



VNXe Service Commands

Technical Notes

January 2013

Copyright © 2013 EMC Corporation. All rights reserved. Published in the USA.

Published January 2013

EMC believes the information in this publication is accurate of its publication date. The information is subject to change without notice.

The information in this publication is provided as is. EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose. Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC2, EMC, and the EMC logo are registered trademarks or trademarks of EMC Corporation in the United States and other countries. All other trademarks used herein are the property of their respective owners.

For the most up-to-date regulatory document for your product line, go to the technical documentation and advisories section on the EMC online support website.

Part Number: 300-013-320 VNXe Service Commands

Table of contents

Executive summary	9
Business case.....	9
Solution overview	9
Key results / recommendations.....	9
Introduction	10
Purpose	10
Scope	10
Audience	11
Terminology	11
Serviceability Description	13
Serviceability Commands	20
Backend Image Restore (svc_backend_image_restore)	20
Description.....	20
Use Cases	20
Related Commands	20
Apply Custom Log-on Banners (svc_banner)	20
Description.....	20
Use Cases	22
Example Usage.....	22
Related Commands	22
Boot Control (svc_boot_control).....	23
Description.....	23
Use Cases	23
Related Commands	23
Single SP Cache Card Control (svc_cachecard).....	23
Description.....	23
Use Cases	23
Related Commands	24
Anti-Virus Configuration (svc_cava).....	24
Description.....	24
Use Cases	24
Example Usage.....	24
Related Commands	25
Configure Backup and Recovery (svc_cbr)	25
Description.....	25
Use Cases	25
Related Commands	25

Clear Dirty Cache (svc_cdca)	25
Description.....	26
Use Cases	26
Related Commands	26
System Upgrade (svc_change_config).....	26
Description.....	27
Use Cases	27
Example Usage.....	27
Related Commands	27
Clean Solid State Disk (svc_clean_ssd).....	27
Description.....	27
Use Cases	28
Example Usage.....	28
Related Commands	29
ConnectEMC Configuration (svc_connectemc).....	29
Description.....	29
Use Cases	29
Example Usage.....	29
Related Commands	30
Upload SSL Certificates (svc_custom_cert).....	30
Description.....	30
Use Cases	30
Example Usage.....	30
Related Commands	30
Data Collection (svc_dc).....	31
Description.....	31
Use Cases	31
Example Usage.....	32
Related Commands	33
System Diagnostics (svc_diag).....	33
Description.....	33
Use Cases	33
Example Usage.....	34
Related Commands	34
Enable Internal PCIe Interface (svc_enable_internal_network)	34
Description.....	34
Use Cases	35
Related Commands	35
ESRS Setup (svc_esrs)	35
Description.....	35
Use Cases	35
Example Usage.....	36

Related Commands	36
Fips Mode (svc_fips_mode)	36
Description.....	36
Use Cases	36
Example Usage.....	37
Related Commands	37
Help (svc_help).....	37
Description.....	37
Use Cases	37
Related Commands	37
Hardware Component Tree (svc_hw_tree)	37
Description.....	37
Use Cases	37
Example Usage.....	38
Related Commands	39
Initial Configuration (svc_initial_config)	39
Description.....	39
Use Cases	39
Example Usage.....	40
Related Commands	40
Inject Troubleshooting Software Tool (svc_inject).....	40
Description.....	40
Use Cases	41
Example Usage.....	42
Related Commands	43
Interface with Encrypted Drives (svc_key_restore)	43
Description.....	43
Use Cases	44
Related Commands	45
View Locks (svc_lockd)	45
Description.....	45
Use Cases	45
Example Usage.....	45
Related Commands	45
Mount Storage (svc_mount)	45
Description.....	46
Use Cases	46
Related Commands	46
End-to-End Storage Mapping (svc_neo_map).....	46
Description.....	46
Use Cases	46
Example Usage.....	47

Related Commands	47
Create Management Interface (svc_network).....	47
Description.....	47
Use Cases	47
Related Commands	48
Network Configuration Information (svc_networkcheck).....	48
Description.....	48
Use Cases	48
Example Usage.....	49
Related Commands	49
Synchronize time (svc_ntp)	49
Description.....	49
Use Cases	49
Example Usage.....	50
Related Commands	50
Collect Performance Information (svc_perfcheck).....	50
Description.....	50
Use Cases	50
Example Usage.....	51
Related Commands	52
Pre-Upgrade Health Check (svc_puhccheck)	52
Description.....	52
Use Cases	52
Example Usage.....	52
Related Commands	52
Delete Error Logs (svc_purge_c4logs).....	52
Description.....	52
Use Cases	52
Related Commands	53
Restore VNXe OE (svc_reimage)	53
Description.....	53
Use Cases	53
Related Commands	53
Reinitialize VNXe to Factory Settings (svc_reinit)	53
Description.....	54
Use Cases	54
Related Commands	54
Service Mode Hint (svc_rescue_reason)	54
Description.....	54
Use Cases	54
Related Commands	55
Service Mode Information (svc_rescue_state)	55

Description.....	55
Use Cases	55
Example Usage.....	55
Related Commands	56
Management Stack Controls (svc_restart_service)	56
Description.....	56
Use Cases	56
Example Usage.....	57
Related Commands	57
Restart Management Stack (svc_restart_mgmt)	57
Description.....	57
Use Cases	57
Example Usage.....	57
Related Commands	57
Output UI Information (svc_save_cem_config)	57
Description.....	57
Use Cases	58
Example Usage.....	58
Related Commands	59
Enable serial port transfers (svc_serial_transfer).....	59
Description.....	59
Use Cases	59
Related Commands	60
Service User Password Configuration (svc_service_password)	60
Description.....	60
Use Cases	60
Example Usage.....	61
Related Commands	61
Service Shell (svc_service_shell).....	61
Description.....	61
Use Cases	61
Related Commands	61
Shutdown (svc_shutdown).....	61
Description.....	62
Use Cases	62
Example Usage.....	62
Related Commands	63
Output System Partition's SMART Information (svc_smartdata).....	63
Description.....	63
Use Cases	64
Example Usage.....	64
Related Commands	64

Enable Secure Shell (svc_ssh).....	64
Description.....	64
Use Cases	64
Example Usage.....	65
Related Commands	65
Run FSCK on Storage (svc_storage_integritycheck)	65
Description.....	65
Use Cases	65
Example Usage.....	65
Related Commands	66
Output Storage Information (svc_storagecheck).....	66
Description.....	66
Use Cases	66
Example Usage.....	67
Related Commands	67
Output VNXe System Status (svc_sysstatus)	67
Description.....	67
Use Cases	67
Related Commands	67
Redirect Output (svc_tcpdump).....	67
Description.....	68
Use Cases	68
Example Usage:.....	69
Related Commands	69

Executive summary

- Business case** The VNXe series is designed to be serviced by the user. Common system problems can be solved from within Unisphere on the system's Service Page. However, a problem may occur that is not diagnosable or solvable by the options found on the Service Page.
- Solution overview** A set of problem diagnostic, system configuration, and system recovery commands are installed on the system's operating environment. These commands provide an in-depth level of information and a lower level of system control than is available through Unisphere. This document describes these commands and their common use cases.
- Key results / recommendations** The Service (svc) Commands listed in this document are a subset of the operating environment's software tools for servicing a VNXe system. Users may use the UEMCLI scriptable system configuration for additional capability. UEM is not discussed in this document.

Introduction

This document describes the set of operating environment commands to diagnose and solve VNXe system problems.

Purpose

This document describes the commands available for diagnosing and solving system problems that may not be correctable through Unisphere. It also discusses common uses for the Service Commands.

Scope

This document provides a list of software tools available within the VNXe's SSH that, when combined with the proper methodology, can troubleshoot VNXe system problems. *Troubleshooting* is the process of finding a problem, diagnosing its cause, and applying its solution.

The Service Commands listed are available on VNXe systems running OE revisions 2.0.0.X through 2.3.X.X. Some changes have occurred between revisions. Enter the "svc_help" command to review commands available in a specific OE version.

Additional Service Commands (not found in this document) may have been installed on the system by authorized technical-support personnel for troubleshooting purposes. Do not run additional commands without the approval of your authorized Service Representative.

The commands provide the following high-level problem solving functions:

- **Configuration**—Set or reset the state of individual VNXe system hardware or software components.
- **Diagnostic**—Test or report the state of the system's hardware or software.
- **Operations**—Supports commands for advanced troubleshooting. These tools are intended for use by authorized Support Representatives only.
- **Recovery**—Restore the system hardware or software components to a known state.

Commands are run after logging into a VNXe Storage Processor (SP) as the Service user account. The login is through the console's Secure Shell (SSH) or a terminal attached to the platform's serial connector. Before using these commands, apply for the Service Account password, enable SSH, get any terminal hardware ready, and install supporting software applications.

VNXe SPs can run in one of two operational modes: Normal Mode or Service Mode. Some Service Commands run in both modes. Different Service Commands run in different SP modes:

- Normal Mode—some commands only run in standard operational mode.
- Service Mode—some commands only run in maintenance and troubleshooting mode.
- Both—some commands run in either mode.

Audience

VNXe storage system administrators, EMC, EMC partner field, service personnel and support personnel.

The Service Commands run on the VNXe's Linux operating environment. You need to be familiar with the Linux shell, your VNXe's installed hardware, and the VNXe's operating environment to use these commands.

Additional information that may be helpful when using these commands can be found in www.emc.com/vnxesupport.

Terminology

This paper includes the following terminology.

Table 1. Terminology

Term	Definition
Admin user	The "admin" account has the ability to manage and configure servers as well as provision the storage system. This account is the only default user that may log in to Unisphere in Normal Mode.
Backend Repository	A portion of the first four drives in the DPE are dedicated to VNXe system space. A section of this space is the backend repository, which is dedicated to maintaining known good images of VNXe software.
Boot Counters	Boot counters are a troubleshooting mechanism designed into the VNXe architecture to help pinpoint the specific cause of software or hardware events during the boot sequence. If a problem with software or hardware components is detected by the VNXe system, its corresponding Boot Counter is incremented. If a counter reaches a predefined threshold, the SP boots into Service Mode during its next boot cycle. See <code>svc_diag</code> for troubleshooting steps.
ConnectEMC	A component of the EMC internal-standard Global-Services-approved solution for transporting event data from an EMC product to EMC support. ConnectEMC simplifies and standardizes the method that EMC products utilize to transport event files—error, informational, configuration, and others—from a service workstation to EMC support systems.
Disk Processor Enclosure (DPE)	A DPE is a physical, rack-mountable enclosure that includes one or two SPs, power supplies, and at least four drives.
EMC Secure Remote Support (ESRS)	ESRS IP Solution (ESRS IP) provides a secure, IP-based, distributed support solution for command, control, and visibility into a system by an authorized remote support representative.
Normal Mode	Standard operational mode for a VNXe system A user can manage and configure servers as well as provision storage. User data is accessible while the system is in Normal Mode.

Term	Definition
Service Mode	The VNXe system's reduced operational mode. This mode is entered for maintenance and troubleshooting. In this mode, a limited interface through Unisphere or a Command Line Interface (CLI) enables problem resolution. A SP in Service Mode does not process data requests. User data is not accessible when all SPs are in Service Mode. Also, users cannot manage or provision new servers. Certain operations performed in Service Mode – such as changing the Service Account password or injecting a service tool – are not persistent across reboots and their effect will not be present in Normal Mode.
Service User Account	The Service Account has the rights to perform maintenance and troubleshooting on the VNXe system. This account is the only one that can log into the VNXe CLI (via SSH or serial connection) and the Service Page within Unisphere.
Storage Processor (SP)	A discrete, highly-available server that hosts both file and block level virtualized storage as well as management for these services. SPs are physically located within the DPE and hold the CPU, memory, onboard SSD, and Battery Backup Unit (BBU).

