SV SAN Builder

Installation

and

User Guide

SUN RELEASE

Document Identification:
Number: 310-606154

Release: 2.5.4
Nov 15, 2001

Copyright

Vicom Systems Inc. 47281 Bayside Parkway Fremont, CA 94538	http://www.vicom.com ph: (510) 743 - 1130 fx: (510) 743 - 1131

Trademarks

CONTENTS

PREFACE

Document Overview

This document describes the installation and operation of SV SAN Builder. This program monitors, services, and configures storage area networks (SANs), which are composed of SV Routers, host adapters, and disk drive subsystems. SV SAN Builder provides both local and remote access to the storage loop and is equipped with a daemon and a Command Line Interface.

Chapter 1 describes the system requirements, the SLIC Daemon, the two user interfaces, and the different types of drives you can create and manipulate with SV SAN Builder.

Chapter 2 describes how to install SV SAN Builder and explains the SLIC partition and the configuration file.

Chapter 3 explains the SV SAN Builder Command Line Interface (CLI), what functions are available, and how to use it.

Related Publications

SV Router FC-FC 3 – Installation and User Guide – Sun Release	prt no. 310-606155
SV Zone Manager – Installation and User Guide – Sun Release	prt no. 310-606156
SV SNMP Agent – Installation and User Guide – Sun Release	prt no. 310-606157
Vicom SVE Service Manual for UNIX – Sun Release	prt no. 310-606187

SV SAN Builder Revision History

Software Version	Date	Document
2.5	Sep 17 2001	Preliminary Release: 2.5.1
2.5	Oct 15 2001	Release: 2.5.2

Service and Support

Please fill out and mail or fax the warranty registration card included with the hardware as soon as possible. Each installation must be registered in order to qualify for technical support.

Vicom provides 24x7x365 support. Customers may call: 1-877-868-4266 or 510-743-1427.

At any time, customers may request support via email at support@vicom.com. Responses to requests will be made during the following business day.

CHAPTER 1

INTRODUCTION TO SV SAN BUILDER

This chapter explains the system requirements and basic drive information necessary for using SV SAN Builder. It includes these sections:

System Requirements

- SUN Solaris® 2.6, 2.7 (7), 8 (server and client CLI application);
- TCP/IP socket support (server and client) with the following socket ports reserved for the SLIC Daemon:
  - Daemon-Application Socket Port Number: 20000;
  - Daemon-Daemon Socket Port Number (for failover): 20001;
  - Daemon-Router Socket Port Number: 25000;
- Network card (10BaseT or 100BaseT);
- Router Microcode version 8.1.0 or later.

SLIC Daemon Overview

The SLIC Daemon runs on the host server and communicates with an SV Router. Once this communication is established, it allows the SV SAN Builder software to communicate with any SV Routers in the SAN and, from there, to monitor the entire SAN.

The SLIC Daemon requires a SignOn Path, which must be defined in the configuration file. The SignOn Path is the communication path between the client, the daemon, and the SV Router. If you are using in-band communication, the SLIC Daemon also requires a SLIC partition (see ‘The Configuration File’ on page 17).

Once the daemon is started, it runs continually in the background until it is stopped. For instructions on starting and stopping the daemon, see ‘SLIC Daemon Setup’ on page 24. While the daemon is running, it automatically polls all errors and writes them to a system file. It also can be configured to email certain errors to a user-defined email address as they are detected.

Caution ! ***Only one active SLIC Daemon is allowed per SAN.***

Command Line Interface

SV SAN Builder supports a command line interface which allows you to:

- Display the physical components of the SAN;
- Display and modify the properties of an SV Router, a disk pool, a virtual drive, or a MultiPath drive;
- Download microcode to the SV Router to which the client is connected;
- Display each of the storage network maps (FC maps);
- Display the Event Log for the subsystem;
- Run a script-based program to monitor or maintain the storage subsystem;
- Display the Vital Product Data of the devices on the storage network;
- Display or print each or all of the storage network maps (FC or SCSI maps);
- Run a diagnostic test on either an SV Router or a disk drive;
- Perform other diagnostic functions, such as setting a master SV Router, clearing the check mode, or displaying a list of SV Routers attached to a particular host.

Drive Information

SV SAN Builder can be used to configure and maintain physical drives, logical drives, and virtual drives on the SAN. Connectivity to the target devices may be either Fibre Channel or SCSI. When viewing a Fibre Channel drive with SV SAN Builder, you can see the Fibre Channel LUN of each drive in the serial/storage loop. When viewing a SCSI drive, however, you can see both a SCSI ID and LUN for each drive in the serial/storage loop.

Physical Drive

This is an individual disk drive that exists in the storage subsystem. Physical drives also can have mapped attributes.

Logical Drive

Logical drives are created from physical drives, have a single ID and LUN, and are viewed by the host as one single drive. They can be simple, mirror, composite, mirror composite, or Instant Copy drives.

Drives added to a composite drive or a mirror drive become “member drives” of that drive.

Simple Drive

Simple drives can be used as spare drives or storage drives, be combined with other simple drives to form logical drives, or they can be added to a disk pool to create standby drives.

Simple drives can be mapped or unmapped. Mapped simple drives have a LUN and can be viewed by the host. Unmapped simple drives are invisible to the host but visible to Vicom software or hardware.

Drive Conditions

- Offline: The drive has been physically removed from the storage subsystem.
- Unresponsive: The drive is connected, but not responding.
- Rejected: The drive is either offline or unresponsive.
- Member: The drive is a physical or logical/composite drive within a logical drive. All logical drives may contain member drives, except general spares.

Device Support Limitations

Do not exceed the following limitations:

- Up to 512 devices/disks per storage subsystem.
- Up to 16 SV Routers as initiators per storage subsystem.
- Up to 64 logical drives (mirror, composite, Instant Copy) per storage subsystem.
- Up to 16 composite drives per storage subsystem, with no more than 128 drive members for all composite drives combined.

Drive Mapping

When mapping is applied to a drive, the drive is given an ID by the host. These mapped drives can be seen by and interact with the host. An unmapped drive is not assigned an ID and is invisible to the host. Mapping can be assigned to all drives except general spares.

Note: If a SCSI host is used, the drive will be mapped with a SCSI ID and LUN; if an FC host is used, then the drive will be mapped with an FC LUN.

Virtual Drive

Virtual drives can be considered the opposite of composite drives. Where composite drives merge multiple drives into one single drive with one LUN, virtual drives are carved out of a larger drive, creating multiple drives. Because each carved drive is assigned its own LUN, this often is referred to as LUN carving, and the drives are called Virtual LUNs.

Only simple or physical spare drives can be used to create virtual drives. Before the drives can be carved, they must be added to a disk pool in a SAN. Each drive in the disk pool then can be carved into as many as 128 virtual drives of at least 0.5 GB each using the Command Line Interface or the Graphical User Interface (GUI).

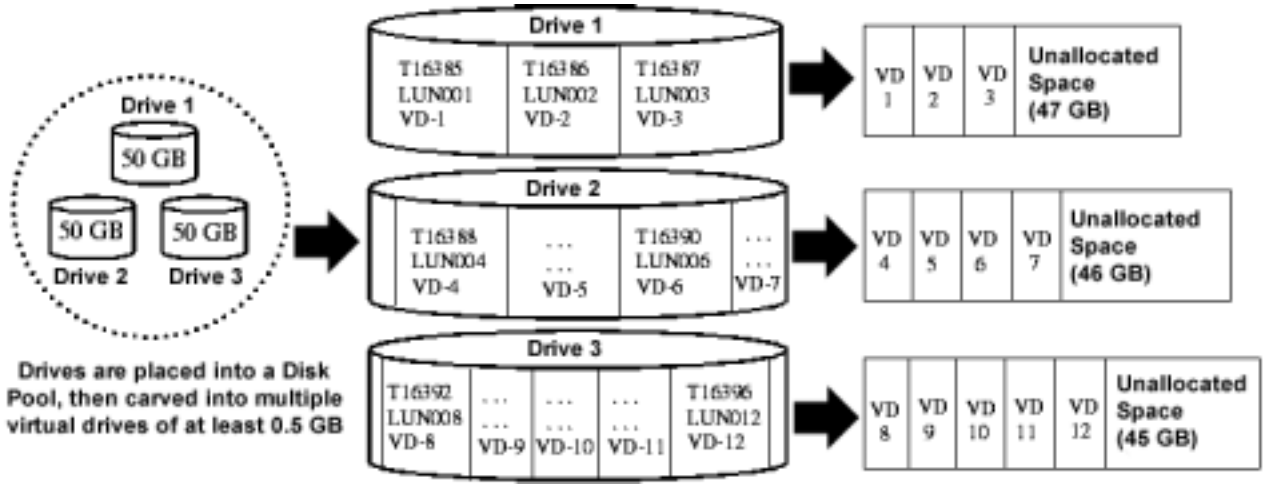


Figure 1-1 Virtual Drives / LUN Carving

MultiPath Drive

Note: MultiPath Drive functionality is supported only in conjunction with the Sun StorEdge™ T3 Array.

A MultiPath Drive is a logical drive created to hide multiple paths into a storage array. Although a single initiator partner-pair storage array, such as the Sun StorEdge™ T3 Array, has only two LUNs, when it is connected to two SV Routers, it is seen as four separate target drives. Two of these drives represent the active paths to each LUN in the storage array, and two represent the passive paths. The passive path is used when the active path is unavailable.

Multiple hosts attempting to access the passive and active paths simultaneously can trigger a failover loop. To prevent this, the paths must be hidden from the host. When you run the MultiPath CLI command, the active and passive paths are combined into a new drive with one LUN. This MultiPath drive contains the path information. The paths still are functioning normally but now are hidden (see Figure 1-2).

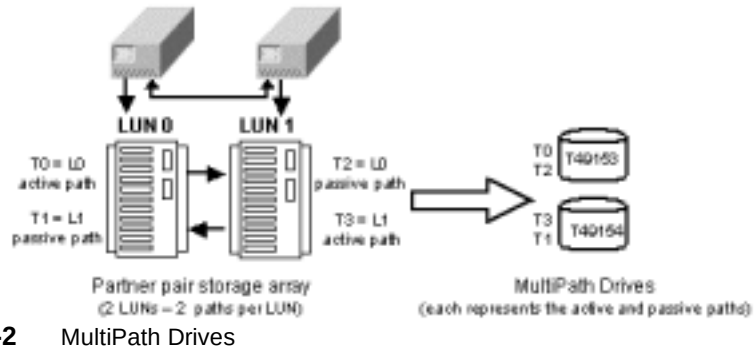


Figure 1-2 MultiPath Drives

Notes and Cautions

1. SV SAN Builder provides management of multiple SV Routers and subsystems by allowing multiple clients (requesting) and multiple servers (supplying) for remote or local computer access.

   - If providing remote access, the client side will run the SV SAN Builder program, and the server side will run the SLIC Daemon (see Figure 1-3).

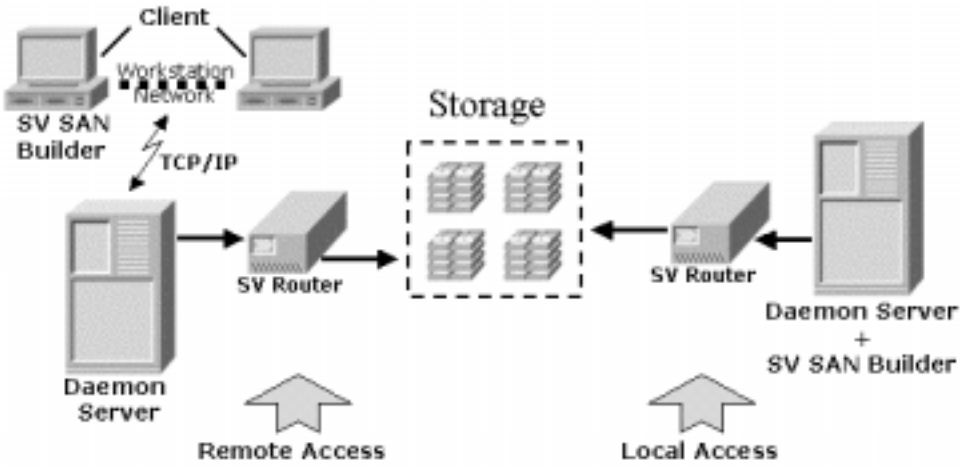


Figure 1-3 Remote or Local Access

   - If providing local access, the host will run both the SV SAN Builder program and the SLIC Daemon (see Figure 1-3).

2. A single SLIC Daemon can run multiple SANs, but only one active daemon is allowed per SAN.

3. Both server and client stations must support TCP/IP socket.

4. Some commands should never be run while there are outstanding I/Os. Running these commands will cause the I/Os to be lost or corrupted, resulting in system panic.

   - Do not download new firmware to the SV Router or to any attached drive that is in use;
   - Do not run the SLIC Diagnostics command line if any attached drives are in use;
   - Do not run the Drive Diagnostics command line if the drive is in use.
   - See Appendix D on page 103 for a complete list of commands that could possibly cause system panic or data loss.

CHAPTER 2

SV SAN BUILDER INSTALLATION

This chapter explains how to set up and install SV SAN Builder. It includes these sections:

Installing SV SAN Builder

In a server directly connected to the Ethernet port of the SV Router, install both the server and the client application of the SV SAN Builder software, edit the configuration file to reference the IP Address of the SV Router, and start the SLIC Daemon (see 'SLIC Daemon Setup' on page 24).

For SV SAN Builder to operate, the SLIC Daemon (installed with the server program) has to be able to communicate with the SV Router. The path this communication takes is called the SignOn Path (see 'The SignOn Path' on page 17). It must be defined in the configuration file **`svengine.cfg`** (see 'Editing the Configuration File' on page 17).

The management server uses out-of-band or Ethernet communication. All control-related signals are transmitted through the Ethernet port using TCP/IP. If the server is also used as a data server then the control-related signals are transmitted via the Ethernet port but the data travels through the fibre channel cabling. Because of the Ethernet communication, you must assign an IP address to the SV Router and reference it in the configuration file before you can run SV SAN Builder.

The IP address can be assigned by adding the MAC address and the IP address to a RARP server, or by connecting to the SV Router serial port and using a terminal emulator program to access the User Service Utility Menu. In the User Service Utility Menu, select **`6`** (View/ Change Interface Configuration) from the main menu, select **`E`** (Ethernet), then select **`A`** (Change IP Address). See the *SV Router FC-FC 3 – Installation and User Guide* for more information.

The IP Address that you enter must be copied exactly in the configuration file.

SUN Solaris Package Installation

There are three possible installations:

- Server Package: installs the server application including the SLIC daemon, and it saves the firmware microcode to the server directory. The default location for firmware microcode is **`/svengine/sdus.`**

  This application must be installed in the server directly connected to the SV Router via the Ethernet port. Once installed this server becomes the management server.

- Client Package: installs the client applications (SV SAN Builder CLI and SV Zone Manager CLI). This application should be installed in the management server as well as in any computer which shares the same LAN as the management server. Once installed the client may have full access to the daemon.

  Note: To enable client access to the daemon running on the management server, the config file located in the management server must be altered to permit client access.

- Reference Manual Package: installs the man page. This application should be installed on the management server and on the client.

If you want to install the packages in a directory other than the default directory, **`-R`** option is available. Use the parameter **`-R`** to define the full path name of a directory to use as the root_path. All files, including package system information files, are relocated to a directory tree starting in the specified root_path.

SUN Solaris Installation Server Package

Use this command to install the daemon software.

1. Login as root.
2. Insert the Vicom SVE software module v.2.5 in the CD-ROM drive.
3. Mount the CD-ROM.
4. Type **`pkgadd -d . SUNWveser`**, and press enter. The default directory is **`/svengine`**.

   The user will be prompted to start the daemon automatically each time the server boots. If the user chooses this option then after installation the user must reboot the server to start the daemon.

   If the installation was successful, the following message is displayed; **`Installation of <SUNWveser> was successful.`**

Note: svengine.cfg must be properly configured before running the daemon. See 'The Configuration File' on page 17 for more information.

