

Elastic Storage Server
Version 5.3.3

Command Reference



Elastic Storage Server
Version 5.3.3

Command Reference



Note

Before using this information and the product it supports, read the information in “Notices” on page 139.

This edition applies to version 5.3.3.x of the Elastic Storage Server (ESS) for Power, to version 5 release 0 modification 2 of the following product editions, and to all subsequent releases and modifications until otherwise indicated in new editions:

- IBM Spectrum Scale Data Management Edition for IBM ESS (product number 5765-DME)
- IBM Spectrum Scale Data Access Edition for IBM ESS (product number 5765-DAE)

Significant changes or additions to the text and illustrations are indicated by a vertical line (|) to the left of the change.

IBM welcomes your comments; see the topic “How to submit your comments” on page ix. When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright IBM Corporation 2014, 2019.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Tables	v
-------------------------	----------

About this information	vii
---	------------

Who should read this information.	vii
Related information	vii
Conventions used in this information	viii
How to submit your comments	ix

Chapter 1. ESS commands	1
--	----------

gssaddnode command	3
gsscallhomeconf script	5
gsscallhomeevent command	7
gsscheckdisks command	9
gsschenv command.	13
gsscrchxml command	15
gssfindmissingdisks command	17
gssgencluster command	19
gssgenclusterrgs command	22
gssgennetworks command	25
gssgendisks command	30
gssinstallcheck command.	34
gssnettest command	37
gssnodedetails command	39
gssprecheck command.	42
gssruntask command	45
gssstoragequickcheck command	47
gsssnap script	49
gssstress command	51
gssutils command	53
gssutils main menu.	56
gssutils - Install/Deploy menu	58
gssutils - Create GPFS cluster menu	62
gssutils - Create and manage GPFS cluster menu	64
gssutils - Install the ESS GUI menu	66
gssutils - Manage high-speed network menu	68
gssutils - EMS node deployment menu	71
gssutils - EMS node update menu	74
gssutils - Deploy I/O nodes menu.	76
gssutils - I/O nodes precheck menu	79
gssutils - I/O nodes post check menu	81

gssutils - Plug N Play and Fusion mode deployment menu	83
gssutils - Fusion mode deployment menu	85
gssutils - SSR Tools menu	87
gssutils - Upgrade menu	89
gssutils - Upgrade precheck menu.	92
gssutils - Upgrade EMS menu	94
gssutils - Upgrade EMS menu	96
gssutils - Extract ESS binaries menu	98
gssutils - Upgrade EMS detailed menu	100
gssutils - Actual EMS upgrade menu	103
gssutils - EMS pre-upgrade task menu	105
gssutils - EMS post-upgrade task menu.	107
gssutils - Upgrade I/O server menu.	108
gssutils - Upgrade I/O server menu.	109
gssutils - Upgrade I/O server node detailed menu	110
gssutils - Actual I/O server node upgrade menu	112
gssutils - Pre-upgrade I/O server menu	114
gssutils - Post-upgrade I/O server menu	116
gssutils - Validation and stress test menu	117
gssutils - Storage subsystem details menu	119
gssutils - Collect misc service data menu	121

Chapter 2. ESS scripts	123
---	------------

gssdelvdisks script	124
gssdeploy script	125
gssinstall script.	130
gsssnap script	132
gssupgrade.sh script	134

Chapter 3. ESS environment variables	137
---	------------

Notices	139
--------------------------	------------

Trademarks	140
----------------------	-----

Glossary	143
---------------------------	------------

Index	149
------------------------	------------

Tables

1. Conventions	viii	3. ESS environment variables	137
2. Supported tasks for gssruntask	45		

About this information

This information guides you in planning for and deploying the Elastic Storage Server (ESS) Version 5.3.3.x for Power® and all subsequent modifications of and fixes for this release.

Who should read this information

This information is intended for system operators and service technicians who have extensive knowledge of networking and Serial Attached SCSI (SAS) technology.

Related information

ESS information

The ESS 5.3.3.x library consists of following information units. You can access these publications on IBM® Knowledge Center or IBM Publications Center.