Serviceability Description

Commands may be categorized by:

- **Function**—which step of the troubleshooting process the command is best used in.
- **Mode**—the system’s operational mode in which the command is enabled.
- **Usage**—see the “Scope” section for a definition of the available Function and Mode categories.

In addition, the Target Audience is included in the Usage category. These categories include:

- **General Use**—No special knowledge is needed to run or understand the results.
- **Technical Service**—Advanced training is required to run the command or understand the results. Do not run these commands without the approval of your authorized Service Representative.

The following tables describe the usage of the commands by Function, Mode, and Usage.

Table 2. Service Commands by Function

Function	Name
Configuration	Apply Custom Log-on Banners (svc_banner)
	ConnectEMC Configuration (svc_connectemc)
	Upload SSL Certificates (svc_custom_cert)
	ESRS Setup (svc_esrs)
	Fips Mode (svc_fips_mode)
	Initial Configuration (svc_initial_config)
	Service User Password Configuration (svc_service_password)
Diagnostic	Anti-Virus Configuration (svc_cava)
	Data Collection (svc_dc)
	System Diagnostics (svc_diag)
	View Locks (svc_lockd)

Function	Name
	End-to-End Storage Mapping (svc_neo_map)
	Network Configuration Information (svc_networkcheck)
	Collect Performance Information (svc_perfcheck)
	Pre-Upgrade Health Check (svc_puhccheck)
	Service Mode Hint (svc_rescue_reason)
	Service Mode Information (svc_rescue_state)
	Output Storage Information (svc_storagecheck)
	Output VNXe System Status (svc_sysstatus)
	Redirect Output (svc_tcpdump)
Recovery	Backend Image Restore (svc_backend_image_restore)
	Configure Backup and Recovery (svc_cbr)
	Clear Dirty Cache (svc_cdca)
	Create Management Interface (svc_network)
	Restore VNXe OE (svc_reimage)
	Reinitialize VNXe to Factory Settings (svc_reinit)
	Enable serial port transfers (svc_serial_transfer)
	Service Shell (svc_service_shell)
	Run FSCK on Storage (svc_storage_integritycheck)
System Operations	Boot Control (svc_boot_control)
	Single SP Cache Card Control (svc_cachecard)
	System Upgrade (svc_change_config)

Function	Name
	Clean Solid State Disk (svc_clean_ssd)
	Enable Internal PCIe Interface (svc_enable_internal_network)
	Help (svc_help)
	Interface with Encrypted Drives (svc_key_restore)
	Synchronize time (svc_ntp)
	Output UI Information (svc_save_cem_config)
	Run FSCK on Storage (svc_storage_integritycheck)

Table 3. Service Commands by Mode

Mode	Name
Normal Only	Apply Custom Log-on Banners (svc_banner)
	Anti-Virus Configuration (svc_cava)
	Configure Backup and Recovery (svc_cbr)
	Clear Dirty Cache (svc_cdca)
	System Upgrade (svc_change_config)
	ConnectEMC Configuration (svc_connectemc)
	Upload SSL Certificates (svc_custom_cert)
	ESRS Setup (svc_esrs)
	Hardware Component Tree (svc_hw_tree)
	Initial Configuration (svc_initial_config)
	View Locks (svc_lockd)
	End-to-End Storage Mapping (svc_neo_map)

	Network Configuration Information (svc_networkcheck)
	Synchronize time (svc_ntp)
	Collect Performance Information (svc_perfcheck)
	Pre-Upgrade Health Check (svc_puhccheck)
	Delete Error Logs (svc_purge_c4logs)
	Restart Management Stack (svc_restart_mgmt)
	Management Stack Controls (svc_restart_service)
	Output UI Information (svc_save_cem_config)
	Run FSK on Storage (svc_storage_integritycheck)
	Service User Password Configuration (svc_service_password)
	Run FSK on Storage (svc_storage_integritycheck)
	Output Storage Information (svc_storagecheck)
Service Only	Backend Image Restore (svc_backend_image_restore)
	Boot Control (svc_boot_control)
	Enable Internal PCIe Interface (svc_enable_internal_network)
	Mount Storage (svc_mount)
	Create Management Interface (svc_network)
	Restore VNXe OE (svc_reimage)
	Reinitialize VNXe to Factory Settings (svc_reinit)
	Service Mode Hint (svc_rescue_reason)
Both	Single SP Cache Card Control (svc_cachecard)

Clean Solid State Disk (svc_clean_ssd)
Data Collection (svc_dc)
System Diagnostics (svc_diag)
Fips Mode (svc_fips_mode)
Help (svc_help)
Inject Troubleshooting Software Tool (svc_inject)
Interface with Encrypted Drives (svc_key_restore)
Service Mode Information (svc_rescue_state)
Enable serial port transfers (svc_serial_transfer)
Service Shell (svc_service_shell)
Shutdown (svc_shutdown)
Output System Partition's SMART Information (svc_smartdata)
Enable Secure Shell (svc_ssh)
Output VNXe System Status (svc_sysstatus)
Redirect Output (svc_tcpdump)

Table 4. Service Commands by Usage

Usage	Name
General Use	Apply Custom Log-on Banners (svc_banner)
	Anti-Virus Configuration (svc_cava)
	System Upgrade (svc_change_config)
	Upload SSL Certificates (svc_custom_cert)
	Data Collection (svc_dc)
	System Diagnostics (svc_diag)

Usage	Name
	Fips Mode (svc_fips_mode)
	Help (svc_help)
	View Locks (svc_lockd)
	Network Configuration Information (svc_networkcheck)
	Synchronize time (svc_ntp)
	Collect Performance Information (svc_perfcheck)
	Pre-Upgrade Health Check (svc_puhccheck)
	Service Mode Hint (svc_rescue_reason)
	Service Mode Information (svc_rescue_state)
	Shutdown (svc_shutdown)
	Enable Secure Shell (svc_ssh)
	Output Storage Information (svc_storagecheck)
Technical Service	Backend Image Restore (svc_backend_image_restore)
	Boot Control (svc_boot_control)
	Single SP Cache Card Control (svc_cachecard)
	Configure Backup and Recovery (svc_cbr)
	Clear Dirty Cache (svc_cdca)
	Clean Solid State Disk (svc_clean_ssd)
	ConnectEMC Configuration (svc_connectemc)
	Enable Internal PCIe Interface (svc_enable_internal_network)
	ESRS Setup (svc_esrs)
	Hardware Component Tree (svc_hw_tree)

Usage	Name
	Initial Configuration (svc_initial_config)
	Inject Troubleshooting Software Tool (svc_inject)
	Interface with Encrypted Drives (svc_key_restore)
	Mount Storage (svc_mount)
	End-to-End Storage Mapping (svc_neo_map)
	Create Management Interface (svc_network)
	Delete Error Logs (svc_purge_c4logs)
	Restore VNXe OE (svc_reimage)
	Reinitialize VNXe to Factory Settings (svc_reinit)
	Management Stack Controls (svc_restart_service)
	Restart Management Stack (svc_restart_mgmt)
	Output UI Information (svc_save_cem_config)
	Service Shell (svc_service_shell)
	Redirect Output (svc_tcpdump)

Serviceability Commands

This section describes the Service Commands available and provides appropriate-usage examples and use cases.

Note that many Service Commands support a *help* option. This option is run with either the "--help", "- h" or "- ?" switch (no quotes). This will list usage syntax, usage examples, and other information about the command's use. For example, to see the help option for `svc_ssh`, run: `svc_ssh --help`

Backend Image Restore (`svc_backend_image_restore`)

Overwrites the OE stored on the backend drive image repository.

Function: Recovery

Mode: Service

Usage: Technical Service

Description

This allows the Service user to overwrite the copy of the OE image stored in the image repository. When the SP's image repository gets corrupted, this service command is used to update the image repository with a known-good image. After a successful update, the `svc_reimage` service command writes this updated image to the SP's system partition.

All SPs have to be placed in Service Mode to run this command.

Use Cases

Usage: `svc_backend_image_restore [<path>]`

Related Commands

Restore VNXe OE (`svc_reimage`), Reinitialize VNXe to Factory Settings (`svc_reinit`)

Apply Custom Login Banners (`svc_banner`)

Configures the login banner type for Linux and Unisphere.

Function: Configuration

Mode: Both

Usage: General Use

Description

This allows the Normal user to specify what type of banner is displayed when logging into Linux via SSH or serial terminal connection. The `svc_banner` operations only need be run on one SP and the changes will synchronize between all SPs.

A custom banner may also be configured to display when authenticating through Unisphere as the Service user.

The login banner may be one of three different types:

- Simple—Displays a message containing:
 - VNXe system type
 - system hostname

- system software version
- Complex—Displays same information as a simple banner in addition to:
 - System serial number
 - Unisphere IP Address
- Custom—Contains any message, including support for localized banners
 - If the custom banner file is named “en_US.txt” the banner’s contents will also be displayed when authenticating with Linux

Setting any of the banner types destroys all current banner information present on the system. For example, setting a “complex” banner will destroy any custom banner files present.

Custom or localized banners can only be set in Service Mode. As the Service user, create plain-text files in /home/service whose names follow the convention of:

- Two-letter lower-case language code (ISO 639-2)
- Underscore (_)
- Two-letter upper-case country code (ISO 3166)
- “.txt”
 - Example: US English banner would be named: en_US.txt

After creating these banner files, run `svc_banner --put <list of files>` to apply them to the system.

Use Cases

Usage: `svc_banner [-q] [-s] [action] [parameter]`

Where the qualifiers are:

`-q`: Quiet mode - suppressed all output. This must be first parameter.

`-s`: Single SP mode - must be specified before action and after quiet (if applicable).

Where the actions are:

`--set-simple`: Sets system banner to default simple - destroys any custom banner installed on the system.

`--set-complex`: Sets system banner to include more system information, such as: hostname, software version, IP address. Destroys any custom banner installed on the system.

`--activate [-a]`: Activates custom login banner if already in non-volatile memory

`--put [-p]`: Puts files into CEM Customization directory & activates file as banner if it is named "en_US.txt"

Parameter:

[files]

`--delete [-d]`: Clears the login banner text from non-volatile memory and restores default system banner

`--dump`: Dumps the contents of the system banner as plain text to stdout

Example Usage

To activate banner files in US English, Belarusian and Chinese, firstly create files with localized content named:

- en_US.txt
- be_BY.txt
- zh_CN.txt

Then, to apply these files, run the following command as the Service user:

```
service@spa spa:~> svc_banner --put en_US.txt be_BY.txt zh_CN.txt
```

To revert back to the default style banner, run the following command as the Service user:

```
service@spa spa:~> svc_banner --set-simple
```

Related Commands

None.

Boot Control (svc_boot_control)

Lists or sets up the boot control blocks.

Function: System Operations

Mode: Both

Usage: Technical Service

Description

Sets, clears, or lists boot control block tallies and breakpoints before the VNXe specific software is loaded. It sets up for booting directly into the Linux operating environment for troubleshooting.

Use Cases

Usage: `svc_boot_control [<qualifiers>] <param>`

Where the qualifiers are:

`--help [-?]`: Display this message

`--set [-s]`: Request stop before c4-boot

Parameter:

[net]

`--clear [-c]`: Clear request to stop before c4-boot

Parameter:

[net]

`--list [-l]`: Lists boot_control current settings.

Related Commands

None.

Single SP Cache Card Control (svc_cachecard)

Sets up and manages the cache card on a single SP system.

Function: System Operations

Mode: Both

Usage: Technical Service

Description

Manages, configures, and upgrades the cache card on a single SP VNXe 3100. By default, the single SP VNXe 3100 comes with its cache card configured. This command gets information about the cache card and runs card functions such as rebooting and updating its firmware.

