new ASUP message needs to be collected or dispatched. The action specifies either `all` or `test`.
Invoke a controller health image file	Informs the Persistent Monitor to collect an available controller health image file and send it with a new ASUP message. A case number is included in the action.
Retransmit	Informs the Persistent Monitor to collect an ASUP from the Pending or Delivered repository, with a particular sequence number.
Polling Rate	Informs the Persistent Monitor to change the polling rate for poll messages. The default for each storage array is once per hour. The new polling rate is specified.

Example Action - Changing the Polling Rate

The following graphic and table illustrate the sequence of steps to change the polling rate of an array. The sequence for other actions is similar.
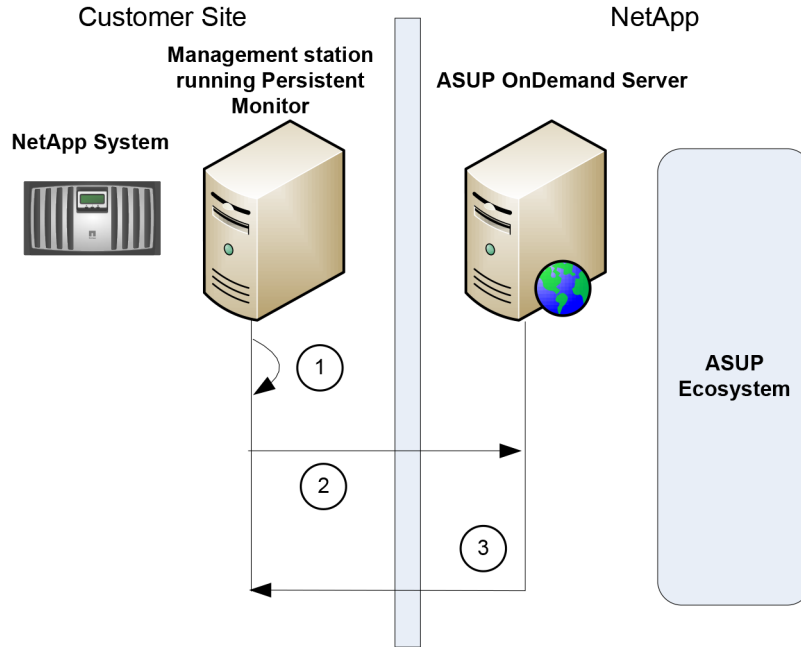
Step	Description
1	The technical support engineer, using tools in the ASUP Ecosystem, requests a change to the polling rate for an array.
2	The ASUP Ecosystem forwards the request to the ASUP OnDemand Server (AODS). AODS queues the request, waiting for the Persistent Monitor to check in via a Trigger AOD message or a Poll AOD message for the array.
3	The polling interval for the given array is reached, and the Persistent Monitor generates a history file for the array.
4	The Persistent Monitor generates a Poll AOD message and dispatches it via HTTPS/POST to the URL for the AODS.
5	AODS replies with a response that includes the Action of Polling Rate.
6	The Persistent Monitor validates the new polling rate.
7	The new polling rate for the array is stored on the array and immediately goes into effect.

AutoSupport commands

See the *SANtricity 11.30 Command Line Interface and Script Commands Programming Guide* for more information on using these commands.

ASUP SMcli commands

Important: These commands are SMcli commands, not script commands. You must run these commands from a command line. You cannot run these commands from the script editor in the storage management software. These commands impact all ASUP capable arrays in the management domain.

`SMcli enable autoSupportFeature`	Enable AutoSupport at the Enterprise Management Window domain level.