SUN Solaris Installation Client Package

Use this command to install the application software (SV SAN Builder and SV Zone Manager). This may be installed on the management server that will run the daemon. To provide remote access to the management server, you may also install the client package on a server sharing the same network as the management server.

1. Login as root.
2. Insert the Vicom SVE software module v.2.5 in the CD-ROM drive.
3. Mount the CD-ROM.
4. Type **`pkgadd -d . SUNWveclt`**, and press enter. The default directory is **`/svengine`**.

   If the installation was successful, the following message is displayed: **`Installation of <SUNWveclt> was successful.`**

SUN Solaris Installation Reference Manual Package

Use this command to install the man page software.

1. Login as root.
2. Insert the Vicom SVE software module v.2.5 in the CD-ROM drive.
3. Mount the CD-ROM.
4. Type **`pkgadd -d . SUNWveman`**, and press enter. The default directory is **`/svengine`**.

   If the installation was successful, the following message is displayed: **`Installation of <SUNWveman> was successful.`**

SUN Solaris Uninstall Server Package

Use this command to remove the server package containing the daemon software.

- Type **`pkgrm SUNWveser`**, and press enter.

  If successful, the following message is displayed: **`Removal of <SUNWveser> was successful.`**

SUN Solaris Uninstall Client Package

Use this command to remove the client package containing the application software (SV SAN Builder and SV Zone Manager).

- Type **`pkgrm SUNWveclt`**, and press enter.

  If successful, the following message is displayed: **`Removal of <SUNWveclt> was successful.`**

SUN Solaris Uninstall Reference Manual Package

Use this command to remove the reference manual package.

- Type **`pkgrm SUNWveman`**, and press enter.

  If successful, the following message is displayed: **`Removal of <SUNWveman> was successful.`**

SUN Solaris Server Package Information

Use this command to determine if the package is installed or to display package details.

- Type **`pkgparam -l SUNWveser`**, and press enter.

  If successful, a message similar to the following is displayed:

```
CLASSES='none'
BASEDIR='/'
PKG='SUNWveser'
NAME='Vicom SVE Software Module -- Server Package'
DESC='Vicom Server, SVE module'
PRODNAME='Virtualization Engine'
PRODVERS='2.5'
VERSION='2.5, REV=2001.11.01.118'
ARCH='sparc'
CATEGORY='application'
VENDOR='Sun Microsystems, Inc.'
HOTLINE='Please contact your local service provider'
EMAIL=''
MAXINST='1000'
PSTAMP='sagem01122034'
PKGINST='SUNWveser'
INSTDATE='Nov 01 2001 18:06'
```

SUN Solaris Client Package Information

Use this command to determine if the package is installed or to display package details.

- Type **pkgparam -l SUNWveclt**, and press enter.

```
CLASSES='none'
BASEDIR='/'
PKG='SUNWveclt'
NAME='Vicom SVE Software Module -- Client Package'
DESC='Vicom Client, SVE module'
PRODNAME='Virtualization Engine'
PRODVERS='2.5'
VERSION='2.5, REV=2001.11.01.118'
ARCH='sparc'
CATEGORY='application'
VENDOR='Sun Microsystems, Inc.'
HOTLINE='Please contact your local service provider'
EMAIL=''
MAXINST='1000'
PSTAMP='sagem01122055'
PKGINST='SUNWveclt'
INSTDATE='Nov 01 2001 18:09'_)F
```

Sun Solaris Reference Manual Package Information

Use this command to determine if the package is installed or to display package details.

- Type **pkgparam -l SUNWveman**, and press enter.

```
CLASSES='none'
BASEDIR='/'
PKG='SUNWveman'
NAME='Vicom SVE Software Module -- Reference Manual Package'
DESC='Vicom Reference Manual, SVE module'
PRODNAME='Virtualization Engine'
PRODVERS='2.5'
VERSION='2.5, REV=2001.11.01.118'
ARCH='sparc'
CATEGORY='application'
VENDOR='Sun Microsystems, Inc.'
HOTLINE='Please contact your local service provider'
EMAIL=''
MAXINST='1000'
PSTAMP='sagem01122317'
PKGINST='SUNWveman'
INSTDATE='Nov 01 2001 18:09'
```

The Configuration File

The configuration (or config) file defines the function of the SLIC Daemon, including how it communicates and what security features are allowed.

You must edit the SignOn Path Specification before you can use SV SAN Builder. In addition, you can add sections to enable the Call Home™ Feature, Global Security Features, Failover Daemon support, and SAN Specific Security.

The SignOn Path

The SignOn Path is the communication path between the client, the daemon, and the SV Router. The SignOn Path points to the route that the communication follows (the IP address of the SV Router), and it must be defined in the configuration file.

Editing the Configuration File

When SV SAN Builder is installed, a sample file, **`svengine.cfg`**, is automatically installed in the default directory **`/svengine/sdus`**. It is a text file and can be edited with any text editor.

When you first open the configuration file, all lines are commented out:

```
# Syntax:
# SignOn_Path_Name = {
# internet_path = router_ip_address;
# };
```

Find the section you are going to be editing, remove the comment marks (**`#`**), and replace the samples with the correct information.

SignOn Path Specification

The SignOn Path specification section of the configuration file is required before you can start SV SAN Builder. It defines the SignOn Path, or how the router and the daemon will communicate.

The sample structure provided tells the SLIC Daemon to use the SignOn Path **`r0`**, through the **`internet_path`** (the IP Address) to communicate with the SV Router.

```
# r0 = {
# internet_path = 123.123.123.1;
# };
```

Advanced Ethernet Setup

The SLIC Daemon acts like a client to the SV Router and acts like a server to the client applications. By default, the SLIC Daemon communicates with the SV Router at port 25000. If the defaults are changed, this must be reflected in the configuration file, or communication will not be possible.

If the server port number has been changed in the SV Router (default=25000), the following line must be included in the SignOn Path specification section of the configuration file:

```
router_ip_port_number = <port number>;
```

Example:

```
r0 = {
internet_path = 123.123.123.123;
router_ip_port_number = <port number>;
};
```

Additional Elements of the Configuration File

The configuration file also can be edited to enable the Call Home™ Feature, Global Security features, the Failover Daemon, and SAN Specific Security. These configuration changes are listed in the system section.

```
# system = {
# <optional configuration definitions ...;>
# };
```

Note: There can be only one system entry defined in the configuration file.

Call Home™ Feature

The Call Home™ feature allows you to be notified via email when certain (user-designated) Service Request Numbers (SRNs) occur. This allows you to receive an alert automatically when certain changes or problems occur within the storage system. See Appendix A ‘SRN Reference Table’ on page 85 for a list of SRNs.

The system section of a configuration file edited to include the Call Home feature would look like the following:

```
# system = {
# email = EMAIL_ADDRESS_1, EMAIL_ADDRESS_2...;
# profile = "email program name";
# srn   = Service Routine Numbers...;
# email_header_file = FILENAME_OF_EMAIL_HEADER;
# };
```

Definitions:

email	The email address(es) where the SRN notifications are to be sent.
profile	The email program (Windows NT/2000 only).
srn	The specific service request numbers (SRNs) that are to trigger a message. More than one SRN can be applied.

Note: When entering the SRN an x can be used as a wild card. The x represents all numbers from the point the x is applied to the point that the five digit SRN ends. The higher the wildcard, the more email you will receive.

Nx:	Receives all SRNs beginning with the letter N.
N7x:	Receives all SRNs beginning with the letter N followed by number 7.

email_header_file	The path/name of the file to be included in each SRN message received. The file should be located in the same directory as svengine.cfg.

Global Security Features

There are two security features that can be applied globally across the SAN. To enable these features, include them in the configuration file system structure. If this already has been defined (for example, in the Call Home feature), append them to the existing list.

1. The IP Management feature prevents unauthorized users from accessing the daemon.

```
system = {
RemoteClientAllowed = yes/no;
AnyRemoteClient = yes/no;
HostListFileName = <path>;
AuthorizedHosts = IP1, IP2;
};
```

Definitions:

RemoteClientAllowed	Enables or disables access by remote client to the daemon.
AnyRemoteClient	Provided RemoteClientAllowed is enabled, allows any remote client to connect to the daemon.
HostListFileName	Provided RemoteClientAllowed is enabled and AnyRemoteClient is disabled, allows only hosts listed in this text file (list path) to connect.

Note: Although the text file is merely a list of IP addresses (one per line), once the text file is created, it can be modified only with the CLI **`clientip`** command (see 'Client IP Address Control [clientip]' on page 38).

Once this is enabled, removing all hosts will prevent ***any*** hosts from accessing the daemon. Use the CLI **`setmode`** command to enable remote access (see 'Setting Remote Access [setmode]' on page 37).

AuthorizedHosts	Provided RemoteClientAllowed is enabled and AnyRemoteClient is disabled, allows only IP Addresses listed here (ipaddress1, ipaddress2) to connect.

Note: Use either HostFileListName or AuthorizedHosts. Do not use both.

2. Using a password file prevents an unauthorized daemon from accessing the SV Router. The password file is a text file that lists the SignOn path and the password name in the following format:

```
c0                  abcxyz
c1                  xyzabc
```

Edit the configuration file to include the password file:

```
system = {
password_file = password_file_name;
};
```

Definition:

password_file	Path to a simple text file that contains the password.

Note: This same password also must be included in the SV Router settings. Using the SV Router User Service Utility Menu, select 4 (Router Management Program Configuration Menu), then select A (Assign Password). See the *SV Router FC-FC 3 – Installation and User Guide* for more information.

Failover Daemon

The Failover Daemon allows a backup daemon and SV Router to take over when the master daemon stops communicating with the primary SV Router. A connection is established between the two daemons, and the secondary daemon periodically pings the first daemon to ensure that it is responding. If this communication heartbeat fails, then the secondary daemon becomes the master daemon, and starts communicating through the second SV Router. If the secondary daemon fails, then the backup daemon will become the master daemon and start communicating through the third SV Router. If the last daemon in the line fails, it will continue attempting to establish communication with that same SV Router.

Note: There must be at least two different routers to establish Failover.
Although the same host can be used, ideally, there should be two hosts as well.

Although the failover ability is built into each router, in order to use this feature, a new section listing must be included in the configuration file. To configure additional SANs (the limit is 4), create a new section with a new name and add it under the system section of the configuration file. The name of the section must match the name given to the SAN.

The new section of the configuration file should look like this:

```
SAN_name = {
name = SAN_name;
PrimaryDaemon = Primary_Daemon_Name IP;
SecondaryDaemon = Secondary_Daemon_Name IP;
BackupDaemon = Backup_Daemon_Name IP;
};
```

Definitions:

name	The name of the SAN. This must match the name of the section.
PrimaryDaemon	The IP address and name of the primary daemon to be monitored (for example: 100.21.5.4, c0).
SecondaryDaemon	The IP address and name of the daemon that will take over if the primary daemon goes down.
BackupDaemon	The IP address and name of a backup daemon to take over if the secondary daemon goes down.

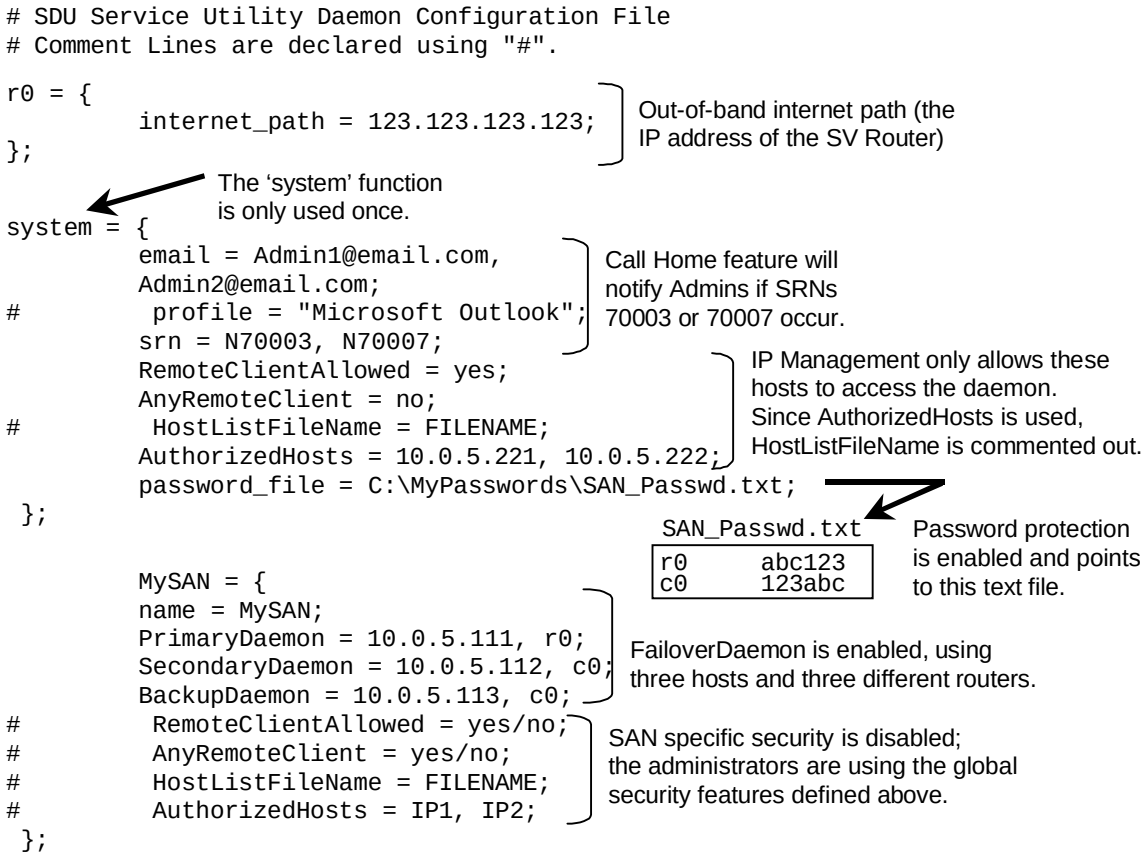
SAN Specific Security

This can be used only with the Failover Daemon. Once the new section of a configuration file has been established, you can add SAN specific security to the SANs defined. These features are the same as the IP Management features, but any remote access accessibility changes made at this point will override what is listed in the system section.

This should be appended to the Failover Daemon section of the configuration file:

```
RemoteClientAllowed = yes/no;
AnyRemoteClient = yes/no;
HostListFileName = <path>;
AuthorizedHosts = IP1, IP2;
```

See ‘Global Security Features’ for the definitions.




```
# SDU Service Utility Daemon Configuration File
# Comment Lines are declared using "#".

r0 = {
          internet_path = 123.123.123.123;
};

system = {
          email = Admin1@email.com,
          Admin2@email.com;
#          profile = "Microsoft Outlook";
          srn = N70003, N70007;
          RemoteClientAllowed = yes;
          AnyRemoteClient = no;
#          HostListFileName = FILENAME;
          AuthorizedHosts = 10.0.5.221, 10.0.5.222;
          password_file = C:\MyPasswords\SAN_Passwd.txt;
 };

          MySAN = {
          name = MySAN;
          PrimaryDaemon = 10.0.5.111, r0;
          SecondaryDaemon = 10.0.5.112, c0;
          BackupDaemon = 10.0.5.113, c0;
#          RemoteClientAllowed = yes/no;
#          AnyRemoteClient = yes/no;
#          HostListFileName = FILENAME;
#          AuthorizedHosts = IP1, IP2;
 };
```

Figure 2-1 SAN Configuration File Example

SLIC Daemon Setup

Once the SV SAN Builder program is installed, it is the SLIC Daemon that communicates between the client and the subsystem (SV Routers and drives). The SLIC Daemon runs in the background on the local or the remote computer and waits to perform a task when required. Only one active (Master) daemon is allowed per SAN.

The SLIC Daemon periodically will poll the SLIC for all subsystem errors and for topology changes. The error information is written to the file C0_SLICERR.LOG [where C0 is the SignOn Path name]. The SLIC Daemon also performs the Call Home™ function. For more information, see 'Call Home™ Feature' on page 19.

Note: Errors in the storage subsystem are logged only when the SLIC Daemon is running.