Use Cases

Usage: `svc_cachecard [options]<param>`

Options:

`--help [-h]`: display this help message

`--info [-i]`: display cache card information

`--reboot [-r]`: reboot cache card

`--upgrade [-u]`: upgrade cache card with firmware

Parameter:

[firmware]

`--init [-init]`: cache mirror init (any vaulted cache will be removed)

```
--iostat [-io]: see if I/O is going through cache card
--status [-s]: displays info on cache card status
--firmware [-f]: displays info on cache card firmware
--query [-q]: determines if cache card is enabled and seen
by the operating environment.
--configured [-c]: determines if cache card has been
configured after a re-initialization.
```

Related Commands

None.

Anti-Virus Configuration (svc_cava)

Sets up and manages the CIFS file system's anti-virus protection via the EMC® Celerra® AntiVirus Agent (CAVA).

Function: Diagnostic

Mode: Normal

Usage: General Use

Description

This command allows a knowledgeable user to perform anti-virus activities from the CLI. CAVA is an antivirus solution for clients using industry-standard CIFS (Common Internet File System) protocols in a Microsoft Windows server. CAVA uses third-party antivirus software to identify and eliminate known viruses before they infect files on the storage system.

Use Cases

```
Usage: svc_cava { <servername> | ALL }
        [ -audit | -update ]
        | [ -set accesstime={ now | none |
[[[yy]mm]dd]hh]mm[.ss] } ]
        | [ -fsscan [ <fs_name> { -list | -create [offline]
| -delete } ] ]
```

Example Usage

```
root@VNXe-spa spa:/nas/bin> svc_cava server_2
server_2 :
10 threads started.
2 Checker IP Address(es):
10.x.x.216 OFFLINE at Tue Feb 21 18:45:41 2012 (GMT-00:00)
MS-RPC over SMB, CAVA version: 4.6.8, ntStatus:
CONNECTION_DISCONNECTED
AV Engine: Network Associates
Server Name: win2008-srv-01.pedomain.local.116.245.10.in-
addr.arpa
Last time signature updated: Mon Feb 20 22:19:42 2012 (GMT-
00:00)
10.x.x.171 ONLINE at Tue Feb 21 18:45:41 2012 (GMT
00:00)
MS-RPC over SMB, CAVA version: 4.6.8, ntStatus: SUCCESS
AV Engine: Network Associates
Server Name: win-95j4fp0i27r.pedomain.local
```



```

Last time signature updated: Mon Feb 6 22:57:08 2012 (GMT-
00:00)
31 File Mask(s):
*.EXE *.COM *.DOC? *.DOT *.XL? *.MD? *.VXD *.386 *.SYS *.BIN
*.RTF *.OBD
*.DLL *.SCR *.OBT *.PP? *.POT *.OLE *.SHS *.MPP *.MPT *.XTP
*.XLB *.CMD *.OVL
*.DEV *.ZIP *.TAR *.ARJ *.ARC *.Z
No file excluded.
Share \\foo675.pedomain.local\CHECK\$.
RPC request timeout=25000 milliseconds.
RPC retry timeout=5000 milliseconds.
High water mark=200.
Low water mark=50.
Scan all virus checkers every 10 seconds.
When all virus checkers are offline:
Shutdown CIFS.
Scan on read disable.
Panic handler registered for 65 chunks.
MS-RPC User: FOO675$
MS-RPC ClientName: foo675.PEDOMAIN.LOCAL

```

Related Commands

None.

Configure Backup and Recovery (svc_cbr)

Sets up backup and recovery (CBR)

Function: Recovery

Mode: Normal

Usage: Technical Service

Description

This command may be used by the Service User to backup a system's configuration at any time. The system automatically performs a daily backup. The backup archives are available at /EMC/backend/service/CBR.

To restore a system, contact your Service Provider. Only EMC Escalation Engineers are authorized to use this recovery option.

Use Cases

usage: svc_cbr [<options>]

options:

--help[-h]: display this help message

Related Commands

None.

Clear Dirty Cache (svc_cdca)

Identifies the LUNs with dirty cache and clears them.

Function: Recovery

Mode: Normal

Usage: Technical Service

Description

This command clears the system of the dirty cache state. When all SPs go offline with acknowledged data in flight, the system may come online in a state of data inconsistency. This is known as having dirty cache. Once a system is in this state, associated user data will be unavailable until the dirty cache status is cleared, either automatically by the system or manually using this command. If the system clears the status, there is no data loss. If the status is manually cleared, there may be limited data loss.

If a system boots into this dirty cache state, authorized support representatives must first evaluate the system and then run this command to restore access to user data.

Use Cases

Usage: `svc_cdca [<qualifiers>]`

Qualifiers:

```
--help[-?]          - displays this text
--list-cdca [-l]    - list all cache dirty luns of current SP.
--clear[-c]        - clear all the cache dirty luns for
current SP, prepare any system luns for auto fsck.
--fsck-list[-f]    - re-do the auto fsck pre-processing, based
on last 'clear' processing.
--user-fs-list[-u] - generate the list of user
filesystems that should be checked, run this where nasdb is
mounted.
--perform-fsck[-p] - actually perform the fsck. Does the
same procesing as --user-fs-list, but also executes required
fscks.
--done[-d]         - clean up all crumbs related to CDCA
script. Will not be able to use --fsck-list or --user-fs-
list until the next clear.
--show-all[-s]    - show all LUN information for both SPs.
--boot-control-stop      - set system to halt boot when we
can clear CDCA
--boot-control-continue - continue a c4 boot that is
currently halted
```

Related Commands

None.

Converts a single SP to a dual SP system.

Function: System Operations

Mode: Both

Usage: General Use

System Upgrade
([svc_change_config](#))

Description

Prepares a single SP-model system for conversion into a dual model SP system. It is also used to increase the memory capacity of dual SP-based VNXe 3100 system from 4GB to 8GB.

This command is used as part of the VNXe 3100 upgrade process.

Use Cases

Usage: `svc_change_config` [<qualifiers>]

Qualifiers ::= `-h` | `-c` | `-a`

Where <qualifiers> are:

`--help[-h]`: print this message

`--commit[-c]`: valid in service mode to commit an upgrade

`--abort[-a]`: valid in service mode to abort an upgrade

`--cleanup_network` : valid in normal mode to remove network interfaces on non-existent ports

Example Usage

```
service@VNXe-spa:~> svc_change_config
```

The current system configuration is: Single SP, 4GB.

These are your options:

- 1) Convert to Dual SP, 4GB.
- 2) Convert to Dual SP, 8GB.
- 3) Quit this program.

Enter selection: 1

The following upgrade is selected: Dual SP, 4GB.

Please verify that all of your upgrade components are available before you continue. Your array will reboot if you continue, and all of your data will be unavailable until the upgrade is completed. The upgrade is a 'one way' change to your array.

Enter 'continue' to proceed: continue

Checking system state, this may take several minutes, please wait...In 30 seconds your array will reboot into Service Mode. Then you can power down your array, and perform the upgrade.

turning OFF write cache..

trying to obtain write-cache...acquired write-cache done

Related Commands

None.

Clean Solid State Disk (`svc_clean_ssd`)

Checks and reclaims capacity on the system partition's storage device.

Function: System Operations

Mode: Both

Usage: Technical Service

Description

Checks if the local or peer SP system partition's available space is above the upgrade threshold. The command will also reclaim space to increase available capacity on the storage device by removing unnecessary system files.

This command can also be used to increase available capacity on the system's partition by deleting files safely and manually.

Use Cases

```
usage : svc_clean_ssd [option] [<mount_point>]
```

Options:

-s: Force Single SP operation.

-h: Displays this message

-f: Find all large files on the system or for the mount point indicated device and generate a file system usage report.

-c: Clear capacity on the root file system or on mount point indicated device.

Example Usage

```
service@VNXe-spb:~> svc_clean_ssd -f
```

```
Now searching for files that could be deleted to prepare for upgrade on LOCAL SP...Done!
```

```
[Thu May 26 19:43:27 UTC 2011] File system Usage Report (LOCAL SP) :
```

```
=====
Local Root file system mount point      : /
Total Size of Drive                     : 5040MB
Current Total Space Used                 : 2563MB, 50.85%
usage
```

```
A total of 164MB can be saved on this system by removing files.
```

```
Total Space Used (after / cleanup) : 2399MB, 47.59% usage
```

```
Writing report file for LOCAL SP ...
```

```
Checking the LOCAL SP for deleted files still consuming space ...
```

```
No deleted files still consuming space were detected!
```

```
...
[content removed]
```

```
Removed PEER file: /var/log/messages
```

```
Cleared 365MB of space off of the PEER SP root file system!
```

```
Checking the PEER SP for deleted files still consuming space ...
```

```
No deleted files still consuming space were detected!
```

```
A log of this tool's activities can be found at the following location:
```

```
/home/service/svc_prepare_upgrade_REPORT-1306439021.txt
```

The information contained in /home/service/svc_prepare_upgrade_REPORT-1306439021.txt will be useful for EMC Engineering to enhance this service tool. Please remember to contact EMC with this information.

Related Commands

None.

ConnectEMC Configuration (svc_connectemc)

Outputs or changes information on the use of ConnectEMC configuration.

Function: Configuration

Mode: Normal

Usage: Technical Service

Description

The ConnectEMC configuration allows you to email notice of critical errors to EMC (referred to as “home”) that occur on the storage system. Information transmitted includes:

- Sending address
- Destination address
- Processing SMTP server

If ConnectEMC is enabled, EMC requests that users disable it during a maintenance window before making changes. This will prevent planned downtime from being reported as a ‘Data Unavailable’ failure to EMC.

Use Cases

Usage: `svc_connectemc [<qualifiers>] <key> <value>`

where <qualifiers> are:

`--help[-?]`: Display this message

`--get[-g]`: Get the value of the key

key:

Host

Sender

Destination

Subject

`--set[-s]`: Set the value of the key

key:

Host

Sender

Destination

Subject

value

`--purge[-p]`: Purge the phone home events

`--start`: Start connectEMC

`--stop`: Stop connectEMC

Example Usage

A user wishes to stop ConnectEMC during a maintenance window.

```
service@VNXe-spa spa:~> svc_connectemc --stop
connectemc (stop) running, process 26783
```

```
connectemc (stop) pre-stop, (main) process 26783
connectemc (stop) stopping, process 26783
connectemc (stop) killed, process 26783
connectemc (stop) post-stop
connectemc (stop) waiting
```

A user needs to change the source email address of the VNXe to meet security requirements.

```
service@VNXe-spa:~> svc_connectemc -g sender
sentry-912-spa@emc.com
service@VNXe-spa:~> svc_connectemc -s sender new@emc.com
init_complete set
ok
ok
ok
service@VNXe-spa:~> svc_connectemc -g sender
```

Related Commands

None.

Upload SSL Certificates (`svc_custom_cert`)

Installs SSL certificates.

Function: Configuration

Mode: Normal

Usage: General Use

Description

This command installs a specified SSL certificate for use with the Unisphere management interface. To use the certificate, you must already have a private key and certificate file. The private key strength needs to be at least 2048 bits and in the PEM format.

The key file must be fully pathed and named according to the following convention: `<certfilename>.pk`

The certificate file must be fully pathed and named according to the following convention: `<certfilename>.cert`

Use Cases

Usage: `svc_custom_cert [<qualifiers>] <cert file base path>`
where `<qualifiers>` are:
`--help[-h]`: Display this message

where `< cert file base path>` is Linux directory path.

Example Usage

To install with private key file: `/home/service/mykey.pk` and `cert /home/service/mykey.cert`:

```
svc_custom_cert /home/service/mykey
```

Related Commands

None.

Data Collection (svc_dc)

Generates a Data Collection bundle for technical analysis.

Function: Diagnostic

Mode: Both

Usage: General Use

Description

Data Collection is the main source of content for remote analysis of the VNXe system. There are a number of additional use cases for the *svc_dc*. These options are outlined below.

The default location for output from this command is:
/EMC/backend/service/data_collection/.