- *Elastic Storage Server: Quick Deployment Guide*, SC27-9526
- *Elastic Storage Server: Protocol Nodes Quick Deployment Guide*, SC27-9527
- *Elastic Storage Server: Problem Determination Guide*, SC27-9529
- *Elastic Storage Server: Command Reference*, SC27-9530
- *IBM Spectrum Scale RAID: Administration*, SC27-9528
- *IBM ESS Expansion: Quick Installation Guide (Model 084)*, SC27-4627
- *IBM ESS Expansion: Installation and User Guide (Model 084)*, SC27-4628
- *IBM ESS Expansion: Hot Swap Side Card - Quick Installation Guide (Model 084)*, GC27-9210
- *IBM ESS Expansion: Hardware Installation and Maintenance Guide (Model 106)*, SC27-9211
- *IBM ESS Expansion: Overview of CMA and Rail Kit Hardware Fasteners (Model 106)*, SC27-9296
- *Installing the Model 024, ESLL, or ESLS storage enclosure*, GI11-9921
- *Removing and replacing parts in the 5147-024, ESLL, and ESLS storage enclosure*
- *Disk drives or solid-state drives for the 5147-024, ESLL, or ESLS storage enclosure*
- For information about the DCS3700 storage enclosure, see:
 - *System Storage® DCS3700 Quick Start Guide*, GA32-0960-04:
 - *IBM System Storage DCS3700 Storage Subsystem and DCS3700 Storage Subsystem with Performance Module Controllers: Installation, User's, and Maintenance Guide*, GA32-0959-07:
<http://www.ibm.com/support/docview.wss?uid=ssg1S7004920>
- For information about the IBM Power Systems EXP24S I/O Drawer (FC 5887), see IBM Knowledge Center :
http://www.ibm.com/support/knowledgecenter/8247-22L/p8ham/p8ham_5887_kickoff.htm

For more information, see IBM Knowledge Center:

http://www-01.ibm.com/support/knowledgecenter/SSYSP8_5.3.3/sts533_welcome.html

For the latest support information about IBM Spectrum Scale RAID, see the IBM Spectrum Scale RAID FAQ in IBM Knowledge Center:

<http://www.ibm.com/support/knowledgecenter/SSYSP8/gnrfaq.html>

Other related information

For information about:

- IBM Spectrum Scale, see:
http://www.ibm.com/support/knowledgecenter/STXKQY/ibmspectrumscale_welcome.html
- IBM Spectrum Scale call home, see Understanding call home.
- Installing IBM Spectrum Scale and CES protocols with the installation toolkit, see Installing IBM Spectrum Scale on Linux nodes with the installation toolkit.
- IBM POWER8[®] servers, see IBM Knowledge Center:
<http://www.ibm.com/support/knowledgecenter/POWER8/p8hdx/POWER8welcome.htm>
- Extreme Cluster/Cloud Administration Toolkit (xCAT), go to the xCAT website :
<http://xcat.org/>
 - xCAT 2.14.5 Release Notes[®]
- Mellanox OFED Release Notes (4.5), go to http://www.mellanox.com/related-docs/prod_software/Mellanox_OFED_Linux_Release_Notes_4_5-1_0_1_0.pdf
- IBM Electronic Service Agent (ESA) documentation, go to <https://www-01.ibm.com/support/esa/>.
- Drive call home, go to Drive call home in 5146 and 5148 systems.

Conventions used in this information

Table 1 describes the typographic conventions used in this information. UNIX file name conventions are used throughout this information.

Table 1. Conventions

Convention	Usage
bold	Bold words or characters represent system elements that you must use literally, such as commands, flags, values, and selected menu options. Depending on the context, bold typeface sometimes represents path names, directories, or file names.
<u>bold underlined</u>	<u>bold underlined</u> keywords are defaults. These take effect if you do not specify a different keyword.
constant width	Examples and information that the system displays appear in constant-width typeface. Depending on the context, constant-width typeface sometimes represents path names, directories, or file names.
<i>italic</i>	<i>Italic</i> words or characters represent variable values that you must supply. <i>Italics</i> are also used for information unit titles, for the first use of a glossary term, and for general emphasis in text.
<key>	Angle brackets (less-than and greater-than) enclose the name of a key on the keyboard. For example, <Enter> refers to the key on your terminal or workstation that is labeled with the word <i>Enter</i> .
\	In command examples, a backslash indicates that the command or coding example continues on the next line. For example: <pre>mkcondition -r IBM.FileSystem -e "PercentTotUsed > 90" \ -E "PercentTotUsed < 85" -m p "FileSystem space used"</pre>
{item}	Braces enclose a list from which you must choose an item in format and syntax descriptions.
[item]	Brackets enclose optional items in format and syntax descriptions.
<Ctrl-x>	The notation <Ctrl-x> indicates a control character sequence. For example, <Ctrl-c> means that you hold down the control key while pressing <c>.

Table 1. Conventions (continued)

Convention	Usage
<i>item...</i>	Ellipses indicate that you can repeat the preceding item one or more times.
	In <i>synopsis</i> statements, vertical lines separate a list of choices. In other words, a vertical line means <i>Or</i> . In the left margin of the document, vertical lines indicate technical changes to the information.