`SMcli disable autoSupportFeature`	Disable AutoSupport at the Enterprise Management Window domain level.
`SMcli autoSupportConfig show`	Display the AutoSupport configuration.
`SMcli -autoSupportSchedule show`	Display AutoSupport message collection schedule for daily and weekly scheduled ASUP messages.
`SMcli -autoSupportConfig deliveryMethod`	Specify the AutoSupport delivery method (HTTP/HTTPS, SMTP). **Note:** You must specify HTTPS to use AutoSupport OnDemand, and subsequently AutoSupport Remote Diagnostics.
`reset autoSupportSchedule dailyTime=<startTime>-<endTime> dayOfWeek=[Sunday|Monday|Tuesday| Wednesday|Thursday|Friday| Saturday] weeklyTime=<startTime>-<endTime>`	Reset AutoSupport message collection schedule for daily and weekly scheduled ASUP messages.
`SMcli autoSupportConfig test`	Send test ASUP message.
`SMcli -autoSupportLog (all|-n storageArrayName" | -w"wwID") inputArchive=n outputLog=filename`	Capture or view an AutoSupport log.

Additional ASUP commands used for ASUP OnDemand

`SMcli enable autoSupportOnDemand`	Enable the AutoSupport OnDemand feature at the Enterprise Management Window domain level. **Note:** The AutoSupport feature must be enabled before this feature can be enabled.
`SMcli enable autoSupportRemoteDiag`	Enable the AutoSupport OnDemand remote diagnostics feature at the Enterprise Management Window domain level. **Note:** The AutoSupport feature and the AutoSupport OnDemand feature must both be enabled before this feature can be enabled.
`SMcli disable autoSupportOnDemand`	Disable the AutoSupport OnDemand feature at the Enterprise Management Window domain level.
`SMcli disable autoSupportRemoteDiag`	Disable the AutoSupport OnDemand remote diagnostics feature at the Enterprise Management Window domain level.

`SMcli autoSupportConfig test`	Test the AutoSupport OnDemand connectivity.

AutoSupport commands for individual storage arrays

After enabling the AutoSupport feature for the management domain, you have control over the feature enablement for each storage array.

Note: You can run these commands using the command line or the script editor in the storage management software. These commands impact a storage array you specify rather than the entire management domain.

`set storageArray autoSupportFeature enable`	Turn on the ASUP feature for a specific storage array.
`set storageArray autoSupportFeature disable`	Turn off the ASUP feature for a specific storage array.
`save storageArray supportData disable`	Manually save the support data for the array.

Copyright information

Trademark information

NetApp, the NetApp logo, Go Further, Faster, AltaVault, ASUP, AutoSupport, Campaign Express, Cloud ONTAP, Clustered Data ONTAP, Customer Fitness, Data ONTAP, DataMotion, Fitness, Flash Accel, Flash Cache, Flash Pool, FlashRay, FlexArray, FlexCache, FlexClone, FlexPod, FlexScale, FlexShare, FlexVol, FPolicy, GetSuccessful, LockVault, Manage ONTAP, Mars, MetroCluster, MultiStore, NetApp Insight, OnCommand, ONTAP, ONTAPI, RAID DP, RAID-TEC, SANtricity, SecureShare, Simplicity, Simulate ONTAP, Snap Creator, SnapCenter, SnapCopy, SnapDrive, SnapIntegrator, SnapLock, SnapManager, SnapMirror, SnapMover, SnapProtect, SnapRestore, Snapshot, SnapValidator, SnapVault, StorageGRID, Tech OnTap, Unbound Cloud, and WAFL and other names are trademarks or registered trademarks of NetApp, Inc., in the United States, and/or other countries. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such. A current list of NetApp trademarks is available on the web.

http://www.netapp.com/us/legal/netapptmlist.aspx

How to send comments about documentation and receive update notifications

You can help us to improve the quality of our documentation by sending us your feedback. You can receive automatic notification when production-level (GA/FCS) documentation is initially released or important changes are made to existing production-level documents.

If you have suggestions for improving this document, send us your comments by email.

doccomments@netapp.com

To help us direct your comments to the correct division, include in the subject line the product name, version, and operating system.

If you want to be notified automatically when production-level documentation is released or important changes are made to existing production-level documents, follow Twitter account @NetAppDoc.

You can also contact us in the following ways:

- NetApp, Inc., 495 East Java Drive, Sunnyvale, CA 94089 U.S.

- Telephone: +1 (408) 822-6000

- Fax: +1 (408) 822-4501

- Support telephone: +1 (888) 463-8277