Use Cases

Usage: *svc_dc* [<options>] [<Params>]

Where options are:

--help [-h | -?]: Print this message and exit;

--lifetime [-l]: Working agents are forcedly terminated after the selected amount of seconds passed since the collection start;

Parameters:

[seconds]: Maximum seconds of execution. Default is 1800 seconds.

--current-sp [-csp] By default data collection on both SPs is run. When this option is used, data collection is run only on the current SP.

--name-prefix [-n]: Output naming and number of files to archive

Parameters:

[prefix]: Output file name prefix used for archive naming.

[keep]: Number of this type of files to be kept.

--wait-for-peer [-w]: Shows the peer process will be waited on. The default maximum wait time is 7200 seconds.

--terminate: Stops all DC instances and releases file locks

--version [-v]: Output version information.

--list-cdumps [-lcd]: Output a list of saved core dump files.

--download-core [-dc]: Compresses (tar) core-dump files and transmits it location given.

Parameters:

[/ <IP>]: Destination IP address.

[/ <directory>]: Destination directory

[/ <file>]: Core dump file name. The suffix will be omitted in case if the full processing was already performed before.

--download-core-nobins [-dcn]: Download core dump without binaries.

Parameters:

[/ <IP>]: Destination IP address.

[/ <directory>]: Destination directory

[/ <file>]: Core dump file name. The suffix will be omitted in case if the full processing was already performed before.

--download-core-abstract [-dca]: Run unpacking and GDB/crash information extraction (if not yet performed) and send only abstract information.

Parameters:

[/ <IP>]: Destination IP address.

[/ <directory>]: Destination directory

[/ <file>]: Core dump file name. The suffix will be omitted in case if the full processing was already performed before.

--download-core-processed [-dcp]: Run full core dump analysis, if was not performed before and send all available data.

Parameters:

[/ <IP>]: Destination IP address.

[/ <directory>]: Destination directory

[/ <file>]: Core dump file name. The suffix will be omitted in case if the full processing was already performed before.

--download-pre-processed [-dpp]: Run log gathering pre-processing (and download all available data. If full analysis was performed before, core abstracts are added to the resulting archive

Parameters:

[/ <IP>]: Destination IP address.

[/ <directory>]: Destination directory

[/ <file>]: Core dump file name. The suffix will be omitted in case if the full processing was already performed before.

--process-core [-pc]: Run full core dump analysis without download.

Parameters:

[name]: Name of core dump file.

Example Usage

```
service@VNXe-spb:~> svc_dc
[DC spb]: invoked from 26555 26554 26555 -bash
[DC spb]: Arguments:
[DC spb]: DCPID 26612
[DC spb]: Spawned group 26687
[DC spb]: Destination folder is:
/EMC/backend/service/data_collection
```



```

[DC spb]: SP status:
          spb  -- Normal Mode
          Peer -- Normal Mode
[DC spb]: Gathering DC information on peer
[DC spb]: Gathering DC information on spb
[DC spb]: Archiving collected data from spb
[DC spb]: Waiting for peer to finish Data Collection and
files transfer...
[DC spb]: Peer finished...
[DC spb]: moving VNXe3300_service_data_FCNCH095103208_2011-
05-26_20_21_31.tar to
/EMC/backend/service/data_collection/VNXe3300_service_data_F
CNCH095103208_2011-05-26_20_21_31.tar
[DC spb]: moving done...
DC data collected at
/EMC/backend/service/data_collection/VNXe3300_service_data_F
CNCH095103208_2011-05-26_20_21_31.tar
[DC spb]: Elapsed time: 2 minutes 19 seconds

```

Related Commands

None.

Runs and outputs troubleshooting information.

Function: Diagnostic

Mode: Both

Usage: General Use

Description

General purpose to diagnose for system troubleshooting. The command's output may include a basic set of general diagnostics, or be targeted to a specific subsystem.

The diagnostics that you can run are:

- basic—General diagnostic
- boot control—System (re)boot status
- cru extended—Current state of the hardware
- network—Frontend port configuration information including MTU size.
- resume—Provides Midplane Part and Serial Numbers.
- sas—Advanced diagnosis of the backend SAS ports.
- software—State of the system software stack
- spinfo—Detailed output of SP hardware and software information

Executing the command without arguments results in the basic running diagnostic.

Use Cases

Usage: `svc_diag [--state=<target>[,<target>]...]`

target ::= `basic` | `bootcontrol` | `cru extended` | `network` | `resume`
| `sas` | `software` | `spinfo`

Example Usage

```
service@VNXe-spa:~> svc_diag --state=network

===== Now executing network state =====
Displaying system network information:
2: eth5: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state UP qlen 1000
3: eth4: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state UP qlen 1000
4: eth3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state UP qlen 1000
5: eth2: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500
qdisc pfifo_fast master bond0 state UP qlen 1000
10: mgmt: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state UP qlen 1000
inet 10.244.244.43/24 brd 10.244.244.255 scope global mgmt
inet 10.244.244.125/24 scope global secondary mgmt:0
15: eth_int: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
qdisc noqueue state UNKNOWN

service@VNXe-spa:~> svc_diag --state=software
===== Now executing software state =====
Displaying system software component status:
583 ? 05:13:47 csx_ic_cdxic2 < DART instance
of server_2 containerized
5696 ? 00:00:00 httpd
15107 ? 00:00:07 apl_task_mgr
15408 ? 07:19:09 mgmtd < Control
Station daemon
17595 ? 04:51:16 ECOM < Management software
- see
20619 ? 00:05:38 httpd
svc_restart_mgmt
20665 ? 00:00:00 httpd
24274 ? 01:43:50 logDaemon.x < All system
logging daemon
25307 ? 17:55:04 csx_ic_ccsx < FLARE
containerized
25815 ? 05:31:15 csx_ic_ccsx_ktr
28693 ? 02:11:35 csx_ic_admin
```

Related Commands

Service Mode Information (svc_rescue_state)

**Enable Internal
PCIe Interface
(svc_enable_intern
al_network)**

Enables the internal Ethernet interfaces on the system.

Function: Systems Operations

Mode: Service

Usage: Technical Service

Description

Enables the VNXe system's internal network between SPs in Service Mode. If the PCIe interface did not automatically become available, this command manually sets up a connection by loading the kernel modules needed for enabling the Ethernet interface.

The command has no inputs. If it's successful, zero (0) is returned.

When the system's storage and administrative software experiences difficulties, it may be necessary to have an authorized support entity use this service script to manually activate an SP's inter-SP communications interface. This operation should only be performed by an approved service entity for advanced troubleshooting and recovery operations.

Use Cases

Usage: `svc_enable_internal_network`

* This script will attempt to load the appropriate kernel modules necessary for enabling the internal ethernet interfaces on the VNXe. It will also attempt to set up a `/etc/hosts` file.

* The script will only work in Service Mode and must be run as the super user.

* It takes no options, returns 0 if successful or anything else for failure.

Related Commands

None.

ESRS Setup (`svc_esrs`)

Outputs the system's EMC Secure Remote Server (ESRS) configuration.

Function: Configuration

Mode: Both

Usage: Technical Service

Description

Enables and disables EMC Secure Remote Support (ESRS) from the command line in Service Mode.

Enabling or disabling ESRS is a local change to the SP. The change does not persist after a reboot.

If no `DEVICE_IP` is present, then the VNXe system is not configured for ESRS access.

Except for the `gqext` information option (`--g/--gwext_info`), options can only run in Service Mode.

Use Cases

Usage: `svc_esrs` [`<option>`] `<param>`

Options:

`--help` [`-h`]: Display help this text

`--enable` [`-e`]: Enable and start ESRS on the local SP - Automatically downloads configuration, if required.

Parameter:

[`-f`]: Forces a reprovision

`--disable` [`-d`]: Disable ESRS

```
--status [-s]: Display ESRS state and connection status

--gwext_info [-g]: Create and output the gwext.ini file with
the required information.

--trshoot_info [-t]: Similar to -g/gwext_info, but does not
generate the ini file. Lists some additional fields that are
useful for ESRS troubleshooting
```

Example Usage

```
service@VNxe-spa:~> svc_esrs -g
PRODUCT_ID=VNxe3100
DEVICE_TLA=FCNCH082100134
DEVICE_IP=
```

Related Commands

ConnectEMC Configuration (svc_connectemc)

Fips Mode (svc_fips_mode)

Turns FIPS mode off/on.

Function: Configuration

Mode: Both

Usage: General Use

Description

Allows the user to specify whether the system's web server in Service Mode should operate in FIPS mode or not.

Operating in FIPS mode restricts the cryptographic functions of the web server to a limited, FIPS 140-2 approved. The client's web browser must be FIPS-compatible in order to connect to Unisphere on a VNxe when operating in FIPS mode. Consult your web browser's help for configuration information.

“set” operations to set FIPS mode are only allowed when all system SPs are in Normal Mode.

“get” operations are valid in any mode.

This tool should only be used if FIPS mode is needed for a particular reason (i.e. Government regulation compliance).

Use Cases

Usage:

```
svc_fips_mode [-s (ON|OFF) | -g | -h]
```

Parameters:

-s --set (ON OFF)	Sets FIPS mode ON or OFF
-g --get	Gets FIPS mode (ON or OFF)
-h --help	Prints this message

Example Usage

```
Get FIPS mode status:  
service@spa:~/>svc_fips_mode --get  
OFF
```

```
Set FIPS mode status to ON:  
service@spa:~/>svc_fips_mode --set ON
```

Related Commands

None.

Help (svc_help)

Lists the VNXe Service Commands.

Function: Systems Operation

Mode: Both

Usage: General Use

Description

Lists the Service Commands available to the service user found on the VNXe system.

In addition to the default available commands, more commands may be added though use of the Inject Troubleshooting Software Tool command. In this case the list includes those commands.

Use Cases

N/A

Related Commands

Inject Troubleshooting Software Tool (svc_inject)

Hardware Component Tree (svc_hw_tree)

Outputs the details of the system's hardware.

Function: System Operations

Mode: Normal

Usage: Technical Service

Description

Creates an XML-formatted data structure describing all details of the VNXe system's hardware.

This command can also be used to generate a data structure containing only a subset of the entire system's hardware tree by using the *include* or *exclude* parameters.

This script is primarily used by other system utilities to create an XML-formatted data structure, which is easily parable by machine and, describes the system's installed hardware components.

It is *not* recommended to run this command continuously inside of a shell script. This rate of execution may have an adverse affect on system performance.

Use Cases

Usage: [papi | svc]_hw_tree [list of arguments], where:

--help[-h]: Display this message.

--include_context: include the context information (local SP, properties).

--output_format: for each property it outputs its string values, possible applicable and modifiable types

--include_service_info: include service info for SP nodes in the tree

--exclude_service_info: exclude service info from SP nodes in the tree

--include_statuses: include status properties for all nodes in the tree

--exclude_statuses: exclude status properties for all nodes in the tree

--include_resumes: include resume properties for all nodes in the tree

--exclude_resumes: exclude resume properties for all nodes in the tree

--include_leds: include LED nodes in the tree

--exclude_leds: exclude LED nodes from the tree

--include_disks: include disk nodes in the tree

--exclude_disks: exclude disk nodes from the tree

--include_network: include network nodes in the tree

--exclude_network: exclude network nodes from the tree

--include_software: include software nodes in the tree

--exclude_software: exclude software nodes from the tree

--include_usb: include USB nodes in the tree

--exclude_usb: exclude USB nodes from the tree

--include_esrs: include information about ESRS status in the tree

--exclude_esrs: exclude information about ESRS status from the tree

Use of either *includes* or *exclude* arguments will narrow the scope of the output. You cannot use both parameters in the same command.