The SLIC Daemon must be running before you can use any command lines and should be started at the end of the initial boot. To start the SLIC daemon, follow the instructions listed in Starting and Stopping the Daemon.

Starting and Stopping the Daemon

Use the following commands to start and stop the daemon. Never use the **`kill`** command to stop the daemon.

Starting the Daemon

1. Log in as root, and open a terminal.
2. Change to the directory that contains the sdus directory (default is **`/svengine`**).

   Example: **`# cd /svengine/`**
3. Change to the sdus directory.

   Example: **`# cd /svengine/sdus/`**

4. Type **`ps –ef | grep slicd`** to ensure that there is no other SLIC Daemon program running on the same system.

   Example with no SLIC Daemon program running:

   ```
   root 1802 213  1  21:36:22 console  0:00   grep slicd
   ```

   Example with SLIC Daemon program running:

   ```
   root 1802 1    0  21:37:01    ?     0:00  ./slicd
   root 1820 213  1  21:38:44 console  0:00  ./slicd
   root 1807 1805 0  21:37:03    ?     0:00  ./slicd
   root 1806 1805 0  21:37:02    ?     0:00  ./slicd
   root 1812 1805 0  21:37:13    ?     0:00  ./slicd
   ```

5. Type **`./slicd`** to start the daemon.

Stopping the Daemon

1. Log in as root, and open a terminal.
2. Change to the svengine/sdus directory **.**
3. Type **`./sdushutdown`** to stop the daemon.

   ```
   # cd /svengine/sdus
   ./sdushutdown
   ```

Killing the Daemon

You should never kill daemon processes, but should use the **sdushutdown** command.

If you accidently kill a daemon process, it may not shut down correctly. Use this procedure to make sure everything related to the daemon is removed.

1. type **ps -ef | grep slicd** to see what daemon processes are running.

```
$ ps -ef | grep slicd
    root 11737     1  0 16:37:23 ?        0:00 ./slicd
    root 11738 11737  0 16:37:23 ?        0:02 ./slicd
    root 11739 11737  0 16:37:23 ?        0:00 ./slicd
    root 11745 11737  0 16:37:33 ?        0:01 ./slicd
    root 11740 11737  0 16:37:33 ?        0:00 ./slicd
```

2. type **kill** followed by the ids to kill the daemon processes.

```
$ kill 11737 11738 11739 11745 11740
```

3. Type **ipcs** to get the shared memory and semaphore ids.

```
# ipcs
IPC status from  as of Wed Sep 12 14:27:43 PDT 2001
Message Queue facility inactive.
T         ID      KEY        MODE        OWNER    GROUP
Shared Memory:
m          0   0x500000c4 --rw-r--r--     root     root
m       1001   0x5555aa8a --rw-------     root    other
m        202   0x5555aaaa --rw-------     root    other
m        203   0x5555aaba --rw-------     root    other
m        204   0x5555aabb --rw-------     root    other
Semaphores:
s     131072   0x5555aa9a --ra-------     root    other
s     131073   0x5555aa7a --ra-------     root    other
s     131074   0x5555aaba --ra-------     root    other
s     131075   0x5555aabb --ra-------     root    other
```

4. Find the KEYS that start with 0x5555aa** (* = wildcard). These are associated with the Daemon. Type **ipcrm -m (memory ID) -s (semaphore ID)** followed by the ID# to kill these memory and semaphores.

```
# ipcrm -m 1001 -m 202 -m 203 -m 204 -s 131072 -s 131073 -s 131074
-s 131075
```

CHAPTER 3

SV SAN BUILDER COMMAND LINE INTERFACE (CLI)

This chapter explains the Command Line Interface and the commands that are available in SV SAN Builder. For a summary of the commands available, see Appendix C ‘Command Line Interface Quick Reference’ on page 97. It includes these sections:

Using the SV SAN Builder Software Command Line Interface

The following commands can be used to administer the storage subsystem and its components. They are accessed from the operating system's command prompt.

Most CLI commands are run from the sduc directory. The **clientip**, **setmode** and **setmasterdaemon** commands, however, are run from the sdus directory:

Getting Started

1. Open a terminal window.
2. Change to the directory that contains the sduc directory (default is **svengine**).

   Example: **# cd /svengine**
3. Change to the sduc (or sdus) directory.

   Example: **# cd sduc**

Common CLI Parameters

The following parameters are used commonly in the SV SAN Builder CLI.

- **-d Cx**

  Cx represents the SignOn path, as specified in the config file. The configuration file (svengine.cfg) is located in the sdus directory in the server running the daemon. If the SignOn path name is unknown, view the config file to determine it.

  In the config file, the example used to show the SignOn path configuration is r0. You may use any name desired. The name you use in the config file must also be uses in the command line.

  Example SignOn Path used in config file:

  ```
  r0 = {
  internet_path = 123.123.456.789;
  };
  ```

  In this example, the **-d Cx** parameter would be written in the CLI as **-d r0**.

- **`-h Host`**

  The host represents the name or IP address of the server running the daemon.

  The **`-h Host`** parameter would be written in the CLI as **`-h 100.1.2.32`**.

- **`-t Txxxxx [Ixxxxx/s/sa]`**

  The **`-t`** parameter represents the target number of a drive or SV Router.

`-t Txxxxx`	is used to represent the target number of a simple drive.
`-t s`	is used to represent the local SV Router.
`-t Ixxxxx`	is used to represent the Initiator/SLIC number of a particular SV Router.
`-t sa`	is used to represent all SV Routers in a SAN.

- **`-v`**

  The **`-v`** parameter is used both for verbose mode (the result of the command is printed to the screen) and for user confirmation mode (confirmation is required before the command is executed).

Viewing Device Configuration Commands

Listing Device Connections [showmap]

Use the **showmap** command to list all the physical and logical devices present in the SAN. The SignOn path is always required.

The lists of physical and logical devices are presented in tables:

1. The List of SLICs table may have one or two entries: SLICs in Initiator Mode and/or SLICs in Initiator Mode. Each of these tables displays the SLIC/Initiator Number, Alias, UID or WWN, and Type for each individual SV Router within the SAN. Offline SV Routers will be placed in the List of Offline Devices Table, and will not be displayed in this table.

   The '*' denotes the Master SV Router.

```
List of SLICs in Initiator Mode:
SLIC Number SLIC Name        SLIC UID          TYPE Version
I00001                       28000060-22000073 FCFC 08.04     *
I00002                       28000060-220000CC FCFC 08.04

List of SLICs in Target Mode:
SLIC Number SLIC Name        SLIC UID          TYPE Version
```

2. The List of Target Devices table provides you with the Target Number, UID or WWN, LUN (native), VPD, Type, and Capacity of each physical target device. Offline physical target devices will be placed in the List of Offline Devices Table, and will not be displayed in this table.

```
List of Target Devices:
Target   Target UID        LUN     VPD             TYPE Capacity
Number
T00003   50020F20-00009CC3 0001    SUN   T300       DISK 488641  MB
T00004   50020F20-00009C08 0000    SUN   T300       DISK 488641  MB
T00005   50020F20-00009C08 0001    SUN   T300       DISK 488641  MB
T00006   50020F20-00009BED 0000    SUN   T300       DISK 488641  MB
T00007   50020F20-00009BED 0001    SUN   T300       DISK 488641  MB
```

3. The List of Logical Devices table provides you with the Target Number, Name, Serial Number, and Capacity for each logical device.

```
List of Logical Devices:
Target  Complex Name   TYPE       Serial Number   Capacity
Number
T49152  VMPDR000 MULTIPATH DRV  62526964-30305A54 488641 MB
T49153  VMPDR001 MULTIPATH DRV  62526964-30305A55 488641 MB
T49154  VMPDR002 MULTIPATH DRV  62526964-30305A56 488641 MB
T49155  VMPDR003 MULTIPATH DRV  62526964-30305A57 488641 MB
```

4. The Map table lists all physical and logical devices that have been mapped and displays their SCSI/FC ID and LUN, Target Number, and UID/Complex Name. The maps listed in this table are global SAN maps and not the localized SV Router zone maps.

```
FC Map:
FC       Target   UID/Complex Name
MAP      Number
00-000   T49152   VMPDR000
00-001   T49153   VMPDR001
00-002   T49154   VMPDR002
00-003   T49155   VMPDR003
```

5. The List of Unmapped Drives table provides the Target Number and UID/Complex Name of any target devices that have not been mapped globally.

```
List of Unmapped Drives:
Target   UID/Complex Name
Number
T00000       50020F20-000093B5 0000
```

6. The List of General Spare Drives table provides the Target Number and UID of any target devices that have been allocated as general spares.

```
List of General Spare Drives:
Target   UID
Number
T00002       50020F20-000093B5 0002
```

7. The List of Offline Devices table provides the Target or SLIC/Initiator Number, UID or WWN, and Type of any physical devices that are offline.

```
List of Offline Devices:
SLIC/Target  UID                       Type
Number
T00001       50020F20-000093B5 0001
```

Usage:

showmap -d Cx {-m [lists_option] -f File_Name -h Host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-m [lists_option]	Optional. Selective display of tables. If this parameter is not specified, the program will display all tables relevant to the physical and logical devices in the SAN.
all	Output map showing all tables.
target	Output map of all target devices.
fc	Output global FC map table.
slic	Output List of SV Routers table.
physical	Output List of all physical device tables.
spare	Output List of General Spare Drives table.
offline	Output List of Offline Devices table.
unmapped	Output List of Unmapped Drives table.
-h Host	Optional. The name or IP address of the host running the daemon.
-f File_Name	Optional. Print maps to the file **File_Name**.
-v	Optional. Print maps to console (default).

Examples:

The following example will display all tables. The **showmap** command is executed on a server running the daemon (r0).

```
showmap -d r0
```

The following example will display the general spare drives. The **showmap** command is executed on a server running the daemon (r0).

```
showmap -d r0 -m spare
```

The following example will display all tables. The **showmap** command is executed remotely from the daemon (r0) running on server (myhost).

```
showmap -d r0 -h myhost
```

SLIC Daemon Commands

Listing SLIC Daemons [sgetname]

Use the **sgetname** command to list the SignOn paths for the daemon processes running on each host.

Usage:

```
sgetname -h Host
```

-h Host The name or IP address of the host running the daemon.

Examples:

The following example will display the SignOn paths for the SLIC Daemon processes running on host 205.119.173.113.

```
sgetname -h 205.119.173.113
```

Setting the Master Daemon [setmasterdaemon]

The **SetMasterDaemon** command is used to set the master daemon in the SAN. After failover, the primary daemon no longer acts as the master daemon. Run this command to reset the primary daemon back to the master daemon. This command can be executed only from the sdus directory in the daemon server.

Note: **setmasterdaemon** is only used in a failover daemon environment.

If there are any users connected to the current daemon, resetting back to the master daemon will cause them to be disconnected. The system will display the following message:

```
# users connected to daemon. Proceed? (Y or N):.
```

Type **Y** to disconnect the other users and continue setting the master daemon; type **N** to abort the command.

Usage:

setmasterdaemon -d Cx {-h Host -f}

-d Cx	Cx is the SignOn path, as specified in the config file.
-h Host	Optional. The name or IP address of the host running the daemon.
-f	Optional. Bypass user confirmation step.

Examples:

In the following example, the command **setmasterdaemon** is executed remotely. It will force the primary daemon (c0) running on server (myhost) to become the master daemon.

setmasterdaemon -d c0 -h myhost

Listing SAN Communication Properties [signoninfo]

The **signoninfo** command displays how communication between the daemon and the SV Router has been established. It lists the SignOn SLIC (SV Router) UID and the IP address of the router.

Usage:

signoninfo –d Cx {–h Host}

-d Cx	Cx is the SignOn path, as specified in the config file.
-h Host	Optional. The name or IP address of the host running the daemon.

Example

The following example will display the SignOn SLIC (SV Router) UID and the IP address of the router connected to the server running the daemon (c0).

signoninfo -d c0

Setting Remote Access [setmode]

The **setmode** command enables or disables remote access to the daemon server. This command can be executed only from the sdus directory in the daemon server. When setmode is disabled, only local access to the storage subsystem is allowed through the daemon server. If a remote host application is connected to the storage subsystem when setmode (remote access) is disabled, the remote host application will continue to have access. However, once the remote host application is disconnected, it will not be able to gain access again.

Enable allows any remote client to access the daemon server; disable allows no remote clients. Once specific hosts are enabled, only authorized hosts can access the daemon server. If all authorized hosts are removed, no host will be able to access the server until setmode is enabled.

The default setting is remote access enabled.

If the **setmode** command is executed without specifying any options (-o), the command displays the current remote access mode (whether it is enabled or disabled) of the daemon server.

Usage:

```
setmode {-o [enable/disable]}
```

-o [enable/disable] Optional. Enables or Disables remote access.

Example

The following example will enable remote access to the daemon server.

```
setmode -o enable
```

Client IP Address Control [clientip]

The **clientip** command allows you to add, delete, and view the IP addresses that are allowed to access the SLIC Daemon. This command can only be executed from the sdus directory in the daemon server. The IP addresses are listed in a text file that is referenced by the svengine configuration file.

Adding an IP Address

Use the **clientip add** command to add client host IP address(es) to be enabled for remote access to the daemon.

Usage:

clientip add –h IP_Address {-v}

-h IP_Address	Client host IP address. If more than one, separate with spaces (for example **255.118.9.16 255.118.9.17**).
-v	Optional. User confirmation required.

Example

The following example will add two client hosts (123.1.1.111 and 123.1.1.222) to the list of hosts enabled for remote access to the daemon.

```
clientip add -h 123.1.1.111 123.1.1.222
```

Deleting an IP Address

Use the **`clientip del`** command to delete client host IP address(es) from the list of hosts enabled to access the daemon.

Usage:

`clientip del –h IP_Address {-v}`

`-h IP_Address`	Client host IP address. If more than one, separate with spaces (for example **`255.118.9.16 255.118.9.17`**).
`-v`	Optional. User confirmation required.

Example

The following example will delete two client hosts (255.118.9.16 and 255.118.9.17) from the list of hosts enabled to access the daemon.

```
clientip del -h 255.118.9.16 255.118.9.17
```

Viewing the IP Address List

Use the **`clientip view`** command to view the list of client host IP addresses that are enabled for remote access to the daemon. No parameters required.

Usage:

```
clientip view
```

Examples:

The following example will display the list of client hosts that are enabled for remote access to the daemon.

```
clientip view
```

Connection List [connlist]

Use the **`connlist`** command to list the IP addresses of the hosts currently connected to the SLIC Daemon and the applications they are connected with. If the SignOn Path is not specified, it will display this information for all SANs the daemon is controlling.

Usage:

```
connlist {-d Cx -h host}
```

`-d Cx`	Optional. Cx is the SignOn path, as specified in the config file.
`-h host`	Optional. The name or IP address of the host running the daemon.

Example:

The following example will list the IP addresses of the hosts currently connected to the daemon (c0) running on host 123.123.123.224, and what applications these hosts are connected with.

```
connlist -d c0 -h 123.123.123.224
```

Example return from connlist command:

```
IP client Host          client app name         date
100.100.100.1           vlun                    09:00:30AM
```

Listing SLIC Daemon Configuration Information [saninfo]

The **saninfo** command displays the SLIC Daemon configuration information. If the SignOn Path is not specified, it will display the configuration information for all SANs the daemon is controlling.

Usage:

```
saninfo {-d Cx -h host}
```

-d Cx	Optional. Cx is the SignOn path, as specified in the config file.
-h host	Optional. The name or IP address of the host running the daemon.

Examples:

The following example will display the configuration for daemon (c0) running on host 123.123.123.224.

```
saninfo -d c0 -h 123.123.123.224
```

Simple Drive Commands [drvprop]

The **drvprop** command allows you to manage the attributes of simple drives.