Example Usage

```
service@VNXe-spa:~/> svc_hw_tree -include_service_info
<?xml version="1.0" encoding="UTF-8"?>
<node type="Storage system" id="1">
```

```

<property key="Node state" value="OK"/>
<property key="Is simulation mode" value="False"/>
<children>
  <node type="DPE" id="2">
    <property key="Node state" value="OK"/>
    <property key="Status details" value=""/>
    <property key="Fault reason" value="No fault"/>
    <property key="DPE type" value="VNXe3300"/>
  ...

```

Related Commands

Hardware Component Tree (svc_hw_tree), System Diagnostics (svc_diag)

Initial Configuration (svc_initial_config)

Sets up I/O connectivity.

Function: Configuration

Mode: Normal

Usage: Technical Service

Description

Provides a CLI-based method to perform the steps of initial configuration through the Connection Utility GUI. Note that the Connection Utility is recommended whenever possible and is available at www.emc.com/vnxesupport. You need to set up:

- friendly hostname
- management IP address
- subnet mask
- default gateway

To complete this command successfully, the system must not have been configured with a management IP. This command is only used with a serial connection to the SP when the VNXe Connection Utility fails to automatically discover a system.

Use Cases

Usage: `svc_initial_config [Options]`

Options:

`--add [-a]`: Attempts to configure the VNXe with the given `friendly_name` and network parameters. Requires options described below.

`--force` Force configuration even if previous configuration detected.

Parameters:

`--network [-n]`: IP address, netmask and default gateway for management interface. Must be enclosed by quotes (") and given in that exact order."IP NMASK DEF_GW" Use "disable" instead of "IP NMASK DEF_GW" to disable ipv4 configuration

Requires `-f` to be set.

--networkv6 [-6]: IP address, netmask and default gateway for management interface MUST be enclosed by quotes ("") and given in that exact order! "<static> IP NMASK DEF_GW". Use "disable" instead of "<static> IP NMASK DEF_GW" to disable ipv6 configuration.

Requires -f to be set.

--friendly_name [-f]: New friendly_name for this SP, required

Example Usage

```
service@VNxe-spb:~> svc_initial_config -a -f NewVNxe -n "10.2.2.42 255.255.255.0 10.2.2.1"
```

```
service@VNxe-spb:~> ifconfig mgmt:0
mgmt:0      Link encap:Ethernet  HWaddr 00:60:16:36:XX:XX
            inet addr:10.2.2.42  Bcast:10.2.2.255
Mask:255.255.255.0
            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
            Interrupt:16
```

```
service@VNxe-spb:~> svc_initial_config -a -f Host_Name -n "10.244.X.X 255.255.255.0 10.244.X.1"
```

Error: this system has already been initialized with the following configuration:

```
ip=10.2.2.42 255.255.255.0 10.2.2.1
friendly_name=NewVNxe
```

Related Commands

None.

Installs an encrypted, validated diagnostic and repairs software package.

Function: Systems Operations

Mode: Both

Usage: Technical Service

Description

This script allows additional troubleshooting tools to be injected onto a VNxe system. The VNxe system hotfix procedure also uses this functionality to inject changes to the system OS as deemed necessary by Engineering.

Injectable tools are packaged in one of the following ways:

- Encrypted and security designed challenge key (key-based injection)
- Encrypted and securely packaged file (file-based injection)

The key-based option allows a user to provide authorized support representatives with a unique string to enable root access. File-based injection allows for discrete, securely-signed files to be uploaded to the VNxe SP directly.

It is recommended that software tools be injected in Normal Mode. Generally, tools injected while an SP is in Service Mode will not persist and therefore will not be available when the system returns to Normal Mode. However, in some cases the key-

Inject
Troubleshooting
Software Tool
(svc_inject)

based injection mechanism may allow for injected service tools to be persisted between Normal and Service operational modes.

Since software version 2.1.x, when performing a service tool injection a Dual-SP, VNXe will attempt to inject the service tool on both SPs, regardless of their operational mode.

Note: Once a service tool package file is used for injection, it is removed from the VNXe system. Injected tools also have an expiration date. After that date, they may not be run. The tool must be re-injected to re-enable it when it has passed its expiration date.

Use Cases

```
Usage: svc_inject [<options>] [<params>]
[<AdditionalParams>]
```

options:

--help [-?]: Displays this text

--key [-k]: Perform key-based injection- see more info below

Parameters:

[serial]: Serial key

--hotfix [-h]: Install or erase a hotfix

Parameters:

-- Install [-I [package]]: Install

-- Erase [-e [package]]: Erase

--tool [-t]: Install or erase a service tool

Parameters:

-- Install [-I [package]]: Install

-- Erase [-e [package]]: Erase

--list [-l]: List currently-installed hotfixes, service tools or both hotfixes and service tools.

Parameters:

-- hotfixes[-h]: list hotfixes

-- servicetools [-t]: list service tools

-- all [-a]: list all

--repository [-r]: Add, delete, list repositories, or dump package list

Parameters:

-- add [-a]: add repository

```
--delete [-d]: delete repository
--list [-l]: list all
--package [-p [PACKAGE]]: Output package list
--package [-p [PACKAGE]]: Output detailed info
about PACKAGE
```

Additional Parameters:

```
--local [-s]: Run only local SP, do not attempt to
inject on peer SP

--quietmode [-q]: Suppresses extraneous output, useful
for scripts
```

Example Usage

Gaining root privileges with the *svc_service_shell -k* key-based option:

```
service@VNXe-spa:~> svc_inject -k serial
INFO: serial-based key authentication requested.
System Serial Number: FNM00102800281
Enter response (or CTRL-C to quit): SQRZN-ZFWHK4-LPI3T-
T2PI5-111511
INFO: Successfully validated response.
INFO: Enabling tool ...
INFO: Successfully enabled svc_service_shell
service@VNXe-spa:~> svc_service_shell
INFO: Successfully enabled svc_service_shell
The svc_service_shell service tool will expire in:
13 day(s), 4 hour(s), 28 minute(s) and 56 second(s).
```

```
--- Start of service shell session ---
```

```
*** WARNING *** VNXe service shell activated! *** WARNING
***
root@VNXe-spa:~>
```

Note, that the response can only be provided by an authorized support representative.

Gaining root privileges with the *svc_service_shell* file-based injection method:

```
service@VNXe-spa:/tmp> ls | grep svc_
svc_service_shell-service-
tool.0.1.114.0.VNXe_MR1_expires_2011_05_24.tgz.bin.gpg.sig
service@VNXe-spa:/tmp> svc_inject -l -t
```

No service tools appear to be injected.

```
service@VNXe-spa:/tmp> svc_inject -t -i svc_service_shell-
service-
tool.0.1.114.0.VNXe_MR1_expires_2011_05_24.tgz.bin.gpg.sig
```

```

INFO: Performing dual-SP tool injection.
INFO: Verifying peer connectivity ...
Installing package svc_service_shell-service-tool ...
INFO: this tool file expires on: Tue May 24 13:27:52 UTC
2011
Running integrity check on image...
Successfully installed svc_service_shell-service-tool
Installing package svc_service_shell-service-tool ...
INFO: this tool file expires on: Tue May 24 13:27:52 UTC
2011
Running integrity check on image...
Successfully installed svc_service_shell-service-tool
service@VNxe-spa:/tmp> svc_inject -l -t
Service tools currently injected:

```

```

Tool Name Expiration Date
=====
=====
svc_service_shell          Thu Jun  2 13:15:48 UTC 2011
=====
=====

```

```

service@VNxe-spa:/tmp> sudo svc_service_shell
The svc_service_shell service tool will expire in:
13 day(s), 16 hour(s), 59 minute(s) and 44 second(s).

```

Related Commands

None.

Interface with Encrypted Drives (svc_key_restore)

Interface with encrypted drives.
 Function: System Operations
 Mode: Both
 Usage: Technical Service

Description

svc_key_restore provides an interface for performing “restore” and “revert” operations on encrypted data disks.

A restore operation will attempt to restore the Authentication Key specified by the user on the command line. It will also prompt for the password that was used to encrypt the Authentication key when the Key was backed up. Furthermore, a restore operation is only available when all system SPs are in Service Mode. The command only needs to be issued on one SP. This command requires that you have a backup key file and provide its path on the command line.

A revert operation will revert an encrypted drive back to its factory defaults. Revert operations are only allowed with both SPs in Normal Mode. The data on the disk will be securely and permanently erased and authentication on the drive will be disabled. This command requires that the CRU identifier and PSID of the disk are given as arguments. The CRU identifier is the drive's CRU ID made up of SAS port bus, enclosure and slot. PSID is an alphanumeric string that is printed on the label of an encrypted drive. This value can be manually read and entered or could be read using a bar code scanner.

The string "--sed" must be passed as the first command line argument in order to issue revert or restore commands.

Example usage syntax for revert and restore operations is provided below.

Use Cases

Usage: `svc_key_restore` [<qualifiers>]

where the qualifiers are:

```
-?|--help           : Display this message
-s|--sed            : Perform operations on a SED
Drive Key.
-r|--restore <path/name>: restore the SED Drive Key
using named file
-v|--revert <cru_identifier> <PSID>
                    : Reverts a SED drive to factory
default state.
--validate          : Check the current SED key, to
see if it appears to be valid or corrupt
```

This service script interfaces with specified drives and provides the following functionality:

Restore:

This option will attempt to restore the Authentication Key specified by the "--restore" qualifier. It prompts for password.

Please use the same password used to encrypt the authentication key during backup.

This option is only available when all system SPs are in Service Mode and only needs to be issued on one SP.

Example:

```
> svc_key_restore --sed --restore /home/service/backup.key
```

Revert:

This procedure will revert a drive to factory defaults. The data on the drive will be crypto erased and authentication on the drive will be disabled.

CRU identifier is the drive's CRU ID made up of SAS port bus, enclosure and slot.

PSID is an alphanumeric string that is printed on the label of a SED drive. This value could be manually read or could be read using a bar code scanner.

This function can only be run in Normal Mode.

Example:

```
> svc_key_restore --sed -revert <bus_encl_slot> <PSID>
> svc_key_restore --sed -revert 0_0_10
ABCD1E23FG4H5IJKL7MN9OPQR1STUV23
```

View Locks (svc_lockd)

Related Commands

None.

Views information of NFS locks.

Function: Diagnostic

Mode: Normal

Usage: General Use

Description

Allows the Service User the ability to view information about NFS locks which are currently held for provisioned VNXe storage. It will only run in Normal Mode when the system storage software is active.

The user must specify which storage server to work with on the command line. The tool can list all active locks using the “list” command. Statistics can be listed using the “stat” command; in addition, a stat reset can be issued by specifying “reset” after the “stat” command on the command line. To view more detailed information about a particular lock, the “info” command can be used. To remove a known stale NFS lock, use the “remove” command. “remove” and “info” require a lock be specified on the command line.

Use Cases

Usage : `svc_lockd server_[2|3] [command]`

Commands:

<code>h, help</code>	Prints this message
<code>list</code>	lists current locks info
<code>stat [reset]</code>	shows statistics about current locks
<code>info</code>	shows detailed information about specified lock
<code>remove</code>	removes specified lock

Example Usage

Example: Get locks info for server_2
> `svc_lockd server_2 info`

Example: Perform a stat reset for server_3
> `svc_lockd server_3 stat reset`

Related Commands

None.

Mount Storage (svc_mount)

Mounts system partitions in Service Mode.

Function: System Operations

Mode: Service

Usage: Technical Service

Description

Mount system partitions when the VNXe is in Service Mode. This command is automatically run by the SP when entering Service Mode. It can also be run manually by the service user to mount the system partitions. If the system startup has a failure in the management software, partitions used in system troubleshooting may fail to mount. This command may be used to attempt a manual force mount of those partitions.

The mountable system partitions include: SSD, Backend, and Cores. The SSD partition contains the VNXe's Linux installation and other storage/ management software required for system operation. The Backend partition stores persistent configuration information and service data. The Cores partition is where system core dumps are stored for triage and diagnostic purposes.

This command will attempt to mount the SSD, backend mirror and Cores partitions at `/mnt/ssdroot`, `/mnt/backend`, and `/mnt/cores` respectively with read-only privileges. If `svc_mount` is invoked manually, the "-w" qualifier will attempt to mount partitions with write privileges if absolutely necessary for troubleshooting or recovery operations.

Use Cases

Usage: `svc_mount` [`<qualifiers>`]

where the qualifiers are:

- `--help [-?]`: Display this message
- `--write-mode [-w]`: Mount with read/write permissions
- `--ssd-only [-s]`: Only mount SSD.
- `--cores-only [-c]`: Only mount cores
- `--backend-only [-b]`: Only mount the backend
- `--unmount [-u]`: Unmount `ssd`, `cores`, and `backend` partitions

Related Commands

Boot Control (`svc_boot_control`), Create Management Interface (`svc_network`)

End-to-End Storage Mapping (`svc_neo_map`)

Maps internal storage from file system to disk as well as from disk to file system.

Function: Diagnostic

Mode: Normal

Usage: Technical Service

Description

Lists all pools, WWNs, status of the storage and disks.

Use Cases

Usage: `svc_neo_map` [`<qualifiers>`] `<param>`

where `<qualifiers>` are:

- `--help[-?]`: Display this text.
- `--lun [-l]`: Display mappings of a specific LUN

```

Parameter:
  <num>:
--filesystem [-fs=]: Display mappings of a specific File
System
Parameter:
  <name>: File system name

```

Example Usage

```

service@VNXe-spa:~> svc_neo_map --lun=1
#### Neo Map Utility.  Get map based on FLARE LUN 1.

# Show LUN (1), WWN, Raid Group WWN, and status.
          RGoffset      bindOffset  physCapacity
actualUserCap XLU          bindState  Dflt Owner  Current
Owner  % zeroed % Rebuilt % Expanded
-----
LUN 0001          wwn =
60:06:01:60:78:56:34:12:f1:3c:d5:4d:1c:d2:13:75
          group wwn =
06:00:00:00:05:00:00:00:00:00:00:00:00:00:00:64
          0x000000000000 0x000000000000 0x00001420e000
0x000011400f00    1 (3)Assigned          SP-A(0)      YES
(1)      N/A      N/A      N/A

# Show RG info for (
06:00:00:00:05:00:00:00:00:00:00:00:00:00:00:64)
Group Number    0:
          wwn =
06:00:00:00:05:00:00:00:00:00:00:00:00:00:00:64

[output continues]

```

Related Commands

System Diagnostics (svc_diag)

Configures the remote access management interface for troubleshooting or recovery.

Function: Diagnostic

Mode: Service (Revision 2.2.x and higher: Both)

Usage: Technical Service

Description

Allows the creation of a non-persistent management configuration for SSH access when no management IP has been configured or if the system software is unable to automatically configure the interface.

Use Cases

Usage: `svc_network <-? | -d> [parameters] <qual>`

Optional operators are:

`--help [-?]`: Display this message

`--debug [-d]`: Turns on extra debugging messages

Create
Management
Interface
(svc_network)

Required parameters are:

-i: Ethernet interface to set up

Qual:

<interface>

-a: IPv4 address the given interface will be assigned

Qual:

<IP address>:Assigned IP address

-n: Network mask for the connection

Qual:

<netmask>:Assigned netmask

-g: Default route (gateway) for the connection

Qual:

< gateway >:Assigned gateway.

Related Commands

Boot Control (svc_boot_control), Mount Storage (svc_mount)

Network Configuration Information (svc_networkcheck)

Collects network information and performs diagnostics using Storage Server and Linux network interfaces.

Function: Diagnostic

Mode: Both

Usage: General Use

Description

Collects network information about one or all SPs. It allows the service user to run common network troubleshooting utilities. These utilities include *netstat*, *tracert*, *ping*, and *ethtool*. It provides information relating to the management IP connectivity. Besides, it performs specialized network checks using the storage server network interfaces, and the Linux management interface.

With the exception of the "-m, --management" command which can be run from Service Mode, this command can only be run in Normal Mode on the primary SP. The output of the command is logged to `/home/service/svc_networkcheck.log`.

Use Cases

Usage: `svc_networkcheck [command] <parameters>`

where the required commands are:

--help [-h]: Display this message

--info [-i]: Perform network checks such as TCP stats, NAS routing tables, and the underlying network configuration

--replication [-r]: Perform replication checks

--netstat [-n]: Perform standard netstat

--tracert [-t]: Perform tracert to IP


```

--ping [-p]: Perform a ping of IP address or hostname from
all servers
    Parameter
        [ip/hostname]: IP address
--ethtool [-e]: Display information about the mgmt port
(default) using ethtool. [--peer][all|<portname>]
    Optional Args:
        A specific port name or "all" can be specified to run
        ethtool on other ports
        "--peer" can specified before the ports to run the
        command on the peer
--management [-m]: [--routing|--dhcpclient|--dns|--all]
: Display the system's management IP information
    Optional Args:
        "--routing" displays additional information about
        management routing
        "--dhcpclient" displays additional information
        regarding DHCP
        "--dns" displays additional DNS query information for
        the mgmt IP
        "--all" displays all options

```

Example Usage

```

service@VNXe-spa:~> svc_networkcheck -e all
===== [spa][Thu May 19 22:50:38 UTC 2011]
Beginning Run =====

```

```

Settings for eth2:
    Supported ports: [ TP ]
    Supported link modes:   10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Full
    Supports auto-negotiation: Yes
    Advertised link modes:  10baseT/Half 10baseT/Full

```

[output continues]

Related Commands

Collect Performance Information (svc_perfcheck), Data Collection (svc_dc)

Synchronize time (svc_ntp)

Synchronizes time with the NTP server.

Function: System Operations

Mode: Normal

Usage: General Use

Description

Allows the Service User to synchronize the VNXe's time with the NTP server. It will only run in Normal Mode. It can be only used on the primary SP with a management IP configured. In addition, this tool can list information about the configured NTP servers and their status.

Use Cases

Usage: svc_ntp [OPTION]

Allows users to force an NTP time sync.

Options:

<code>-i, --info</code>	Display NTP information
<code>-s, --sync</code>	restart the NTP server and force the SP to update the time
<code>-h, --help</code>	Display help and exit

Example Usage

Example: Synchronize NTP time

```
> svc_ntp -s
```

Example: List the status about NTP servers

```
> svc_ntp -i
```

Related Commands

None.

Collect
Performance
Information
(`svc_perfcheck`)

Outputs performance metrics.

Function: Diagnostic

Mode: Normal

Usage: General Use

Description

Outputs frontend and backend setup and performance metrics. This script is designed to extract information that may be useful for performance troubleshooting. Similar content is available in Data Collects bundles – see `svc_dc`.

Some performance statistics that users can monitor for are visible through the options on `svc_perfcheck`.

Use `--sar` option to produce output similar to the Linux `sar` command for the SP. This output includes CPU usage distribution.

The `--ktrace` option provides low-level information of every IO operation. Its use is intended for storage experts only.

The `--getconfig` option provides significant information about frontend configuration. It outputs to `/home/service/` in the format `report.neo_getconfig.<system name>-<date>.txt`.

The command only runs on the primary SP. It will return an error if run on the secondary SP.

Use Cases

Usage: `svc_perfcheck <qualifiers> [<value>]`

Where `<qualifiers>` are:

`--help [-h|-?]`: Display this message

--sar [-s]: the Linux *sar* command for the SP

Value:

- seconds [-i]: seconds to run *sadc*
- number [-n]: number of times to run(max 1000)

--ktrace [-k] : Run *Ktrace* utility

Value:

- tracefile name [-i]: output file
(tracefilename.trc)
- number [-n]: number of seconds to run(max 300)

--getconfig [-g]: Frontend port setup and metrics.

Example Usage

```
service@VNXe-spb:~> svc_perfcheck -s -i 10 -n 15
Running sadc on SPA 15 times at 10 seconds interval.....
Peer is up, running sadc on SPB 15 times at 10 seconds
interval.....
Waiting for 155 seconds for sadc to finish....
sadc finished on SPA
sar finished on SPA
Finished sadc on SPB
sar finished on SPB
sar output is located at /EMC/backend/perf_stats/sar/
```

```
service@VNXe-spb:~> svc_perfcheck --ktrace -i output.trc -n
5
Running ktrace (rba) on SPA for 5 seconds.....
Peer is up, running ktrace (rba) on SPB for 5 seconds.....
sleeping for 7 seconds to let rba finish its job.....
:tcd:0001
:lun:0005
:fru:000d
:tcd:0001
:lun:0005
:fru:000d
flushing rba contents to spa_output.trc ...
closing spa_output.trc ....
spa_output.trc closed
flushing rba contents to spb_output.trc ...
closing spb_output.trc ....
spb_output.trc closed
ktrace
100% 107      0.1KB/s   00:00
ktrace file(s) saved at /EMC/backend/perf_stats/ktrace/
```

Related Commands

Collect Performance Information (svc_perfcheck)

Pre-Upgrade Health Check (svc_puhccheck)

Runs a Pre-upgrade health check and outputs detailed information.

Function: Diagnostic

Mode: Normal

Usage: General use

Description

Lists all commands and output from the Pre-Upgrade Health Check. It also allows for running the Linux `dig` utility. This utility returns records from a specified Domain Name Server (DNS).

Use Cases

```
Usage: svc_puhccheck [command] <parameters>
where the required commands are:
--help [-h]: Display this message
--check [-c]: Perform various health checks
--dig [-d]:: Perform a dig of named server.
Parameter
[server_name]: Server to run on.
```

Example Usage

```
service@VNXe-spa:~> svc_puhccheck -c

===== [Thu May 19 23:13:01 UTC 2011]
Beginning Run =====

===== Now running /EMC/Platform/bin/sptool -s ALL
... =====
dpe:                OK
temp:               24
spa:                OK
dimmm0:             OK
dimmm1:             OK
```

[output continues for 455 lines]

Related Commands

System Diagnostics (svc_diag)

Delete Error Logs (svc_purge_c4logs)

Deletes error logs from the system log database.

Function: System Operations

Mode: Normal

Usage: Technical Service

Description

This command purges the system logs from log database. These system logs are needed for root cause analysis of failure events. This script is intended for use only by authorized technical representatives when requested by Engineering.

Use Cases

```
Usage: svc_purge_c4logs [<qualifiers>] <param>
```

where <qualifiers> are:

--help[-?]: Display this message

--num[-n]: Maximum log records to keep

Parameter:

<num>: Logs to keep.

Related Commands

None.

Restore VNXe OE (svc_reimage)

Overwrites the SP's System Partition with a known-good software image while retaining the system's configuration and all customer data.

Function: Recovery

Mode: Service

Usage: Technical Service

Description

Overwrites the SP's system partition with a known good image stored on the backend system device while maintaining the persistent configuration information, such as hostname, host registration, and user data.

Using this command is a valid troubleshooting step when an SP is in Service Mode, because the SYSTEM Boot Counter has reached its threshold. It is also used automatically by the system software after a SP has the onboard System Drive replaced or if the SP detects that the system partition is corrupt or has too little space left for normal operation.

Use Cases

Usage: `svc_reimage` [<qualifiers>]

where the qualifiers are:

--help [-h]: Prints this text

--reboot [-r]: Reboot after reimage

--powerdown [-p]: Powerdown after reimage

--force [-f]: Skip all prompts

The script does no shutdown/reboot actions by default. Note that the SP must be rebooted before a reimage operation will occur. In this case, the Boot Counters must be cleared using the `svc_rescue_state` service tool prior to rebooting the system else it will reboot back into Service Mode and the reimage operation will not occur.

Related Commands

Service Mode Information (`svc_rescue_state`)

Reinitialize VNXe to Factory Settings (svc_reinit)

Returns VNXe system to its factory-delivered state.