Changing a Simple Drive to a General Spare Drive

Use the **drvprop spare** command to convert a simple drive to a general spare drive. To create multiple general spares simultaneously, add additional target numbers (**-t Txxxxx Txxxxx Txxxxx**) to the command line.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

```
drvprop spare -d Cx -t Txxxxx {-h Host -v}
```

-d Cx	Cx is the SignOn path, as specified in the config file.
-t Txxxxx	Target number of a simple drive. If more than one, separate with spaces (for example **T00001 T00002**).
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will convert the simple drives T00001 and T00009 to general spare drives. The **drvprop** command is executed on a server running the daemon (r0).

```
drvprop spare -d r0 -t t1 t9
```

Changing Properties of a Simple Drive

Use the **drvprop change** command to change the global map of a simple drive.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

drvprop change -d Cx -t Txxxxx -s [TxDy/Lxxx] {-h Host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-t Txxxxx	Target number of the simple drive.
-s [TxDy/Lxxx]	
TxDy	For SCSI host, assigns SCSI ID (x) and LUN (y).
Lxxx	For FC host, assigns Fibre Channel LUN (xxx).
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will change the map of T00000 from LUN 000 to LUN 016. The **drvprop** command is executed on a server running the daemon (r0).

```
drvprop change -d r0 -t t0 -s L16
```

The following example will change the attributes of general spare drive T00011 to LUN 011. The **drvprop** command is executed on a server running the daemon (r0).

```
drvprop change -d r0 -t t11 -s L11
```

MultiPath Drive Commands [mpdrive]

The **mpdrive** command is used to configure and manage MultiPath drives. MultiPath drive functionality is supported only in conjunction with the Sun StorEdge™ T3 Array.

Autocreating a MultiPath Drive

Before you can carve the T3 LUNs into virtual drives, you must first convert the LUNs presented by the T3 disk array into multipath drives, and then add the multipath drives to the disk pool.

Use the **mpdrive autocreate** command to check if a MultiPath drive can be created, read the active and passive paths, and create the MultiPath drive. It assigns a name and a global map to the new drive and displays the target number.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

mpdrive autocreate -d Cx {-h Host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will automatically create a set of MultiPath drives. The **mpdrive** command is executed on a server running the daemon (r0).

mpdrive autocreate -d r0

Removing a MultiPath Drive

Use the **`mpdrive remove`** command to remove a MultiPath drive. The members of the removed MultiPath drive will become general spare drives.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

`mpdrive remove -d Cx -m Txxxxx {-h host -v}`

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-m Txxxxx`	The target number of the MultiPath drive to be removed.
`-h Host`	Optional. The name or IP address of the host running the daemon.
`-v`	Optional. User confirmation required.

Examples:

The following example will remove a MultiPath drive (T49153). The **`mpdrive`** command is executed on a server running the daemon (r0).

```
mpdrive remove -d r0 -m t49153
```

Using MultiPath Drive Failback

Use the **`mpdrive failback`** command to switch between the active and passive paths by specifying the storage system's controller serial number. Use the **`mpdrive view`** command to obtain the controller serial number associated with the MultiPath drive.

First re-establish the primary path, and issue the command **mpdrive failback.** Next, send I/O to one of the LUNs via its primary path. Failback will occur once I/O transmission begins.

Usage:

`mpdrive failback -d Cx -j UID {-h host -v}`

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-j UID`	The controller serial number of the MultiPath drive. This must be entered in upper case. See 'Viewing MultiPath Drive Properties' on page 49 to find the controller serial number.
`-h Host`	Optional. The name or IP address of the host running the daemon.
`-v`	Optional. User confirmation required.

Examples:

The following example will perform a failback on the set of MultiPath drives (after a prior failover incident has occurred on the array with controller serial number 20000001000001FE). The mpdrive command is executed from a remote host to the daemon (r0) running on server (100.2.34.120).

`mpdrive failback -d r0 -j 20000001000001FE -h 100.2.34.120`

Replacing a MultiPath Drive

Use the **mpdrive replace** command to update the MultiPath drive's properties when a T3 controller is replaced. In this case, SAN Builder scans the T3 for changes and updates the SAN Database with the new information.

Usage:

mpdrive replace -d Cx -f {-h host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-h Host	Optional. The name or IP address of the host running the daemon.
-f	Force replace.
-v	Optional. User confirmation required.

Examples:

The following example will update the MultiPath drives' properties after a T3 controller has been replaced. The **mpdrive** command is executed on a server running the daemon (r0).

mpdrive replace -d r0 -f

Changing MultiPath Drives

Use the **`mpdrive change`** command to change the map of the MultiPath drive.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

`mpdrive change -d Cx -m Txxxxx -s map {-h host -v}`

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-m Txxxxx`	The target number of MultiPath drive.
`-s map`	The new global map of the MultiPath drive.
`-h Host`	Optional. The name or IP address of the host running the daemon.
`-v`	Optional. User confirmation required.

Examples:

The following example will change the map of a MultiPath drive (T49154) to LUN 008. The **`mpdrive`** command is executed on a server running the daemon (r0).

```
mpdrive change -d r0 -m t49154 -s L8
```

Viewing MultiPath Drive Properties

Use the **`mpdrive view`** command to display the name, target number, drive capacity, active and passive paths, and controller serial number of a MultiPath Drive.

Usage:

`mpdrive view -d Cx {-m Txxxxx -h Host}`

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-m Txxxxx`	Optional. The target Number of MultiPath drive.
`-h Host`	Optional. The name or IP address of the host running the daemon.

Examples:

The following example will display the MultiPath drive (T49154) properties. The **`mpdrive`** command is executed on a server running the daemon (r0).

```
mpdrive view -d r0 -m t49154
```

Disk Pool Commands [vdiskpool]

The **vdiskpool** command is used to configure and manage disk pools. Before drives can be carved, they must be added to a disk pool in a SAN. Simple drives, general spare drives, and MultiPath drives can be used as pool drives in the disk pool for LUN carving.

Note: Before you can carve the T3 LUNs into virtual drives, you must first convert the LUNs presented by the T3 disk array into multipath drives, and then add the multipath drives to the disk pool.

Creating a Disk Pool

Use the **vdiskpool create** command to create disk pools. Disk pools can be assigned names for easy management. Disk pool names must be unique in a SAN. Optionally, simple drives, general spare drives and MultiPath drives can be added to the disk pool during the creation process.

Usage:

vdiskpool create -d Cx -n PoolName {-t [Txxxxx/all] -h host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-n PoolName	The name of the disk pool.
-t [txxxxx/all]	Optional. Add a particular drive or drives by target number, or add all simple, spare, and MultiPath drives.
txxxxx	Add a particular drive or drives by target number.
all	Add all simple, spare, and MultiPath drives.
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will create an empty disk pool named MYPOOL. The **vdiskpool** command is executed on a server running the daemon (r0).

```
vdiskpool create -d r0 -n MYPOOL
```

The following example will create a disk pool named MYPOOL with two pool drives T49152 and T49153. The **vdiskpool** command is executed on a server running the daemon (r0).

```
vdiskpool create -d r0 -t t49152 t49153 -n MYPOOL
```

Adding Pool Drives to a Disk Pool

Use the **vdiskpool add** command to add one or more drives to an existing disk pool. If there is more than one disk pool in a SAN, a disk pool name can be specified to assign the drives to that disk pool.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

vdiskpool add -d Cx -t [Txxxxx/all] {-p PoolName -h Host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-t [txxxxx/all]	Optional. Add a particular drive or drives by target number, or add all simple, spare, and MultiPath drives.
txxxxx	Add a particular drive or drives by target number.
all	Add all simple, spare, and MultiPath drives.
-p PoolName	Optional. The name of the disk pool.
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will add a pool drive T49154 to a disk pool named MYPOOL. The **vdiskpool** command is executed on a server running the daemon (r0).

```
vdiskpool add -d r0 -t t49154 -p MYPOOL
```

Deleting Pool Drives from a Disk Pool

Use the **`vdiskpool del`** command to delete one or more pool drives from the disk pool.

Remove all virtual drives from a pool drive before running this command. If any of the pool drives contain virtual drives, the command will be aborted.

If the deleted pool drives were physical drives, they revert to general spare drives. If they were MultiPath drives, they revert to unmapped drives.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

`vdiskpool del -d Cx -t [Txxxxx/all] {-p PoolName -h host -v}`

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-t [txxxxx/all]`	Optional. Delete a particular drive or drives by target number, or delete all pool drives from the disk pool.
`txxxxx`	Add a particular drive or drives by target number.
`all`	Delete all pool drives.
`-p PoolName`	Optional. The name of the disk pool.
`-h Host`	Optional. The name or IP address of the host running the daemon.
`-v`	Optional. User confirmation required.

Examples:

The following example will delete pool drives T49152 and T49153 from a disk pool named MYPOOL. The vdiskpool command is executed on a server running the daemon (r0).

`vdiskpool del -d r0 -t t49152 t49153 -p MYPOOL`

Removing a Disk Pool

Use the **vdiskpool remove** command to remove the disk pool.

To remove the disk pool, you must first remove all pool drives in it. If the disk pool contains a pool drive, the command will be aborted.

Note: To remove a pool drive, you must remove all the virtual drives in the pool drive.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

vdiskpool remove -d Cx {-p PoolName -h host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-p PoolName	Optional. The name of the disk pool.
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will remove a disk pool named MYPOOL. The **vdiskpool** command is executed on a server running the daemon (r0).

```
vdiskpool remove -d r0 -p MYPOOL
```

Renaming a Disk Pool

Use the **vdiskpool change** command to modify the name of a disk pool.

Usage:

vdiskpool change -d Cx -p PoolName -n NewPoolName {-h host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-p PoolName	The current name of the disk pool.
-n NewPoolName	The new name of the disk pool.
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will change the name of the disk pool MYPOOL to MyNewPool. The **vdiskpool** command is executed on a server running the daemon (r0).

```
vdiskpool change -d r0 -p MYPOOL -n MyNewPool
```

Viewing a Disk Pool

Use the **vdiskpool view** command to show all pool drives that belong to a particular disk pool. If no disk pool is specified, all pool drives in all disk pools will be shown.

The physical table of a disk pool lists all the pool drives and their properties (WWN/Complex Name and capacity) in the disk pool. The logical table of a disk pool lists all the virtual drives and their properties in the disk pool.

Usage:

vdiskpool view -d Cx {-p PoolName -m [physical/logical/all] -h host}

-d Cx	Cx is the SignOn path, as specified in the config file.
-p PoolName	Optional. The name of the disk pool.
-m [physical/logical/all]	Optional. Allows selective view of the disk pool.
physical	List pool drives and their properties.
logical	List virtual drives and their properties.
all	View both physical and logical lists.
-h Host	Optional. The name or IP address of the host running the daemon.

Examples:

The following example will display the disk pool(s) properties. The **vdiskpool** command is executed on a server running the daemon (r0).

```
vdiskpool view -d r0
```

Viewing Disk Pool Statistics

Use the **vdiskpool stat** command to show the statistics for each disk pool, including the starting and stopping logical block addresses (LBA) for all free and reserved spaces.

Usage:

vdiskpool stat -d Cx {-p PoolName -h host}

-d Cx	Cx is the SignOn path, as specified in the config file.
-p PoolName	Optional. The name of the disk pool.
-h Host	Optional. The name or IP address of the host running the daemon.

Examples:

The following example will display the statistics of the disk pool(s). The **vdiskpool** command is executed on a server running the daemon (r0).

vdiskpool stat -d r0

Virtual Drive Commands [vlun]

The **`vlun`** command is used to configure and manage virtual drives. Virtual drives can only be carved from pool drives in a disk pool.

Creating a Virtual Drive (Virtual LUN)

Use the **`vlun create`** command to create a virtual drive.

Note: Before you can carve the T3 LUNs into virtual drives, you must first convert the LUNs presented by the T3 disk array into multipath drives, and then add the multipath drives to the disk pool.

If the disk pool is specified, but no pool drive target number is specified, the virtual drive will be created from the first available free space (that meets the size requirement) in that disk pool.

If the pool drive target number is specified, the virtual drive will be created from that pool drive. The disk pool, if specified, will be ignored.

If neither the disk pool nor a pool drive is specified, the virtual drive will be created from the first available free space that meets the size requirement.

Note: Whenever you create new virtual drives, the data server will show an error message reading: **`corrupt label - wrong magic number.`** To correct the corrupt label message, use format (1M) to label the newly created virtual LUN.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

vlun create -d Cx -l size {-n VdriveName -p PoolName -s map -t Txxxxx -h host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-l size	The size of new virtual drive in GB.
-n VdriveName	Optional. The name of the virtual drive.
-p PoolName	Optional. The name of the disk pool.
-s map	Optional. The map of the virtual drive.
-t Txxxxx	Optional. The target number of the pool drive to be used.
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will create a virtual drive of size 10GB from any place in the diskpools. The **vlun** command is executed on a server running the daemon (r0).

vlun create -d r0 -l 10

The following example will create a virtual drive of size 10GB, with the name MyVDrive and a global map of LUN 023 from pool drive (T00001). The **vlun** command is executed on a server running the daemon (r0).

vlun create -d r0 -t t1 -n MyVDrive -s L23

The following example will create a virtual drive of size 10GB, with name MyVDrive and a global map of LUN 023, from any free space in the disk pool (MYPOOL). The **vlun** command is executed on a server running the daemon (r0).

vlun create -d r0 -p MYPOOL -n MyVDrive -s L23

Autocreating a Virtual Drive (Virtual Lun)

Use the **`vlun autocreate`** command to carve several virtual drives in one instance. If a disk pool is specified, that disk pool will be used. If no disk pool is specified, the virtual drives will be autocreated based on free space. Pool drive target numbers will be ignored.

Note: Before you can carve the T3 LUNs into virtual drives, you must first convert the LUNs presented by the T3 disk array into multipath drives, and then add the multipath drives to the disk pool.

All drives must be the same size. If a drive name is specified, all drives created will share the same name.

Note: Whenever you create new virtual drives, the data server will show an error message reading: **`corrupt label - wrong magic number.`** To correct the corrupt label message, use format (1M) to label the newly created virtual LUN.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

```
vlun autocreate -d Cx -c count -l size {-n MyVDrive -p PoolName -h
host -v}
```

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-c count`	The number of virtual drives to be created.
`-l size`	The size of each virtual drive to be created in GB.
`-n VdriveName`	Optional. The name of the virtual drive.
`-p PoolName`	Optional. The name of the disk pool.
`-h Host`	Optional. The name or IP address of the host running the daemon.
`-v`	Optional. User confirmation required.

Examples:

The following example will automatically create 5 virtual drives of size 10GB each, with a common name of MyVDrive. The **vlun** command is executed on a server running the daemon (r0).

```
vlun autocreate -d r0 -l 10 -c 5 -n MyVDrive
```

The following example will automatically create 5 virtual drives of size 10GB each, with a common name of MyVDrive in any free space in disk pool (MYPOOL). The **vlun** command is executed on a server running the daemon (r0).

```
vlun autocreate -d r0 -c 5 -l 10 -n MyVDrive -p MYPOOL
```

Removing a Virtual Drive (Virtual Lun)

Use the **vlun remove** command to remove one or more virtual drives (specified by target number) from the disk pool.

The removed virtual drives revert to free space in the disk pool.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

```
vlun remove -d Cx -t Txxxxx {-h host -v}
```

-d Cx	Cx is the SignOn path, as specified in the config file.
-t Txxxxx	The target number of the virtual drive(s) to be removed.
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will remove a virtual drive (T16386). The **vlun** command is executed on a server running the daemon (r0).

```
vlun remove -d r0 -t t16386
```

Changing a Virtual Drive (Virtual Lun)

Use the **`vlun change`** command to modify the name and/or global map of a virtual drive.

Caution ! ***Please wait until all I/O activity is completed in the selected drives before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

`vlun change -d Cx -t Txxxxx -n VdriveName -s map {-h host -v}`

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-t Txxxxx`	The target number of the virtual drive.
`-n VdriveName`	The new name of the virtual drive.
`-s map`	The new global map of the virtual drive.
`-h Host`	Optional. The name or IP address of the host running the daemon.
`-v`	Optional. User confirmation required.

Examples:

The following example will change the name of virtual drive (T16387) to VDRIVE_NEW. The **`vlun`** command is executed on a server running the daemon (r0).

`vlun change -d r0 -t t16387 -n VDRIVE_NEW`

Viewing Virtual Drive (Virtual Lun) Properties

Use the **`vlun view`** command to show the properties of a virtual drive.