Function: Recovery

Mode: Service

Usage: Technical Service

Description

Reinitializes a VNXe system to remedy configuration problems or to prepare a system for decommission.

The onboard SSD that contains the VNXe system's OE is overwritten with the image currently contained in the backend image repository. All persistent configurations on the system, such as hostname, storage configuration, and host registration, are permanently overwritten. All storage provisioning information and user data is lost.

This command can only be used when all installed SP(s) have been placed into the Service Mode.

This command should be considered a last resort troubleshooting solution. It is recommended to try a SP reimage (*svc_reimage*) to try to correct the problem before using this command. It's also recommended that users obtain a Data Collection and consult with their support provider before making the decision to run this command.

Use Cases

Usage: `svc_reinit` [<qualifiers>]

where the qualifiers are:

`--help` [-h]: Display this message

`--force` [-f]: Skip all prompts

Related Commands

Backend Image Restore (`svc_backend_image_restore`), Restore VNXe OE (`svc_reimage`)

Service Mode Hint (`svc_rescue_reason`)

Specifies the reason why the SP has entered Service Mode

Function: Diagnostic

Mode: Service

Usage: Technical Service

Description

The rescue reason signifies the specific reason an SP is in Service Mode. The rescue reason is a hexadecimal-formatted code. It is used mainly by the system's utilities.

The command will return a value of "0x0000" if the system has no rescue reason set, or the rescue reason has it been cleared before it was run.

In most cases, `svc_diag` should be used to view the system's rescue reason. That command contains the rescue reason code, the rescue reason's definition, and additional information about the current system state.

Use Cases

Usage: `svc_rescue_reason` [<qualifiers> | none]

Where the qualifiers are:

`--help` [-h]: Display this message

Service Mode Information (svc_rescue_state)

Related Commands

System Diagnostics (svc_diag)

Outputs or changes the boot counters controlling the system's boot into Service Mode.

Function: Diagnostic

Mode: Both

Usage: General Use

Description

Views, sets, or clears the boot counters. This command is used in the recommended remote shutdown procedures. It can also be used to determine which system component is responsible for the boot into Service Mode.

The List option details why the logged-on SP has booted into Service Mode. See `svc_diag` for further content.

The Set option manually forces the SP to boot into Service Mode the next time it is rebooted.

The Clear option initializes all the Boot Counters, and instructs the SP to attempt to boot into Normal Mode on the next reboot. Note that errors or faults may still cause the system to boot back into Service Mode. Address the reason the system re-booted into Service Mode before proceeding with clearing the counters.

See the `svc_shutdown` command for the system reboot procedure.

Use Cases

Usage: `svc_rescue_state [<qualifiers>]`

where <qualifiers are:

- `--help[-?]`: display this message
- `--list[-l]`: list all boot counters
- `--set[-s]`: set the Service-Mode boot counter
- `--clear[-c]`: clear all the boot counters

Example Usage

```
service@VNXe-spa:~> svc_rescue_state -s
1
service@VNXe-spa:~> svc_rescue_state -l | egrep
"action|USER"
      comp      code      count  max      action
      USER      0x09      1      1      0x05 (RESCUE
MODE REQUESTED)
service@VNXe-spa:~> svc_rescue_state -c
      Resetting boot control data to initial defaults...
service@VNXe-spa:~> svc_rescue_state -l | egrep
"action|USER"
      comp      code      count  max      action
```

USER 0x09 0 1 0x05 (RESCUE)
MODE REQUESTED)

Related Commands

System Diagnostics (svc_diag), Shutdown (svc_shutdown)

Management Stack Controls (svc_restart_service)

Restarts system management software.

Function: System Operations

Mode: Normal

Usage: Technical Service

Description

This command replaces `svc_restart_mgmt`. The new command has extended features.

A user can run these commands to initialize the system software for troubleshooting purposes, by exiting the process and then restarting it on the local SP.

Restarting portions of the system software interrupts their function.

The features which may be restarted are:

- **MGMT**—Launches Unisphere when an SP is in Normal Mode. MGMT is also responsible for snapshot schedules and expansion of backend storage. For example, when auto-file system extension runs out of local pooled resources. Restarting of the management software can disrupt management activities such as provisioning storage, configuration of networking, and other activities. Use `svc_restart_service MGMT` only after confirming that no other users are managing the system. The service user can run the Linux command `last` to see if other users are logged into the system).
- **NAS_A**—Controls SPA file server virtualization software. Restarting **NAS_A** will result in 'Date Unavailable' for all servers and shares accessed through this service.
- **NAS_B**—Controls SPB file server virtualization software. Restarting **NAS_B** results in 'Date Unavailable' for all servers and shares accessed through this service.
- **NASDB**—Controls the services used for NAS management activities from failover to configuration. Data is still accessible during this reboot activity.

Use Cases

Usage: `svc_restart_service <qualifiers>`

where the `qualifiers` are:

`-MGMT`: Restart the MGMT stack

`NAS_A`: Restart the NAS_A stack

`NAS_B`: Restart the NAS_B stack

`NASDB`: Restart the NASDB stack

This script performs a restart of the specified stack

Example Usage

```
service@VNXe-spa:/tmp> svc_restart_service
```

Related Commands

Restart Management Stack (svc_restart_mgmt)

Restart Management Stack (svc_restart_mgmt)

Restarts the ECOM management software.

Function: System Operations

Mode: Normal

Usage: Technical Service

Description

ECOM controls the ability to use the Unisphere GUI when an SP is in Normal Mode. It also is responsible for snapshot schedules and expansion of backend storage. Expansions of backend storage occur when auto-file system extension runs out of local pooled resources.

Restarting ECOM can disrupt management activities such as provisioning storage, configuration of networking, and other activities. It is recommended to only use this command after confirming no other users are managing the system. The Service user can run the Linux command `last` to see if other users are logged to the system.

Note: This command was replaced by `svc_restart_service MGMT`.

Use Cases

Usage: `svc_restart_mgmt`

Example Usage

```
service@VNXe-spa:/tmp> svc_restart_mgmt
Found ECOM on spa
```

Related Commands

System Diagnostics (svc_diag)

Output UI Information (svc_save_cem_config)

Outputs a record of the system configuration.

Function: System Operations

Mode: Normal

Usage: Technical Service

Description

Produces a file that contains details of the system configuration. The file can be created in either a text or XML format. By default, the data captured in the file does not include private data like IP addresses, user comments and user defined names. This private data can be included by specifying the `"-showPrivateData"` option. The data includes:

- System identification
- Hardware inventory
- Software inventory
- License and service contract data

- Application data
- Network data
- Storage data

The command writes the output to
/EMC/backend/service/configuration_history.

Additional SSH commands or programs may be needed to parse the file for specific information.

Use Cases

Usage : `svc_save_cem_config[-xml [-showPrivateData]]`

Options:

No options	Capture configuration in text format.
-xml	Capture configuration in XML format.
-showPrivateData	By default, sensitive information is excluded from the XML capture. This includes items like IP addresses. When this option is set, all data is returned.
-h, --help	Display help and exit

Example Usage

Running the command for reviewing all parts and part numbers:

```
root@egreen-1131-SPA spa:~> svc_save_cem_config -xml
```

```
/EMC/backend/service/configuration_history/20120508_112204_FNM00120600554_EMU-UEM-Telemetry.xml
```

With -xml option:

```
cat 20120508_112204_FNM00120600554_EMU-UEM-Telemetry.xml
```

```
<?xml version="1.0" encoding="UTF-8"?>
<SAN:SANConfigurationFiles
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:FILEMETADATA="http://emc.com/uem/schemas/Common_CLARiion_Type_schema" xmlns:SAN="http://emc.com/uem/schemas/Common_CLARiion_SAN_schema" xsi:schemaLocation="http://emc.com/uem/schemas/Common_CLARiion_SAN_schema
Common_CLARiion_SAN_schema.xsd"
xmlns:ARRAY_REG="http://emc.com/uem/schemas/ArrayRegistrationSchema" xmlns:UEM_VNXE="http://emc.com/uem/schemas/EMU_UEM_Telemetry_VNXe_Schema">
<FILEMETADATA:FileMetaData type="Category">
<FILEMETADATA:FileVersion
type="Property">03.00.00</FILEMETADATA:FileVersion>
<FILEMETADATA:FileCreator
type="Property">arrayconfig.pl</FILEMETADATA:FileCreator>
```

Without -xml option:

```

----
System Information:
----

----
CEMSYSTEMS
-----

----CEMSYSTEM---
ExternalModelNumber:VNXe3150
Status:ok
CaptureDurationS:2012-05-06 06:22:19
SystemName:egreen-1131
ManagementSubnetMask:255.255.255.0
Model:NEO B JR 25 DUAL DPE W 4 CORE 8G
ManagementIPv6ConfigType:disabled

```

[Output continues]

Related Commands

System Diagnostics (svc_diag), Output VNXe System Status (svc_sysstatus)

Enable serial port transfers (svc_serial_transfer)

Enables serial port for small file transfer operations.

Function: Recovery

Mode: Both

Usage: Technical Service

Description

Allows the user to send or receive files over the SP's serial connection using a software terminal application that can make connections using a modem, or a null modem cable which can emulate modem communication.

Terminal applications that support the Zmodem protocol can be used. Some of those include:

- Microsoft HyperTerminal™
- Linux sz
- PuTTY
- Telnet (with Zmodem support)

Use Cases

Note: For a step-by-step procedure of how to use this functionality, use the -i option.

Usage of the command: `svc_serial_transfer <option>`

Where options are:

```

--help [-h]: Displays this text
--instructions [-i]: Prints instructions on sending a file
using Zmodem protocol.
--receive [-r]: Receives a file over serial
--send [-s]: Sends a file over serial.

```

Related Commands

None.

**Service User
Password
Configuration
(svc_service_pass
word)**

Tests or sets the Service user password.

Function: Configuration

Mode: Normal

Usage: Technical Service

Description

Allows the user to check if the Service user password is set to its default, is valid, or change it.

The default Service user password is “service” (no quotes). When this command shows the default is in use, it is recommended that a new password be set. The `isdefault` mode switch returns a zero (0), if the default is in use. The `reset` option of this command sets the service password back to the default value.

In order for a password to be considered compliant with VNXe security policies and accepted as valid it must adhere to the following guidelines:

- 8-40 characters in length
- At least 1 uppercase character
- At least 1 lower case character
- At least 1 digit
- At least one special character from the following set:
 - !, @ # \$ % ^ * _ ~ ?
- No disallowed characters from the following set:
 - & ' space tab
- Password must be unique from the previous three Service Mode passwords.

An example of a valid password is: “m0de_S3rvice”.

Changes made to the service password using this tool while in Service Mode are considered non-persistent - the service user will have the same password it had before using the command to change it when the system returns to Normal Mode.

Use Cases

Usage: `svc_service_password <qualifiers> [<Password>]`

where <qualifiers> are:

`--help[-?]`: Display this message

`--set[-s]`: Set password with either supplied on command line or prompt.

 Password:New password.

`--reset[-r]`: Reset the default password

`--validate[-v]`: Validate the supplied password

 Password:New password.

--isdefault[-d]: Determine if password is factory default

Example Usage

```
service@VNXe-spa:~> svc_service_password -d
no
service@VNXe-spa:~> svc_service_password -v
Password: <User entered correct current service password>
ok
service@VNXe-spa:~> svc_service_password -r
service@VNXe-spa:~> svc_service_password -d
yes
service@VNXe-spa:~> svc_service_password -s
New Password: <User entered one of last 3 passwords used>
Error: password is one of the last 3 previously-used
passwords.
service@VNXe-spa:~> svc_service_password -s
New Password:
```

Related Commands

None.

Service Shell (svc_service_shell)

Obtains super access to VNXe's operating system.

Function: Recovery

Mode: Both

Usage: Technical Service

Description

Allows an authorized support entity to gain super user access to the VNXe's operating system in both Normal and Service modes.