If a disk pool is specified, the properties of all virtual drives in the disk pool will be displayed. The target number of any virtual drives, if specified, will be ignored.

If no disk pool is specified, and the target number(s) of a virtual drive is, only the properties of the specified virtual drive(s) will be displayed.

Usage:

`vlun view -d Cx {-p PoolName -t Txxxxx -h Host}`

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-p pool name`	Optional. The name of the disk pool.
`-t Txxxxx`	Optional. The target number of the virtual drive.
`-h Host`	Optional. The name or IP address of the host running the daemon.

Examples:

The following example will display the virtual drive (T16387) properties. The **`vlun`** command is executed on a server running the daemon (r0).

`vlun view -d r0 -t t16387`

SLIC (SV Router) Commands

Download Microcode [sdnld]

Use the sdnld command to download the microcode for the SV Router. This function can be used to download microcode to all the SV Routers in the SAN.

Caution ! ***Do not download new microcode to the SV Router FC-FC 3 if it is being used by the operating system. The SV Router will reset itself after the download is complete, which can cause lost I/Os and system panic.***

Usage:

`sdnld -d Cx -t [s/Ixxxxx/sa] -f File_Name {-h Host}`

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-t [s/Ixxxxx/sa]`	Initiator/SLIC number of the SV Router.
`s`	SignOn SV Router.
`sa`	All SV Routers in the SAN.
`Ixxxxx`	Initiator number of the SV Router.
`-f File_Name`	Name of the microcode file.
`-h Host`	Optional. The name or IP address of the host running the daemon.

Examples

The following example will download microcode (fcfc3.ima) to SV Router I00002. The **`sdnld`** command is executed on a server running the daemon (r0).

```
sdnld -d r0 -t i2 -f fcfc3.ima
```

The following example will download microcode (fcfc3.ima) to all the SV Routers in the SAN. The **`sdnld`** command is executed from a remote host to the daemon (r0) running on server (123.123.123.111).

```
sdnld -d r0 -t sa -f fcfc3.ima -h 123.123.123.111
```

SLIC Alias Operations [slicalias]

The **slicalias** command is used to create, change, clear or view an alias assigned to the SV Router. Without an alias, you must refer to the SV Router by its Initiator/SLIC number (for example, **I00001**).

Creating a SLIC Alias

Use the **slicalias create** command to create an alias to identify the SV Router.

Usage:

slicalias create –d Cx –t Ixxxxx –n New_Name {–h Host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-t Ixxxxx	Initiator/SLIC number of the SV Router.
-n New_Name	The alias to be applied to the SV Router (maximum = 16 characters).
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will create an alias (SVEHW_2) for SV Router I00002. The **slicalias** command is executed on a server running the daemon (r0).

```
slicalias create -d r0 -t I2 -n SVEHW_2
```

Changing a SLIC Alias

Use the **slicalias change** command to change the alias assigned to an SV Router.

Usage:

slicalias change –d Cx –t Ixxxxx –n New_Name {–h Host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-t Ixxxxx	Initator/SLIC number of the SV Router.
-n New_Name	The new alias that will be applied to the SV Router (maximum = 16 characters).
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will change the name of the SV Router I2 from SVEHW_2 to SVEHW_NY. The **slicalias** command is executed on a server running the daemon (r0).

```
slicalias change -d r0 -t I2 -n SVEHW_NY
```

Clearing a SLIC Alias

Use the **slicalias clear** command to clear the assigned alias for an SV Router.

Usage:

slicalias clear -d Cx -t Ixxxxx {-h Host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-t Ixxxxx	Initator/SLIC number of the SV Router.
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will clear the alias for SV Router I00002. The **slicalias** command is executed on a server running the daemon (r0).

```
slicalias clear –d r0 –t I2
```

Viewing a SLIC Alias

Use the **`slicalias view`** command to view the alias for an SV Router.

Usage:

`slicalias view –d Cx –t Ixxxxx {–h Host}`

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-t Ixxxxx`	Initator/SLIC number of the SV Router.
`-h Host`	Optional. The name or IP address of the host running the daemon.

Examples:

The following example will display the alias for SV Router I00002. The **`slicalias`** command is executed on a server running the daemon (r0).

`slicalias view –d r0 –t I2`

Force Master SLIC (SV Router) [smaster]

Use the **`smaster`** command to force an SV Router to be the master SV Router in the SAN. The master SV Router provides services including synchronizing of device configuration, and gathering and reporting of device status changes.

If the Initiator/SLIC Number of a SV Router is not specified, the SignOn Router is assumed to be the SV Router for the master SV Router assignment.

Usage:

`smaster -d Cx {-t Ixxxxx -h Host}`

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-t Ixxxxx`	Optional. Initiator/SLIC number of the SV Router.
`-h Host`	Optional. The name or IP address of the host running the daemon.

Examples:

The following example will force SV Router I00002 to be the master SV Router in the SAN. The **`smaster`** command is executed on a server running the daemon (r0).

`smaster –d r0 –t I2`

SAN Configuration File (Backup and Emergency Recovery) [sanconfig]

The **sanconfig** command allows you to save the SAN configuration file to an offline file, and then download the file to the SV Router if needed for emergency recovery.

You should save the SAN configuration periodically for effective Emergency Recovery.

If you save the SAN configuration file before you make any changes to the configuration, you will be able to revert back to the original configuration if necessary.

Reading SAN Configuration File and Saving to File

Use the **sanconfig read** command to read the SAN configuration file from the SV Routers in the SAN to an offline file. To prevent losing drive configuration information, you should make a copy of this file whenever the configuration changes.

If you do not select a -m option, the default (all three options) will be used.

Usage:

sanconfig read -d Cx -e FileName {-m SANCfgType -h host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-e FileName	The SAN configuration file name.
-m SANCfgType	Optional. SAN Configuration type. Choose one or more of the options. The default includes all three options (physical, logical, and Zone configuration information).
physical	Physical components in the SAN (i.e. SV Routers and drives).
logical	Logical drive configuration in the SAN.
zone	SAN Zone Configuration.
-h host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will save the SAN configuration to a file (SanFile.san) stored locally in the management server. The **sanconfig** command is executed on a server running the daemon (r0).

```
sanconfig read -d r0 -e SanFile.san
```

Writing SAN Configuration File to SV Router

Use the **sanconfig write** command to download the offline SAN configuration file to the SV Routers in the SAN. This will restore all of the drive configurations (simple drives, virtual drives, etc.) from the last save, as well as any zone configurations (Zones, SV Domains, etc.). The physical setup must be exactly the same as it was when the backup was taken.

If you do not select a -m option, the default (all three options) will be used.

Caution ! ***Please wait until all I/O activity is completed before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

```
sanconfig write -d Cx -e FileName {-m SANCfgType -h host -v}
```

-d Cx	Cx is the SignOn path, as specified in the config file.
-e FileName	The SAN configuration file name.
-m SANCfgType	Optional. SAN Configuration type. Choose one or more of the options. The default includes all three options (physical, logical, and Zone configuration information).
physical	Physical components in the SAN (i.e. SV Routers and drives).
logical	Logical drive configuration in the SAN.
zone	SAN Zone Configuration.
-h host	Optional. The name or IP address of the host running the daemon.
-v	Optional. User confirmation required.

Examples:

The following example will download the physical and logical components of the SAN from a SAN configuration file to the SV Routers. The **sanconfig** command is executed on a server running the daemon (r0).

```
sanconfig write -d r0 -e SanFile.san -m physical logical
```

Emergency Recovery for Multiple Router Configuration - Local Access

1. Stop I/O from data servers to SV Routers.
2. Power off both SV Routers.
3. Power on only one SV Router, clear its SAN database, and power off the router.
4. Power on the other router, clear its SAN database, and cycle power.

   At this point, you should have cleared the SAN database in both routers and one router should be powered on.
5. Start the daemon in the management station. Be sure to access the SAN using the powered on SV Router
6. Using the **sanconfig write** command, download the offline drive configuration file to the SV Router. This will restore all of the drive configurations (mirror drives, virtual drives, etc.) The physical setup must be exactly the same as it was when the backup was taken.

Example:

```
sanconfig write -d SAN1r0 -e /svengine/SANconf/T3SAN.san -m
physical logical
```

7. After the download, cycle SV Router power.
8. Ensure the green LED is solid-on in the front of the powered-on SV Router. Then power on the second SV Router. At this point, the second router will download the SAN database from the first router.
9. Use the **showmap** command to view drive configurations and ensure they are restored.

Example

```
showmap -d SAN1r0
```

10. Using the **`sanconfig write`** command, download the offline zone configuration file to one SV Router. This will restore all of the zone configurations to both routers.The physical setup must be exactly the same as it was when the backup was taken.

Example:

```
sanconfig write -d SAN1r0 -e /svengine/SANconf/T3SAN.san -m zone
```

11. Using the **`sadapter view`** command, ensure the HBA sees both SV Routers.

Example:

```
sadapter view -d SAN1r0 -r I1
sadapter view -d SAN1r0 -r I2
```

12. Recovery is complete.

13. Perform T3 failback if necessary.

14. Enable dual multipathing if necessary. This should be done on each data server.

Emergency Recovery for Multiple Router Configuration - Remote Access

1. Stop I/O from data servers to SV Routers.

2. Telnet to one SV Router, and use the *User Service Utility* menu to clear the SAN database. The SV Router will enter a suspended state, and it will flash service code 060.

3. Start the daemon in the management station. Be sure to access the SAN using the powered on SV Router

4. Using the **`sanconfig write`** command, download the offline drive configuration file to the functioning SV Router. It will overwrite the SAN database in the router, and restore all of the drive configurations (mirror drives, virtual drives, etc.) The physical SAN configuration must be exactly the same as it was when the backup was taken.

Example:

```
sanconfig write -d SAN1r0 -e /svengine/SANconf/T3SAN.san -m
physical logical
```

5. After the download is complete, use the *User Service Utility* menu in the suspended router and reboot it. When this router begins functioning, it will download the configuration file from the functioning router.

6. Use the **`showmap`** command to view drive configurations and ensure they are restored.

Example

```
showmap -d SAN1r0
```

7. Using the **`sanconfig write`** command, download the offline zone configuration file to one SV Router. This will restore all of the zone configurations to both routers.The physical setup must be exactly the same as it was when the backup was taken.

Example:

```
sanconfig write -d SAN1r0 -e /svengine/SANconf/T3SAN.san -m zone
```

8. Using the **`sadapter view`** command, ensure the HBA sees both SV Routers.

Example:

```
sadapter view -d SAN1r0 -r I1
sadapter view -d SAN1r0 -r I2
```

9. Recovery is complete.
10. If this is a T3 configuration, perform T3 failback.
11. Enable dual multipathing if necessary. This should be done on each data server.

Importing SAN Zone Configuration

Use the **`sanconfig import`** command to upload Zone information to an SV Router that has been replaced in a multi-router environment.

Note: This is only necessary when there are Zones and more than one SV Router in the configuration. The SV Routers will sync together to get the drive information, but the zone information must be imported.

To replace an SV Router with zoning, follow these steps:

1. Power off the SV Router to be replaced.
2. Replace the SV Router.
3. Power on the new SV Router. The drive information will automatically be copied from the existing Master SV Router.
4. Run the **`sanconfig import`** command to upload the Zone information.

Caution ! ***Please wait until all I/O activity is completed before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

```
sanconfig import -d Cx -e FileName -r NewSLIC# -j CurrentSLIC# {-h
Host -v}
```

`-d Cx`	Cx is the SignOn path, as specified in the config file.
`-e FileName`	The SAN configuration file name.
`-r NewSLIC#`	The new SV Router's Initiator/SLIC number.
`-j CurrentSLIC#`	The current SV Router's Initiator/SLIC number (the unit being replaced).
`-h host`	Optional. The name or IP address of the host running the daemon.
`-v`	Optional. User confirmation required.

Examples:

The following example will import the zoning information from SV Router I00002 to SV Router I00003. The sanconfig command is executed on a server running the daemon (r0).

```
sanconfig import -d r0 -e SanFile.san -r i3 -j i2
```

SAN Import [sanimport]

The **sanimport** command imports the SAN configuration file from the source SAN to the destination SAN, based on the import configuration file defined.

The import configuration file is a text file that contains the following structure:

```
[SAN Parameters]
DB2Import=[yes/no]
DB4Import=[yes/no]
SourceHostName=[IP Address of daemon host of the source SAN]
SourceSlicName=[SignOn Path of the source SAN]
SourceSlicInitiatorNumber=[Initiator/SLIC Number of the SV Router
in the source SAN]
DestinationHostName=[IP Address of daemon host of the destination
SAN]
DestinationSlicName=[SignOn Path of the destination SAN]
DestinationInitiatorNumber=[Initiator/SLIC Number of the SV Router
in the destination SAN]
[Host Adapter Info]
SourceHostAdapterUID=[WWN of the HBA connected to the source SV
Router]
DestinationHostAdapterUID=[WWN of the HBA connected to the
destination SV Router]
```

Caution ! ***Please wait until all I/O activity is completed before running this command. Otherwise this may cause lost I/Os and system panic.***

Usage:

```
sanimport haimport -e FileName {-v}
```

-e FileName	The SAN import configuration file name.
-v	Optional. User confirmation required.

Examples:

The following example will import the SAN configuration file using the information in the file FCsan08haimport.cfg.

```
sanimport haimport -e FCsan08haimport.cfg -v
```

FCsan08haimport.cfg:

```
[SAN Parameters]
DB2Import=yes
DB4Import=yes
SourceHostName=100.123.123.221
SourceSlicName=c0
SourceSlicInitiatorNumber=1
DestinationHostName=10.10.20.37
DestinationSlicName=c0
DestinationInitiatorNumber=1
[Host Adapter Info]
SourceHostAdapterUID=200000E08B0138ED
DestinationHostAdapterUID=20000000C923839C
```

SV Router Statistics

Use the **svstat** command to view the SV Router vital statistics. If the SV Router (**-t Ixxxxx**) is not specified, the program will display the vital statistics of all the SV Routers in the SAN.

Both the SV Router FC-FC 3's host-side and the device-side interfaces provide statistical data for the following:

Link Failure Count	This count reports the number of times the SV Router's Frame Manager detects a not operational state or other failure of N_Port initialization protocol.
Loss of Synchronization Count	This count reports the number of times that the SV Router detects a loss in synchronization.
Loss of Signal Count	This count reports the number of times that the SV Router's Frame Manager detects a loss of signal.
Primitive Sequence Protocol Error	This count reports the number of times that the SV Router's Frame Manager detects N_Port protocol errors.
Invalid Transmission Word	This count reports the number of times that the SV Router 8B/10B decorder did not detect a valid 10-bit code.
Invalid CRC Count	This count reports the number of times that the SV Router received frames with a bad CRC and a valid EOF. A valid EOF includes EOFn, EOFt, or EOFdti.

The SV Router's power must be cycled to reset the counter. Therefore, you should check the accumulation of errors between a fixed time.

Usage:

svstat -d Cx {-t Ixxxxx -h Host}

-d Cx	Cx is the SignOn path, as specified in the config file.
-t Ixxxxx	Optional. Initator/SLIC number of the SV Router.
-h host	Optional. The name or IP address of the host running the daemon.

Examples:

The following example will display the vital statistics for SV Router I00001. The svstat command is executed from a remote host to the daemon (r0) running on server (100.1.1.193).

svstat -d r0 -t I1 -h 100.1.1.193

Error Log Analysis Commands

The **sreadlog** command is used to read and display the error or event logs.

The SLIC Daemon creates an error log file (SignOnPath_SLICERR.LOG) to track all of the errors and events in the SAN. The sreadlog command analyzes the error log and displays the appropriate Service Request Number (SRN) for errors or events that need action. Log entries are returned in the following general format:

TimeStamp:nnn.Txxxxx.uuuuuuuu.SRN=mmmmm

TimeStamp	Time and date when event occurred.
nnn	SignOn Path.
Txxxxx	The device that reported this event.
uuuuuuuu	Unique ID of the device.
mmmmm	The SRN associated with this event (see Appendix A 'SRN Reference Table' on page 85).