This tool must firstly be enabled by the support entity before it can be used. Once enabled, the tool will remain functional for 3 days. After 3 days it will automatically disable itself.

Use Cases

None.

Related Commands

None.

Shutdown (svc_shutdown)

Performs a safe reboot or shutdown of a SP, and shutdown of the system.

Function: System Operations

Mode: Both

Usage: General Use

Description

This service tool is used to perform a safe reboot or power down (halt) of a VNXe SP, and shutdown of the system.

SP reboot and shutdown can be performed in both service and Normal Mode. System shutdown can be performed only when at least one of the SPs is in Normal Mode.

When initiated, the reboot operation attempts to gracefully deactivate the active storage software and all running Linux processes before performing a warm reboot of the SP. The SP will then automatically run through its boot processes and – assuming it is in a healthy state and no boot counters are set – come back online. The halt operation is similar to the reboot in that it performs the same graceful bring down of the system software but it does not reboot the SP. Instead, the SP is left in a powered-on state with no software running. A physical power -down of the SP is required to bring it back online. The halt option is available on VNXe system software revisions 2.2.x and later.

An optional Reboot Reason code may be added to the shutdown command for logging the need for the voluntary reboot. Reason codes are user definable. Reason codes are logged to the `/EMC/C4Core/log/start_c4.log` file.

Note: Systems rebooted when in Service Mode will return to Service Mode if Boot Counters are not cleared. See `svc_rescue_state` for more information on Boot Counter management.

The system-halt operation halts all SPs in the storage system. The system is left in a powered-on state with no software running. A physical power-cycle of the system is required to bring the system back online. The system-halt operation is available on VNXe system software revisions 2.2.0 and later.

Use Cases

Usage: `svc_shutdown` [`<qualifier>`] [`<param>`]

Where `<qualifiers>` are:

`--help` [`-h`]: Display this message

`--halt`: Halt the SP for reason code

Parameter

[`<reason>`]: A `<reason code>` is self defined by the user.

`--reboot` [`-r`]: Reboot the system

Parameter

[`<reason>`]: A `<reason code>` is self defined by the user.

`--system-halt` [`--force`]: Halt the complete system. force omits confirmation.

Example Usage

Example 1: User wants to reboot SPA

```

service@VNXe-spa:~/> svc_shutdown -r
service@VNXe-spa spa:~>
Broadcast message from root@spa
      (unknown) at 21:22 ...
The system is going down for reboot NOW!

```

Example 2: User wants to shutdown the system

```

service@ VNXe-spa:~> svc_shutdown --system-halt

#####
#####

WARNING: This action will shut down the system and you will
have to manually bring it back up afterwards.

#####
#####Enter "yes" if want to proceed with this
action: yes

Normal Mode

1

1

Peer shutdown now in progress

System shutdown now in progress

```

Related Commands

Service Mode Information (svc_rescue_state)

Outputs the health status of the system partition's storage device.

Function: System Operations

Mode: Both

Usage: Technical Service

Description

This command collects and outputs information about the SP hosting the system partition.

Self-Monitoring, Analysis, and Reporting Technology (SMART) is an industry-standardized, drive-based failure detection and monitoring system. SMART monitoring is performed independently by the drive's on-board disk controller. It uses the drive's onboard sensors and the manufacture's diagnostic algorithms to continuously monitor and evaluate drive health. The state of the device is recorded in several *attribute* counters. Each health check has its own attribute.

When run with no options, this command assesses the health of the onboard storage device. The command also estimates the percentage of service hours remaining on the device, based on current usage. The SMART attribute values of the device are recorded at the time of the run into a database. The location option ("--db-location") outputs the logical path to this database.

Output System
Partition's SMART
Information
(svc_smartdata)

The Read option displays the difference between the first run of the command and the most current run. This is useful to see trends in the health status of the device. If a difference in a critical attribute is found, the command returns a fail status.

The `dumpdata` option outputs all the attributes. Format of the output is:

```
RUN#|UTC Timestamp of RUN|Erasable|AttrID#|P/F of attr|Value  
of ATTR ID
```

The command returns zero (0) for a successful run and a healthy disk. A one (1) is returned if an error occurs or the storage device is not healthy. A two (2) is returned if the device has shown an unexpected amount of wear since the last command run.

Use Cases

Usage: `svc_smartdata <[options]> <param>`

Options:

```
--help [-h]: Displays this message  
--read [-r]: Read data  
--dump-data [-d]: Dumps database  
--db-location: Displays location of database  
--parseable: Displays output in parseable format
```

Example Usage

```
service@VNXe-spa:~> svc_smartdata  
INFO: Backend found, attempting to use backend database ...  
SMART overall-health self-assessment test result: PASSED  
INFO: 94% of life remaining for device /dev/ssd.
```

Related Commands

None.

Enable Secure Shell (svc_ssh)

Turns the Secure Shell Daemon (SSHD) on or off.

Function: System Operations

Mode: Both

Usage: Technical Service

Description

Enables or disables the ability for the service user to connect to the VNXe's Linux CLI via SSH connection over the system's Management interface. If SSH connectivity is disabled, or did not turn on when requested from within Unisphere a Service user may manually enable SSH while logged into Linux through a serial connection to the SP with this command.

This command may also be used to disable SSH connections to the VNXe system. If SSH is disabled while a user is connected to the SP through SSH, that user's session will end.

Use Cases

Usage: `svc_ssh [options]`

Options:

```
--help [-h]: Displays this text.  
--enable [-e]: Enables SSHD on the system  
--disable [-d]: Disables SSHD on the system
```


--status [-s]: Prints status of sshd

Example Usage

```
service@VNXe-spa:~> svc_ssh -d
Disabling sshd ...
Connection to 10.x.x.43 closed by remote host.
```

Related Commands

None.

Run FSCK on Storage (svc_storage_integritycheck)

Runs File System Check (FSCK) on storage file systems and returns them to a mountable state.

Function: Recovery

Mode: Normal

Usage: Technical Service

Description

Automatically unmounts file systems that are in use, runs FSCK on them, and then remounts the file systems when it is safe to do so.

This command can be used to get a file system remounted as soon as possible.

A file system will only be listed as an output if it is corrupt and unmounted. The command gives an option to fix and remount corrupted and unmounted file systems.

Use Cases

Usage: svc_storage_integritycheck [<qualifiers>]

Where the qualifiers are:

--help [-h]: Display this message

Example Usage

```
service@sentry-564-spa spa:~> svc_storage_integritycheck
=====
List of corrupt storage elements found:
=====
1 -> tst
=====
Please enter the item you would like to fix from the list,
or enter "q" to quit: 1

#####
#####
WARNING: Attempting to fix this corruption may take a long
time to complete
Please wait for status to say "Done..Done" before proceeding
#####
#####
Enter "yes" if want to proceed with this action: yes
Running check on: 1 -> tst
id = 23
name = tst
```

```

volume = v100
fsck_server = server_2
status = Done..Done
Attempting to mount server_2:tst...
server_2 : done
service@VNXe-spa:~> svc_storage_integritycheck
Storage integrity verified successfully. All storage
elements appear to be healthy.

```

Related Commands

None.

Output Storage Information (svc_storagecheck)

Outputs specific information about storage content.

Function: Diagnostic

Mode: Normal

Usage: General Use

Description

This script allows you to investigate specifics about all virtual storage content on the system. Information about particular kinds of shares (CIFS or NFS), iSCSI storage or lower layers in the configuration stack can be queried.

This script can only be run on the Primary SP. Output is logged to the `/home/service/svc_storagecheck.log` file. The output of this command is lengthy if running all available checks. It is recommended to view the output log at the location specified above using the Linux `less` utility.

Use Cases

Usage: `svc_storagecheck [command] <parameters>`

where the commands are:

```

--help [-h]: Output this text.
--dedupe [-d]
--backend [-b]
--sizes [-s]
--list [-l]
--nascheckpoint [-p
    Parameter:
        [<fs_name>]
--iscsi [-i]: iSCSI port.
    Parameter:
        [<-target>]
        [<-lun>]
        [<-snaps>]
        [<-stats>]
        [<-all>]: All iSCSI metrics
--nfs [-n]
--cifs [-c]
--ndmp [-d]
--replication [-r]
    Parameter:
        [<-sessions>]
        [<-remotesys>]
        [<-interconnects>]
--all [-a]: Output all metrics.

```

Example Usage

```
service@VNXe-spa:~> cd ~
service@VNXe-spa:~> less svc_storagecheck.log
===== [Thu May 5 13:15:32 UTC 2011]
Beginning Run =====
=====
===== Now running
/EMC/CEM/bin/clariiontool/clariiontool lsAll .
.. =====
Dump all information about system

[Output continues]
```

Related Commands

End-to-End Storage Mapping (`svc_neo_map`), Collect Performance Information (`svc_perfcheck`)

Output VNxe System Status (`svc_sysstatus`)

Outputs high-level status information about the VNxe system.

Function: Diagnostic

Mode: Both

Usage: Technical Service

Description

Outputs an XML formatted data structure containing information on the system's current health.

This command is mainly used by other system utilities to create an machine-parsable, XML-formatted, high-level “snapshot” of the system's state. For manual troubleshooting it is recommended to use the `svc_diag` command over this command.

Use Cases

Usage: `svc_sysstatus [<qualifiers>]`

The qualifiers are:

`--help [-?]`: Display this message.

`--quick [-q]`: Quick mode (displays the local subset of system state).

Related Commands

System Diagnostics (`svc_diag`), Hardware Component Tree (`svc_hw_tree`).

Redirect Output (`svc_tcpdump`)

Attempts to run `tcpdump` network diagnostics on a given system interface in a safe, controlled manner.

Function: Diagnostic

Mode: Both

Usage: Technical Service

Description

This service script allows the user to run a Linux tcpdump on a system interface for diagnostic purposes. The output is saved in rotating files of fixed size. When an output file grows to either the size defined by -C or the maximum internally defined size, output will be redirected to another file with the same base name but different postfix. The postfix will be a digit from 0 to the value defined by either the -W option or the internally defined maximum rotation value. Rotating files are filled in numerical order.

Output files from this command may be read using the tcpdump -r command.

Many options are analogous to their tcpdump counterparts so usage is similar to the standard tool.

This script may be run in any operational mode.

Use Cases

Usage: svc_tcpdump [options]

where the options are:

-?|--help

-i|--interface [name] : Interface to dump

-w|--filename [name] : Base file name for output files

-p|--path [path] : Path where the output files will be stored

-W|--rotations [number] : Number of files to be used for output

-C|--size [size] : Size of each output file (in MB)

-t|--timestamp [1-4] : Timestamp commands:

1: Don't print a timestamp on each dump line

2: Print an unformatted timestamp on each dump line

3: Print a delta (in micro-seconds) between current and previous line on each dump line

4: Print a timestamp in default format preceded by date on each dump line.

-v|--verbosity [1-3] : Specify verbosity of output, 3 being most

-D|--dump-intfs : Print the list of the network interfaces available on the system and on which tcpdump can capture packets.

-F|--input-expr [file] : Use file as input for the filter expression

-e|--llheader : Print the link-level header on each dump line

-n|--no-addr : Don't convert addresses (i.e., host addresses, port numbers, etc.) to names.

-q|--quiet : Print less protocol information so output lines are shorter

-y|--dlink [type] : Set the data link type to use while capturing packets to datalinktype.

DEFAULTS AND CONSTRAINTS

Rotations (-W) and size (-C) must be ≥ 1 . The default file size is 50MB and the default number of files is 5. The default interface is mgmt_vdev. By default, the output files are named "dump.out[0-4]" & stored in /home/service

Example Usage:

EXAMPLE

Capture mgmt_vdev, save 2 100MB files called tcpdump.out[0-1] to /home/service:

```
svc_tcpdump -i mgmt_vdev -p /home/service -w tcpdump.out -W 2 -C 100
```

Related Commands

None.