Note: Do not remove or move the error log file (SignOnPath_SLICERR.LOG) while the SLIC Daemon is activated.

Caution ! ***Do not remove or move the error log file (CO_slicerr.log) while the SLIC Daemon is activated.***

Reading the Error Log [sreadlog]

Use the **sreadlog** command to read the error log.

Usage:

sreadlog -d Cx {-f File_Name -h Host -v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-h Host	Optional. The name or IP address of the host running the daemon.
-f File_Name	Optional. Output event logs to File_Name.
-v	Optional. Print event logs to console.

Examples:

The following example will display the event logs. The **sreadlog** command is executed on a server running the daemon (r0).

```
sreadlog -d r0
```

The following example will output the event logs to a file (mylog.txt). The **sreadlog** command is executed from a remote host to the daemon (r0) running on server (myhost).

```
sreadlog -d r0 -f mylog.txt -h myhost
```

Clearing the Check Mode [sclrlog]

Use the **sclrlog** command to mark event entries related to a device as old in the daemon's error log file. Old entries are not displayed the next time you run the **sreadlog** command.

Usage:

```
sclrlog -d Cx -t [all/Txxxxx/Iyyyyy] {-h Host}
```

-d Cx Cx is the SignOn path, as specified in the config file.

-t [all/Txxxxx/Iyyyyy] Specific device or all devices.

- **all** Mark all event entries as old.
- **Txxxxx** Target number of a drive that the host can access.
- **Iyyyyy** Initiator/SLIC Number of SV Router.

-h Host Optional. The name or IP address of the host running the daemon.

Examples:

The following example will clear events related to T00005. The **sclrlog** command is executed on a server running the daemon (r0).

```
sclrlog -d r0 -t t5
```

The following example will clear all events. The **sclrlog** command is executed from a remote host to the daemon (r0) running on server (myhost).

```
sclrlog -d r0 -t all -h myhost
```

Command Line Diagnostic Commands

Disk Drive Diagnostics [sddiag]

Use the **sddiag** command to verify that the physical disk drive is operating properly. All use of the selected drive must be stopped before running this command.

The tests performed on the disk include: stop unit, start unit, read capacity, and read data.

This command is not intended to test logical devices.

Note: If this test finds an error, an error message will be displayed. If the test finds nothing wrong, 'Done' will be displayed.

Caution ! ***Do not run the disk drive diagnostic command if the drives are in use. This can cause lost I/Os and system panic.***

Usage:

```
sddiag -d Cx -t Txxxxx {-h Host}
```

-d Cx	Cx is the SignOn path, as specified in the config file.
-t Txxxxx	Target number of a physical drive that the host can access.
-h Host	Optional. The name or IP address of the host running the daemon.

Examples:

The following example will diagnose physical device T00005. The **sddiag** command is executed on a server running the daemon (r0).

```
sddiag -d r0 -t t5
```

SLIC (SV Router) Diagnostics [sudiag]

Use the **`sudiag`** command to verify the proper operation of the SV Router. Use of all drives connected to the SV Router must be stopped before running this command.

Executing this command starts a reboot test sequence that includes the following: buffer memory check, CPU memory check, FC chip check and Ethernet chip check.

Note: If this test finds an error, an error message will be displayed. If the test finds nothing wrong, 'Done' will be displayed.

Caution ! ***Do not run the router diagnostic command if the router is in use. This can cause lost I/Os and system panic.***

Usage:

```
sudiag -d Cx {-h Host}
```

-d Cx	Cx is the SignOn path, as specified in the config file.
-h Host	Optional. The name or IP address of the host running the daemon.

Examples:

The following example will diagnose the SV Router communicating with the daemon (r0).

```
sudiag -d r0
```

Displaying VPD (Vital Product Data) [svpd]

Use the **svpd** command to display important information (Vital Product Data or VPD) for the physical device selected.

```
Vendor ID - xxxxxx
Product Type - xxxx
Model Number - xxx
Microcode Revision - xxxx
Unit Serial Number - xxxxxxxx
Unique ID – xxxxxxxxxxxxxxxx
```

Note: **svpd** only displays the information for a physical device. No logical information is given.

Usage:

svpd –d Cx –t [s/Txxxxx] {–f File_Name -h host –v}

-d Cx	Cx is the SignOn path, as specified in the config file.
-t [s/Txxxxx]	Target number of the physical device or the SignOn Router.
s	SignOn Router.
Txxxxx	Target number of the physical device.
-f File_Name	Optional. Output VPD to File_Name.
-h Host	Optional. The name or IP address of the host running the daemon.
-v	Optional. Print VPD to console.

Examples:

The following example will display the VPD of T00005. The **svpd** command is executed on a server running the daemon (r0).

```
svpd -d r0 -t t5
```

The following example will display the VPD of the SignOn router. The **svpd** command is executed on a server running the daemon (r0).

```
svpd -d r0 -t s
```

The following example will display the VPD of the SignOn router. The **svpd** command is executed from a remote host to the daemon (r0) running on server (myhost).

```
svpd -d r0 -t s -h myhost
```

Sending FC Echo Frames

Use the **`svprobe`** command to send FC echo frame to a specified FC device from a source router and check and compare the response data with the transmitted data.

It is required for the user to specify the destination FC device. In addition, the user also needs to specify whether it is the host-side or device-side port of the SV Router where the echo frame is initiated.

By default, the payload for each transmission of the echo frame is 224 bytes. That is, if the user does not specify the data pattern file name or size of the echo packet, the **`svprobe`** command will generate a default pattern of 224 bytes to the destination device.

If the file name of the echo data pattern is specified, the **`svprobe`** command will send this data, and the size of the file will be used. The file size should be less than or equal to 224 bytes.

If the file name is not specified, but the size of the echo packet is given, the program will use the default pattern of the specified size and send an echo frame to the destination device.

Optionally, the user can also specify the number of times that the echo frame is to be sent to the destination device. If this count is -1, the program will keep sending the frame until the user issues the Ctrl-C signal. The default setting is 1.

Optionally, the user can also specify the timeout value for this send and receive echo frame operation. The maximum timeout value is 15 seconds. The default value is 2 seconds.

If no source SV Router is specified, the Master SV Router (SLIC) will be used to initiate the echo frame. It is important to note that the source SV Router and the destination FC device cannot be the same.

Usage:

```
svprobe -d Cx -r [Ixxxxx/UID] -m [host/device] {-t [Iyyyyy] -l size
-c count -i timeout -f File_Name -h Host}
```

-d Cx	Cx is the SignOn path, as specified in the config file.
-r [Ixxxxx/UID]	Destination FC device.
Ixxxxx	SLIC/Initiator Number of the SV Router
UID	WWN of the FC Device
-m [host/device]	Initiate echo from SV Router's host side or device side port.
-t Iyyyyy	Optional. SLIC/Initiator Number of the SV Router initiating the echo.

-l size	Optional. Size of echo packet. Default size is 224.
-c count	Optional. Iteration count.
-i timeout	Optional. Timeout value in seconds.
-f File_Name	Optional. Echo data pattern file.
-h Host	Optional. The name or IP address of the host running the daemon.

Examples:

Below is an example in sending an echo request data from SV Router (I00001) to destination FC device (WWN - 2000000000000111) through its device side connection. The echo request data is to be sent three times. The **`svprobe`** command is executed on a server running the daemon (r0).

```
svprobe -d r0 -t I1 -r 2000000000000111 -m device -c 3
```

Command Line Error Codes

The following error codes may appear instead of error messages when using the Command Line Interface.

Error Code	Description
5xx	SV Router reported a software operation related error.
513	No action taken by the SV Router.
538	SV Router is unreachable.
550	Remote SV Router failed to respond to router to router communication.
543	Remote SV Router failed to obtain configuration information.
560	Unknown virtual drive.
40002	Vicom Hardware type not supported.
40004	Vicom Hardware Module has incompatible firmware version.
40040	Invalid target (or SLIC / SV Router) number.
40041	Target device is inactive (or offline).
40042	Target device has responded with unit not ready.
40043	Target device is currently not responsive to command requests.
40072	Invalid FC map.
40073	Invalid map value.
40074	Invalid map type.
40077	Drive is not mapped.
40080	Invalid SLIC / SV Router number.
40081	SV Router is offline.
40082	SV Router is unresponsive.
40083	Consistency error detected. No Master SLIC detected in SAN.
40103	Exceeded maximum number of logical drives.
40702	Reading SAN configuration failed.

Table 3-1 CLI Error Codes

APPENDIX A

SRN REFERENCE TABLE

SRN	Description	Corrective Action
1xxxx	Disk drive Check Condition status. xxxx is the Unit Error Code. (The Unit Error Codes are returned by the drive in Sense Data bytes 20-21 in response to the SCSI ***Request Sense*** command.)	Refer to the Unit Error Code (UEC) in the subsystem service guide. **
2A002 2A003 2A004	These codes are the result of certain SSA ***Asynchronous Alert*** messages received from SSA drives that indicate a serious error condition in the drive. *	Refer to the Unit Error Code (UEC) in the subsystem service guide. **
2A005 2A006	These codes indicate that a disk drive module has detected the loss of redundant power or cooling.	Refer to the Unit Error Code (UEC) in the subsystem service guide. **
2A106	This code indicates that multiple disk drive modules have detected the loss of redundant power or cooling. It is probable that the power or cooling problem is with the chassis in which the drives are located	Refer to the Unit Error Code (UEC) in the subsystem service guide. **
43P00	One of the SLIC's SSA ports has shut down. This is caused by too many physical level SSA errors detected by the port within a short time period. Invoking the ***Clear Check Mode*** function will cause the SLIC to attempt to restore the port to normal operation. P is the port that has shut down (port A1=0 and port A2=1).	Ensure connection between SLIC and drive is good. If need be, replace cable between SLIC and drive.
450xx (4500A and above are displayed in hexadecimal)	This code indicates that the SSA web is not in a loop configuration. The hop number is represented by x. A hop number ensures that the counting sequence for the SSA devices begins with port A1 and ends with port A2.	Check loop

Table A-1 Explanation of Service Request Numbers

70000	SAN Configuration has changed.	
70001	Rebuild process has started.	
70002	Rebuild is completed without error.	
70003	**Rebuild is aborted with a read error.** This means that the drive copying information can not read from the primary drive.	If a spare drive is available, it will be brought in and used to replace the failed drive. If no spare is available, replace the failed drive with a new drive.
70004	**Write error is reported by follower.** If the initiator is master, then its follower has detected a write error on a member within a mirror drive.	If a spare drive is available, it will be brought in and used to replace the failed drive. If no spare is available, replace the failed drive with a new drive.
70005	**Write error is detected by master.** If the initiator is master, then it has detected a write error on a member within a mirror drive.	If a spare drive is available, it will be brought in and used to replace the failed drive. If no spare is available, replace the failed drive with a new drive.
70006	Router to router communication has failed.	Internal error. Update firmware.
70007	**Rebuild is aborted with write error.** This means the primary drive can not write to the drive being built.	If a spare drive is available, it will be brought in and used to replace the failed drive. If no spare is available, replace the failed drive with a new drive.
70008	**Read error is reported by follower.** If the initiator is master, then its follower has detected a read error on a member within a mirror drive.	If a spare drive is available, it will be brought in and used to replace the failed drive. If no spare is available, replace the failed drive with a new drive.
70009	**Read error is detected by master.** If the initiator is master, then it has detected a read error on a member within a mirror drive.	If a spare drive is available, it will be brought in and used to replace the failed drive. If no spare is available, replace the failed drive with a new drive.
70010	CleanUp configuration table is completed.	
70020	SAN physical configuration changed.	If unintentional, check condition of drives.
70021	Drive is offline.	If unintentional, check condition of drives.
70022	SV Router is offline.	If unintentional, check condition of drives.
70023	Drive is unresponsive.	Check condition of drives.
70024	For T3 pack: Master Router has detected the partner SV Router's IP Address.	
70025	For T3 pack: Master Router is unable to detect the partner SV Router's IP Address.	Check the Ethernet connection between the two SV Routers.
70030	SAN configuration changed by SV SAN Builder.	
70040	Host zoning configuration has changed.	

Table A-1 Explanation of Service Request Numbers

70050	MultiPath drive Failover.	Check MultiPath drive.
70051	MultiPath drive Failback.	
70098	Instant Copy degrade.	If no spare is available, replace the failed drive with a new drive.
70099	Degrade because the drive has disappeared.	Reinsert the missing drive, or replace it with a drive of equal or greater capacity.
7009A	**Read degrade recorded.** A mirror drive was written to, causing it to enter the degrade state.	Reinsert the missing drive, or replace it with a drive of equal or greater capacity
7009B	**Write degrade recorded.** If a spare drive is available, it will be brought in and used to replace the failed drive.	The removed drive needs to be (if good) reinserted or (if bad) replaced.
7009C	**Last primary failed during rebuild**. This is a “multi-point failure” and is very rare.	Backup drive data. Destroy mirror drive where failure has occurred. Format (mode 14) drives. Create new mirror drive. Re-assign old SCSI ID and LUN to mirror drive. Restore data.
71000	Router to Router communication has recovered.	
71001	This is a generic error code for the SLIC. It signifies communication problems between the SV Router and the daemon.	1. Check the condition of the SV Router. 2. Check cabling between router and daemon server. 3. Error halt mode also forces this SRN.
71002	This indicates that the SLIC was busy.	1. Check the condition of the SV Router. 2. Check cabling between router and daemon server. 3. Error halt mode also forces this SRN.
71003	SLIC Master unreachable.	Check conditions of the SV Routers in the SAN.
71010	The status of the SLIC Daemon has changed.	
72000	Primary/Secondary SLIC Daemon connection is active.	
72001	Failed to read SAN Drive Configuration.	
72002	Failed to lock on to SLIC Daemon.	
72003	Failed to read SAN SignOn Information.	
72004	Failed to read Zone Configuration.	
72005	Failed to check for SAN changes.	
72006	Failed to read SAN event log.	
72007	SLIC Daemon connection is down.	Wait for 1-5 minutes for backup daemon to come up. If it doesn’t, check the network connection, for SV Router error halt, or hardware failure.

Table A-1 Explanation of Service Request Numbers

80210	SSA Enclosure is warning that the controller card has failed.	Replace the FRU.
80211	SSA Enclosure's Operator Panel has failed.	Replace the FRU.
80221 80222	SSA Enclosure's power supply, in position 1 or position 2, has failed. 80221 (position 1) 80222 (position 2)	Go to "MAP 2020 7133 - Power."
80231 80232 80233	SSA Enclosure's fan unit, in position 1, 2, or 3 has failed. 80231 (position 1) 80232 (position 2) 80233 (position 3)	Replace the FRU.
8024x	SSA Enclosure's bypass card has failed. Where: x = 1 or 2 is bypassed 1-16 x = 3 or 4 is bypassed 4-5 x = 5 or 6 is bypassed 8-9 x = 7 or 8 is bypassed 12-13	Replace the FRU.
80251	SSA Enclosure has detected Critical Over Temperature.	
80252	SSA Enclosure has detected that the Over Temperature Warning level has been exceeded.	
80253	SSA Enclosure has detected Critical Under Temperature.	
80254	SSA Enclosure has detected that the Under Temperature Warning level has been exceeded.	
80260	SSA Enclosure is warning that an Empty Disk Drive Slot can affect the enclosure cooling. Install a disk drive or a dummy carrier.	
*	**An error that is reported via *Async_Alert*, such as *"Loss of Redundant Power/Cooling*," will be logged only in the system that contains the SSA Master SLIC.**	
**	**Subsystem Service guide documents include:** **7133 SSA Disk Subsystem for Open Attachment: Service Guide, SY33-0191** **7131 Model 405, SSA Multi-Storage Tower for Open Attachment: Service Guide, SY32-0405**	

Table A-1 Explanation of Service Request Numbers

APPENDIX B

DRIVE UEC (UNIT ERROR CODE) REFERENCE TABLE

UEC	Description
00 00	No error.
01 01	Degrade Mode – Motor not running.
01 02	Unavailable while Start/Stop Unit Command active.
01 03	Unavailable while Bring-up active.
01 04	Unavailable while Format active.
01 06	Requested P List does not match returned list format. **(READ DEFECT DATA only)**
01 07	Requested G List does not match returned list format. **(READ DEFECT DATA only)**
01 08	Defect List Error prevented one or more defects from being used in a Format Unit command or from being reported in a Read Defect Data command.
01 09	LBA or PBA Conversion Timeout Error
01 0A	Defect list longer than 64K, 64K of data returned. **(READ DEFECT DATA only)**
01 0C	NOID Table integrity Error detected.
01 0D	SSI Port read error detected.
01 0E	SSI port write error detected.
01 0F	Error detected while reading the NOID Table from disk.
01 10	Too few valid GEM measurements available to perform a GEM Predictive Failure Analysis.
01 11	Degrade Mode – Reassign Blocks unsuccessful after pushdown started.

Table B-1 Unit Error Codes for Drives

01 12	Degrade Mode – Format unsuccessful.
01 13	Degrade Mode – Configuration not loaded.
01 16	ROS Microcode Download failed.
01 17	NOID Table build failed during a Format Unit command.
01 1B	Motor Start Failed due to Timer 1 being disabled.
01 1C	Command not allowed while in Write Protect Mode.
01 1F	Mismatch between the Servo Processor and the Reference Track Image.
01 20	Microcode Check Sum error detected during download of Microcode.
01 21	Mismatch between the Interface Processor ROS and Servo Processor ROS.
01 22	Degrade Mode – Bring-up not successful.
01 23	Failure to load Servo Microcode into RAM.
01 24	Mismatch between the Servo Processor ROS and DE.
01 27	Buffer Controller Chip Channel A Error – Parity error during transfer in.
01 28	Buffer Controller Chip Channel A Error – Parity error during transfer out.
01 29	Buffer Controller Chip Channel A Error – Programmed IO Parity error.
01 2A	Buffer Controller Chip Channel A Error – Unexpected Error.
01 2B	Command aborted due to Fatal Hardware error.
01 2C	Self Initiated Reset – Host Interface Chip internally detected error.
01 2D	Cannot resume the operation (data transfer).
01 2F	Mismatch between the Interface Processor ROS and the DE.
01 30	Invalid Operation code.
01 31	Invalid LBA.
01 32	CDB Invalid.
01 33	Invalid LUN.
01 34	Command parameter data invalid.
01 35	Command parameter list length error.
01 36	Microcode and Load ID mismatch during Write Buffer Command.
01 37	Data length error on Read Long or Write Long.
01 38	Invalid field in Parameter Data – See Field Pointer Value.
01 39	Invalid LBA in Reassign Command when Reassign degraded.
01 3A	Invalid Buffer ID in Write Buffer Command.

Table B-1 Unit Error Codes for Drives

01 3B	Microcode and Servo Processor ROS mismatch during Write Buffer Command.
01 3C	Microcode and DE mismatch during Write Buffer Command.
01 3D	Microcode and Interface Processor ROS mismatch during Write Buffer Command.
01 3E	Microcode and Interface Processor RAM mismatch during Write Buffer Command.
01 3F	Self Initiated Reset – Host Interface detected an LRC error during read.
01 40	Unit Attention – Not Ready to Ready Transition. (Format Completed)
01 41	Unit Attention – POR.
01 42	Unit Attention – Mode Selected Parameters have changed.
01 43	Unit Attention – Write Buffer.
01 44	Unit Attention – Command cleared by another Initiator.
0145	Unit Attention – Self Initiated Reset.
01 46	Unit Attention – Inquiry Parameters have changed.
01 49	Unit Attention – Log Parameters Changed.
01 4D	DSI – SCSI Response Packet Check.
01 4E	DSI – Enclosure Unavailable.
01 4F	DSI – Enclosure Transfer Failure.
01 50	Microcode Check Sum error detected During ROS Test.
01 51	Microcode Check Sum error detected During RAM Test.
01 56	GLIST full – cannot add more entries.
01 57	The defects-per-track limit was exceeded during a Format or Reassign operation.
01 5A	Motor is Stuck, Cannot be started.
01 5C	Reassign could not find the target LBA.
01 5D	No Sector Found caused by hardware fault or software.
01 5E	No spare sectors remaining.
01 5F	Error in Primary Defect list.
01 62	Invalid Initiator Connection (duplicate tag).
01 63	Media Problem, Recommend Device Replacement.
01 64	Hardware Problem, Recommend Device Replacement.
01 65	Error in Primary Defect list. **(READ DEFECT DATA only)**
01 66	Error in Grown Defect list. **(READ DEFECT DATA only)**
01 69	Cylinder/Head target mismatch between Servo and Fileside.

Table B-1 Unit Error Codes for Drives

01 6A	Servo Error – Invalid Servo Status Received by the Interface Processor.
01 6B	Arm Electronics Not Ready.
01 6C	Sanity Error during Read Capacity execution.
01 80	SP interrupt on but SP Status Valid bit is off.
01 81	Error in Grown Defect list (used by Format Unit and Reassign Block commands).
01 84	Invalid SP Command Sequence.
01 85	Illegal Head or Cylinder requested.
01 86	A servo command is already active.
01 87	Interface Processor detected Servo Sanity Error.
01 88	Controller/Channel Hardware detected Servo Sanity Error.
01 89	Reserved area sector valid check failed.
01 8A	Servo processor did not finish command in time.
01 8B	Motor timeout error.
01 8C	Configuration or Controller Data Sector valid check failed.
01 8D	Configuration or Controller Data Sector uploaded but Check Sum error.
01 8E	Reserved area sector version check failed.
01 8F	Buffer too small to do a requested function.
01 90	Self Initiated Reset – Invalid Input.
01 92	Miscompare during byte-by-byte verify.
01 93	BATS#2 Error – Read/ Write test failure.
01 94	BATS#2 Error – ECC/CRC test failure.
01 95	BATS#2 Error – Seek test failure.
01 96	BATS#2 Error – Head Offset Test failure.
01 97	Self Initiated Reset – No task available.
01 98	Self Initiated Reset – Cause Unknown.
01 9A	Self Initiated Reset – Buffer Controller Chip Reset unsuccessful.
01 9B	Self Initiated Reset – Zero Divide Error.
01 9C	Self Initiated Reset – Control Store Address Fault.
01 9D	Self Initiated Reset – Unused Op Code.
01 9E	Invalid Hall sensor count error.
01 9F	Self Initiated Reset – Invalid Queue Operation.

Table B-1 Unit Error Codes for Drives

01 A0	Error reading NoID tables.
01 A1	Controller “Ready” Timeout Error.
01 C0	Too many missing Servo IDs detected by the Controller or Channel Hardware.
01 C1	Arm Electronics error.
01 C2	Fake and Extra Index.
01 C3	SP lost.
01 C4	Sector overrun error.
01 C5	Interface Processor write inhibit error.
01 C6	Read Write Ready dropped error.
01 C7	Microjog Write Inhibit.
01 C8	Interrupt Occurred with no interrupt bits set.
01 C9	Write with No Sector Pulses.
01 CB	Motor Speed Error.
01 CD	IP Retract Error.
01 D1	No Data Field sync byte found.
01 D2	Data ECC Check.
01 D3	Data correction applied to Drive data for a Data ECC check.
01 D4	ECC check corrected without using ECC correction.
01 D5	Data Sync error detected while outside of the write band.
01 D6	Data ECC Check detected while outside of the write band.
01 D7	ECC Error Detected while outside band corrected with ECC.
01 D8	ECC Error Detected while outside of write band corrected without ECC.
01 D9	Data recovered using positive offsets.
01 DA	Data recovered using negative offsets.
01 DB	Erroneous sync byte found.
01 F2	Buffer Controller Chip Error – Invalid interrupt error.
01 F8	Buffer Controller Chip Error – Channel parity error on read.
01 FD	Buffer Controller Chip Error – Channel error during a transfer from the Control Store RAM to the Data Buffer.
01 FE	Buffer Controller Chip Error – Drive pointer updated incorrectly.
02 00	Buffer Controller Chip Error – ECC On The Fly timeout.
02 21	Buffer Controller Chip Error – Pipeline already full.

Table B-1 Unit Error Codes for Drives

02 02	Buffer Controller Chip Error – FIFO overun/underun.
02 03	Disk Manager Chip detected a CRC error during write.
02 04	Disk Manager Chip detected a second index pulse.
02 05	Disk Manager Chip detected a sector pulse with current split loaded.
02 16	Servo ID overrun Error.
02 18	Arm Electronics (AE) Idle Error.
02 19	Interface Processor Ready Timeout Error.
02 1B	External Write Inhibit.
02 1C	Error in Target Sector Generation logic.
02 1D	SID Overrun during a Read Operation.
02 1E	Start Pipeline while in the Very Busy state.
02 1F	Parity Error detected in the Skip Sector FIFO.
02 20	Channel Noise Problem, Recommend Device Replacement.
02 21	Channel Assymetry Problem, Recommend Device Replacement.
02 22	Channel Precompensation Problem, Recommend Device Replacement.
02 23	Channel DC Offset Problem, Recommend Device Replacement.
02 24	Channel Timing Offset Problem, Recommend Device Replacement.
02 25	Fly Height Change Problem, Recommend Device Replacement.
02 26	Torque Amplification Problem, Recommend Device Replacement.
02 34	NoID Sequence Error.
02 37	DAC Optimization Error.
02 40	Skip Sector FIFO overflow error.
02 41	Data Sector ID Counter error.
02 42	Skip Sector FIFO Load Check.
02 43	SID Counter error.
02 44	Split Field Table CRC error.
02 45	PRSI Write error.
02 47	Servo Window error.
02 48	No Read or Write Gate after a Sector Pulse.
02 4A	Error detected loading the Split Field Table.
02 4B	IP Register write integrity error.

Table B-1 Unit Error Codes for Drives

02 4C	SID Estimator Error.
02 52	Servo Error – Power dissipation too high.
02 53	Servo Error – Gray code window exceeds 9 tracks (Position).
02 54	Servo Error – Gray code window exceeds 3 tracks (Track Following).
02 55	Servo Error – KF Adjustment too large.
02 57	Servo Error – AE Serial Port read back miscompare.
02 58	Servo Error – AE Select error (wrong AE selected).
02 59	Servo Error – Detected AE serial sequence error.
02 5A	Servo Error – AE serial port timeout (stuck busy).
02 5B	Servo Error – Head Control Port not taking effect (2ms timeout).
02 5C	Servo Error – Spindle Serial Port timeout.
02 60	Recalibrate request from Servo.
02 61	Servo Error – Loss of interrupts from the Controller or Channel Hardware.
02 62	Servo Error – Settle timeout.
02 63	Servo Error – Three consecutive missing SIDs (track following).
02 64	Servo Error – Three consecutive missing SIDs (settle).
02 65	Servo Error – Three consecutive missing SIDs (seeking).
02 66	Servo Error – Half track timeout.
02 67	Servo Error – Seek timeout.
02 68	Servo Error – Target cylinder out of range.
02 69	Servo Error – Command not accepted while actuator retracted.
02 6A	Servo Error – 3 consecutive invalid Gray Codes.
02 6B	Servo Error – Estimator Error Saturated.
02 6C	Servo Error – Head number out of range.
02 6D	Servo Error – Target Cylinder and Gray Code Cylinder mismatch.
02 6E	Servo Error – Invalid command.
02 6F	Servo Error – Estimator Position out of range.
02 70	Servo Error – Offset out of range.
02 71	Servo Error – No sector orientation after breaking free of latch.
02 72	Servo Error – Unable to break free of latch or no SID detected.
02 73	Servo Error – Unable to break free or latch or no SID detected in Servo internal recovery mode.

Table B-1 Unit Error Codes for Drives

02 74	Servo Error – Unable to achieve track following after breaking free of latch.
02 75	Servo Error – Not track following when recalibrate clean-up was entered.
02 76	Servo Error – Seek timeout during clean-up seek to cylinder zero.
02 77	Servo Error – Unable to achieve track following after breaking free of latch in Servo internal recovery mode.
02 7E	Servo Error – AGC not calibrated.
02 7F	Servo Error – AGC limit exceeded.
02 80	Servo Error – Command not accepted – motor not spinning.
02 81	Servo Error – Seek timeout during KT Calibration.
02 82	Servo Error – Command not accepted – SRAM not loaded.
02 83	Servo Error - Seek timeout during AGC calibration.
02 84	Servo Error – Invalid Recalibrate command qualifier.
02 85	Servo Error – AGC saturated during calibration.
02 8C	Servo Error – Loss of Sync, missing SID 6 out of 8 times.
02 8D	Servo Error – ASIC/SP Sector Count mismatch.
02 8E	Servo Error – Unexpected Retract Current Value.
02 8F	Servo Error – Unexpected GC Velocity Error.
02 90	BATS#2 Error – Split Field Table check sum.
02 91	BATS#2 Error – Split Field Table address.
02 92	BATS#2 Error – Split Field Table stuck bit.
02 93	BATS#2 Error – CRC Test failure, mode 1.
02 94	BATS#2 Error – CRC Test failure, mode 2.
02 95	BATS#2 Error – CRC Test failure, mode 3.
0x xx	Unrecovered error during Reassign after pushdown has started. This error can be associated with a number of UEC's.
4x xx	Thermal Asperity Detected during error.
8x xx	Invalid UEC – x xx is Invalid UEC.

Table B-1 Unit Error Codes for Drives

APPENDIX C

COMMAND LINE INTERFACE QUICK REFERENCE

Viewing Device Configuration Commands

Listing all physical and logical devices present:

```
showmap -d Cx {-m [lists_option] -f File_Name -h Host -v}}
```

SLIC Daemon Commands

Listing the SignOn paths for the SLIC Daemon processes running on each host:

```
sgetname -h Host
```

Setting the master daemon in the SAN:

```
setmasterdaemon -d Cx {-h Host -f}
```

Displaying the the communication between the SLIC Daemon and the SV Router:

```
signoninfo –d Cx {–h Host}
```

Enabling or Disabling remote access to the daemon server:

```
setmode {-o [enable/disable]}
```

Adding client host IP address(es) for remote access to daemon:

```
clientip add –h IP_Address {-v}
```

Deleting client host IP address(es) from the list of hosts that can access the daemon:

```
clientip del –h IP_Address {-v}
```

Listing the client host IP addresses that are enabled for remote access to the daemon:

```
clientip view
```

Listing the IP addresses of the hosts connected to the SLIC Daemon and with what application they are connecting:

```
connlist {-d Cx -h host}
```

Displaying the SLIC Daemon configuration information:

```
saninfo {-d Cx -h host}
```

Simple Drive Commands

Changing a simple drive to a general spare drive:

```
drvprop spare -d Cx -t Txxxxx {-h Host -v}
```

Changing the global map of a simple drive:

```
drvprop change -d Cx -t Txxxxx -s [TxDy/Lxxx] {-h Host -v}
```

MultiPath Drive Commands

Autocreating a MultiPath drive:

```
mpdrive autocreate -d Cx {-h Host -v}
```

Removing a MultiPath drive:

```
mpdrive remove -d Cx -m Txxxxx {-h host -v}
```

Using MultiPath drive failback:

```
mpdrive failback -d Cx -j UID {-h host -v}
```

Replacing a MultiPath drive:

```
mpdrive replace -d Cx {-h host -f -v}
```

Changing MultiPath drives:

```
mpdrive change -d Cx -m Txxxxx -s map {-h host -v}
```

Viewing MultiPath drive properties:

```
mpdrive view -d Cx {-m Txxxxx -h Host}
```

Disk Pool Commands

Creating a disk pool:

```
vdiskpool create -d Cx -n PoolName {-t [Txxxxx/all] -h host -v}
```

Adding drives to a disk pool:

```
vdiskpool add -d Cx -t [Txxxxx/all] {-p PoolName -h Host -v}
```

Deleting drives from a disk pool:

```
vdiskpool del -d Cx -t [Txxxxx/all] {-p PoolName -h host -v}
```

Removing a disk pool:

```
vdiskpool remove -d Cx {-p PoolName -h host -v}
```

Modifying the name of a disk pool:

```
vdiskpool change -d Cx -p PoolName -n NewPoolName {-h host -v}
```

Viewing a disk pool:

```
vdiskpool view -d Cx {-p PoolName -m [physical/logical/all] -h
host}
```

Viewing disk pool statistics:

```
vdiskpool stat -d Cx {-p PoolName -h host}
```

Virtual Drive (VLUN) Commands

Creating a virtual drive (virtual LUN):

```
vlun create -d Cx -l size {-n VdriveName -p PoolName -s map -t
Txxxxx -h host -v}
```

Autocreating a virtual drive (virtual LUN):

```
vlun autocreate -d Cx -c count -l size {-n MyVDrive -p PoolName -h
host -v}
```

Removing a virtual drive (virtual LUN):

```
vlun remove -d Cx -t Txxxxx {-h host -v}
```

Changing a virtual drive (virtual LUN):

```
vlun change -d Cx -t Txxxxx -n VdriveName -s map {-h host -v}
```

Viewing virtual drive (virtual LUN) properties:

```
vlun view -d Cx {-p PoolName -t Txxxxx -h Host}
```

SLIC (SV Router) Commands

Downloading Microcode:

```
sdnld -d Cx -t [s/Ixxxxx/sa] -f File_Name {-h Host}
```

Creating a SLIC (SV Router) alias:

```
slicalias change –d Cx –t Ixxxxx –n New_Name {–h Host -v}
```

Changing a SLIC (SV Router) alias:

```
slicalias change –d Cx –t Ixxxxx –n New_Name {–h Host -v}
```

Clearing a SLIC (SV Router) alias:

```
slicalias clear -d Cx -t Ixxxxx {-h Host -v}
```

Viewing a SLIC (SV Router) alias:

```
slicalias view –d Cx –t Ixxxxx {–h Host}
```

Forcing the local SV Router to be the master SV Router:

```
smaster -d Cx {-t Ixxxxx -h Host}
```

Reading SAN configuration file and saving to file:

```
sanconfig read -d Cx -e FileName {-m SANCfgType -h host -v}
```

Writing SAN configuration file to SV Router:

```
sanconfig write -d Cx -e FileName {-m SANCfgType -h host -v}
```

Importing the Zone information from a SAN configuration file to a SV Router that has been replaced in a multi-router environment:

```
sanconfig import -d Cx -e FileName -r NewSLIC# -j CurrentSLIC# {-h
Host -v}
```

Importing the SAN configuration file from the source SAN to the destination SAN:

```
sanimport haimport -e FileName {-v}
```

Viewing the SV Router statistics:

```
svstat -d Cx {-t Ixxxxx -h Host}
```

Error Log Analysis Commands

Reading the error log:

```
sreadlog –d Cx {–f File_Name -h host –v}
```

Clearing the check status (CK) or power cooling (PC):

```
sclrlog -d Cx -t [all/Txxxxx/Iyyyyy] {-h Host}
```

Command Line Diagnostic Commands

Checking disk drive diagnostics:

```
sddiag -d Cx -t Txxxxx {-h Host}
```

Checking SV Router diagnostics:

```
sudiag -d Cx {-h Host}
```

Displaying Vital Product Data (VPD):

```
svpd –d Cx –t [s/Txxxxx] {–f File_Name -h host –v}
```

Sends FC echo frame to a specified FC device.

```
svprobe -d Cx -r [Ixxxxx/UID] -m [host/device] {-t [Iyyyyy] -l size
-c count -i timeout -f File_Name -h Host}
```

APPENDIX D

SYSTEM PANIC/DATA LOSS COMMAND LIST

Issuing the following commands to an SV Router or a drive that is being used for I/O transmission may cause system panic or data loss. Please wait until all I/O activity is completed before running these commands.

Command	Description
drvprop spare	Changes a physical drive to a general spare drive.
drvprop change	Changes the properties of a general spare drive.
vdiskpool add	Adds drives to a disk pool.
vdiskpool del	Deletes drives from a disk pool.
vdiskpool remove	Removes a disk pool.
vlun create	Creates a virtual drive (virtual LUN).
vlun autocreate	Automatically creates a virtual drive (virtual LUN).
vlun remove	Removes a virtual drive (virtual LUN).
vlun change	Changes the virtual drive (virtual LUN) mapping.
mpdrive autocreate	Automatically creates a multipath drive.
mpdrive remove	Removes a multipath drive.
mpdrive change	Changes the multipath drive mapping.

Table 3-2 Possible commands that could cause system panic and/or data loss

Command	Description
sdnld	Downloads SV Router microcode.
sanconfig write	Writes the SAN configuration file to the SV Router
sanconfig import	Imports the Zone information from a SAN configuration file to a SV Router that has been replaced in a multi-router environment.
sanimport haimport	Imports the SAN configuration file from the source SAN to the destination SAN.
sddiag	Performs disk drive diagnostics.
sudiag	Performs SV Router diagnostics.

Table 3-2 Possible commands that could cause system panic and/or data loss

GLOSSARY

async alert — A signal sent by a drive or a storage area router to inform the user that an error has occurred with the originator of the signal.

auto rebuild — The storage router automatically replaces the failed drive with the spare drive. Router then copies the data from the primary drive to the spare drive, which is now a member of the mirror drive.

available drive pool — A list of usable, functional drives. This includes composite, simple, and general spare drives.

command line interface — A program that accepts commands as typed-in phrases for both UNIX and NT operating systems.

complex drive — A group of storage drives that contains a single ID and LUN. Complex drives can be mirror, composite, mirror composite or multipath.

composite drive — A combination of multiple drives that are seen by the host computer as one. The host sees one drive with the capacity of all the drives combined. Maximum number of drives that a user may combine is eight. When writing to this drive, the information is written in a sequential manner.

concatenation — See composite drive.

configuration file (config file) — The configuration (config) file defines the function of the SLIC daemon.

daemon — See SLIC daemon.

daemon server — The server used to run the SLIC daemon.

dedicated spare	A drive assigned to replace any failed drive within a designated mirror set.
delete Instant Copy	Removes Instant Copy member from a mirror drive.
device router	The router connected to the storage loop.
disk partition	A designated section of memory created on a disk drive.
disk pool	The disk pool is a group of drives from which virtual drives are created. The group of drives that make up the disk pool are called pool drives. Pool drives are created from mapped drive(s), unmapped drive(s), spare drive(s), or multipath drive(s).
DMP	An acronym for dynamic multi-pathing. A software based process that provides and manages multiple data paths. It provides load balancing across multiple I/O channels and if a path fails, it redirects the data through an alternate route.
encapsulation technique	Creating a partition on a drive for use by the storage router.
Ethernet communication	Also called out-of-band communication. SAN connection where control-related signals are transmitted through TCP, rather than in-band with the data.
failover	Automatic and seamless possession of a device's operations when it fails.
FC-AL	An acronym for Fibre Channel – Arbitrated loop. A form of Fibre Channel network in which up to 127 nodes are connected in an arbitrated loop topology. All devices share the same bandwidth and only two devices can communicate with each other at the same time.
FC Node	Fibre Channel Architecture. Any device on the FC-AL loop.
GBIC	An acronym for Gigabit Interface Converter. An interface that converts serial optical signals to serial electrical signals and vice versa. The GBIC is designed to transmit signals via Fibre Channel and Ethernet protocol. It can be designed for use with an optical or copper path. The GBIC is also hot-swappable.
general spare	A spare drive prepared to replace any failed mirror drive.

heartbeat	A signal used to identify and ensure that paired failover devices in the network are functioning. Once the partner no longer detects the heartbeat signal then the device will perform failover.
heterogeneous	Dissimilar. In storage it usually refers to servers or storage that have differing protocol (SCSI, FC, SSA etc.) and exist within the same network.
host	The computer that is coordinating the functions of the (local) SV Router in use.
host bus adapter	A device that connects one or more peripheral units to a computer.
host router	The router connected to the host computer.
host server	The computer that is coordinating the functions of the target router in use.
hot plugging (hot swapping)	The connection and disconnection of peripherals or other components without interrupting system operation.
in-band communication	SAN connection where both control-related signals and data are transmitted through the same path.
initiator	A device that originates a signal or a command.
Instant Copy	An Instant Copy drive will duplicate the data on any mirror drive (two-way or three-way) without interrupting normal operating functions.
IOCB	I/O Control Block. It restricts the number of I/O commands sent from the Host Buffer. When the IOCB count is reached, it will issue a "Queue Full" message to the corresponding HBA. Limiting the Queue Depth keeps the host adapters from issuing too many commands, which can slow down system performance.
IOPS	Input/Output Per Second. It is the number of inputs and outputs or read/writes per second.
Ixxxxx	The initiator's identification number.
local SLIC	The SV Router that is attached to the host computer running the daemon.

logical drive	A group of drives that contain a single ID and LUN. Logical drives can be mirror, composite, mirror composite, Instant Copy or multipath.
logical volume	A designated section of memory created on a disk drive.
logical unit number (LUN)	The SCSI identifier of a logical unit within a target. Each SCSI ID can be divided into eight (0-7) logical units. These logical units can represent whole disks. This identifying number determines the device's priority.
LUN mapping	The ability to change the virtual LUN number as presented to the server from the storage. This allows such benefits as the ability for a server to boot from the SAN without the requirement of a local disk drive. Each server requires LUN 0 to boot.
LUN masking	Enables an administrator to dynamically map an HBA to a specified LUN. This allows an individual server or multiple servers access to an individual drive or to multiple drives, and prohibits unwanted server access to the same drive(s).
management information base	See MIB.
mapped drive	A drive that is assigned an ID and/or LUN for addressing purposes.
mapping table	See SAN database.
master SLIC (master router)	This is the SV Router that controls the storage loop including the drive configuration. All changes to drives must come through this master.
member drive	A drive within a complex drive. Within a Mirror drive, a member can be a simple or a composite drive.
media	The permanent storage area of a drive.
MIB	Acronym for Management Information Base. A database that describes the objects of the a device monitored by SNMP agent.
microcode	An instructional program to enable the proper operations between electrical functions of the computer and its corresponding device(s).
mirror composite drive	A combined group of drives seen as one drive by the host and mirrored or copied by another drive or combined group of drives.

mirror drive	A group of two or three members that contain the same information. A member of a mirror drive can be a simple or a composite drive.
mirroring	Writing identical information to separate drives simultaneously. Also known as RAID Level 1.
multipath drive	A logical LUN or drive created to hide, from the data server, the active and passive paths to a disk array that does not support multi-initiator attach.
node	Any device on the storage loop.
node mapping table	See SAN database.
node table	See SAN database.
offline	Describes a device that is not connected to or not installed in the storage subsystem. A drive could be connected physically to the SAN, but if it is not turned on or not in ready mode, it is considered offline.
owner	The SV Router or SV Routers that have access to the corresponding drive.
one-way mirror	A drive that contains only one mirror member. A one-way Mirror Drive is designed specifically to transmit data from a physical or a composite drive to an Instant Copy drive. This feature is only useful with the Instant Copy command.
out-of-band communication	SAN connection where both control-related signals and data are transmitted through separate paths.
physical drive	A drive that exist in the storage subsystem. They can be mapped or unmapped drives.
primary member	The drive that is copied via mirroring by other drives.
pool drives	The name for drives in the disk pool.
private drive	A simple drive or a complex drive that can be accessed only by an authorized storage router.
public drive	A drive (simple or complex) that can be accessed by any router on the storage loop.

quick initialize	Prompts SV SAN Builder to write zeros to the first block of the disk. After this process is complete, the drive appears new to the host. The host then will review the drive's configuration again. It is not a full initialization.
RAID Level 5	Data is striped across three or more drives for performance, and parity bits are used for fault tolerance. The parity bits from two drives are stored on a third drive.
RMBPS	An acronym for Read MegaBytes Per Second. Displays the rate at which data is read from a specific drive within the storage loop.
SAN	Acronym for Storage Area Network. A high-speed network that connects storage devices. The SV Routers are the foundation of the Vicom SAN. They share a common backbone and enable communication between storage device such as; data servers, switches, and disk arrays. In certain cases, the combination of all these devices may also be referred to as a SAN.
SAN database	A data reference source for the configuration of the SAN. The database is shared among all the SV Routers in the SAN, and each SV Router retains a copy of the database. Each time a change occurs in the SAN, all SV Routers are updated.
SLIC	An acronym for Serial Loop IntraConnect. Often used to represent SV Router.
SCSI-FC Extender	Extends SCSI connectivity to 500 meters, overcoming the SCSI distance constraint.
SCSI ID	An acronym for Small Computer Serial Interface Identification. A unique number, given to each device on the SCSI bus. This identifying number determines the device's priority. The numbers range from 0-15, with 7 reserved for the host.
SCSI topology	A map or view of all the complex drives on the storage loop.
service and diagnostic codes	A code composed of numbers referring to problems and events within the storage subsystem. Presented through an LED readout on the SV Router.
service request number	See SRN.
serial loop	A loop of devices connected via fibre channel or SSA protocol.

SignOn drive	The logical or physical drive containing all the configuration data that is located on the storage or serial loop. The host communicates with the SAN through this drive.
SignOn path	The path that points to the location of the SLIC Partition on the sign-on drive.
SignOn router	The router attached to the host computer running the SLIC daemon, through which communication to the SAN is established.
simple drive	One storage drive that contains an ID and LUN. It is not a complex drive.
SLIC daemon	A software agent running on the host (either a local or remote server) that permits communication between the client and the subsystem (SV Routers and Drives).
SNMP	An acronym for Simple Network Management Protocol. A network protocol. Used with software (SNMP agent and manager) that monitors the network and transmit the information to the network administrator.
spare drive	See general spare.
SRN	An acronym for Service Request Number. A number used to notify the user of changes or problems that occur within the storage system
SSA	An acronym for Serial Storage Architecture. A storage loop from IBM with speeds that can reach 160 Mbps. The loop's design provides added security. If one drive fails, access to the storage loop is maintained.
SSA node	Any device on the SSA (Serial Storage Architecture) loop.
SSA topology	A map of the nodes on the SSA loop.
standby drive	An unmapped drive that is a member of a disk pool.
storage subsystem	A combination of disk drives and controllers.
storage capacity	The amount of data that can be stored on each drive or complex drive.
storage virtualization	The secure and dynamic pooling of diverse storage equipment across heterogeneous servers and clients.

SV Router	A Vicom developed hardware module in SVE, which serves as the fundamental building block in a SAN. It provides storage management functions that enable a Fibre Channel host to interface with and control all storage-related elements in a SAN.
SV SAN Builder	A Vicom developed software module in SVE, which creates virtual drives and logical drives on the SAN. Logical drives can be composite drive(s), mirror drive(s), general spare drives, and Instant Copy drives.
SV SNMP Agent	A Vicom developed software module in SVE, which stores and retrieves data from the SAN, and signals the SNMP manager when an event occurs.
SV Zone Manager	A Vicom developed software module in SVE, which enables the system administrator to map logical or physical storage to an HBA. This ability allows the administrator to allocate storage on demand.
target	The recipient of a command or a signal sent by the initiator.
target number	A number assigned to each drive on the loop, except unmapped drives.
target router	The router attached to the host computer.
three-way mirror	Triplicate drives that are created either by data simultaneously written to three separate drives or by data copied from one drive to another drive. Either method ensures that they become duplicates.
two-way Mirror	Duplicate drives that are created either by data simultaneously written to two separate drives or by data copied from one drive to another drive. Either method ensures that they become duplicates.
Txxxxx	The Target's identification number.
unmapped drive	A drive that has not been assigned an ID and/or LUN for addressing purposes.
virtual drive	A logical drive created from the free space of a disk pool.
VPD	An acronym for Vital Product Data. Information about a device that is stored on the device itself. It allows the device to be administered at a system or network level. Typical VPD information includes a product model number, a unique serial number, product release level, maintenance level, and other information specific to the device type.

web walk	The process of a device scanning the storage subsystem.
WMBPS	Acronym for Write MegaBytes Per Second. Displays the rate at which data is written to a specific drive within the storage loop.
zone	A dedicated path between a LUN and the HBA to which it is mapped.
zoning	The act of mapping a LUN(s) to an HBA(s).

INDEX

W