How to submit your comments

Your feedback is important in helping us to produce accurate, high-quality information. You can add comments about this information in IBM Knowledge Center:

http://www.ibm.com/support/knowledgecenter/SSYSP8/sts_welcome.html

To contact the IBM Spectrum Scale development organization, send your comments to the following email address:

scale@us.ibm.com

Chapter 1. ESS commands

This topic includes descriptions of the ESS commands.

Descriptions of these ESS commands follow:

- “gssaddnode command” on page 3
- “gsscallhomeconf script” on page 5
- “gsscallhomeevent command” on page 7
- “gsscheckdisks command” on page 9
- “gsschenv command” on page 13
- “gsscrchxml command” on page 15
- “gssfindmissingdisks command” on page 17
- “gssgencluster command” on page 19
- “gssgenclusterrgs command” on page 22
- “gssgennetworks command” on page 25
- “gssgenvdisk command” on page 30
- “gssinstallcheck command” on page 34
- “gssnettest command” on page 37
- “gssnodedetails command” on page 39
- “gssprecheck command” on page 42
- “gssruntask command” on page 45
- “gssstoragequickcheck command” on page 47
- “gssstress command” on page 51
- “gssutils command” on page 53

For information about ESS scripts, see Chapter 2, “ESS scripts,” on page 123.

For information about these IBM Spectrum Scale RAID commands, see *IBM Spectrum Scale RAID:*

Administration:

- mmaddcomp**
- mmaddcompspec**
- mmaddpdisk**
- mmchcarrier**
- mmchcomp**
- mmchcomploc**
- mmchenclosure**
- mmchfirmware**
- mmchpdisk**
- mmchrecoverygroup**
- mmcrrecoverygroup**
- mmcrvdisk**
- mmdelcomp**
- mmdelcomploc**
- mmdelcompspec**
- mmdelpdisk**
- mmdelrecoverygroup**
- mmdelvdisk**
- mmdiscovercomp**
- mmgetpdisktopology**
- mmlscomp**
- mmlscomploc**
- mmlscompspec**
- mmlsenclosure**

mmlsfirmware
mmlspdisk
mmlsrecoverygroup
mmlsrecoverygroupevents
mmsyncdisplayid

For information about these IBM Spectrum Scale RAID scripts, see *IBM Spectrum Scale RAID:*

Administration:

chdrawer
gnrhealthcheck
mkrinput
topselect
topsummary

For information about other IBM Spectrum Scale commands, see *IBM Spectrum Scale: Command and Programming Reference*.

gssaddnode command

Adds a node (e.g. EMS) to a GPFS cluster.

Synopsis

```
gssaddnode -N ADD-NodeList { --cluster-node ClusterNode | --cluster-node-group ClusterNodeGroup }  
      [ --nodetype NodeType ] [ --prefix Prefix ] [ --suffix Suffix ]  
      [ --accept-license ] [ --no-fw-update ] [ --contact-node ContactNode ]  
      [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The **gssaddnode** command adds nodes to a GPFS cluster. It can be used to add an EMS or I/O server node to the cluster. This command must run on the EMS node when the EMS node is being added to the cluster. The EMS node must be upgraded to the correct software levels (such as RHEL, xCAT and GPFS) before adding the node to the cluster. This command updates firmware levels in the SAS host adapter, firmware in the enclosure and drives.

Parameters

-N *ADD-NodeList*

Provides a comma separated list of nodes to add to an existing GPFS cluster. In case of EMS node it should be the host name of the EMS (For example, ems1).

--cluster-node *ClusterNode*

Provides the name of a node that exists in a GPFS cluster where nodes will be added. This node must be able to run GPFS administrative commands. For example, **--cluster-node** gssio1 where gssio1 is an existing node in the cluster. Either **--cluster-node** or **--cluster-group** must be provided.

--cluster-node-group *ClusterNodeGroup*

Provides the xCAT node group name of the existing cluster. Add node command is run in the first node of the group. For example, **--cluster-node-group** gss_ppc64 where gss_ppc64 is an existing xCAT group. Either **--cluster-node** or **--cluster-node-group** must be provided.

--nodetype *NodeType*

Indicates the type of the node being added. Supported node types include ems and gss. Default nodetype is ems.

--prefix *Prefix*

Provides the hostname prefix. PREFIX is used with the node names provided in the ADD-NODE-LIST to create the actual node names. Use = between **--prefix** and value if the value starts with -.

--suffix *Suffix*

Provides the hostname suffix. SUFFIX is used with the node names in the ADD-NODE-LIST to create the actual node names. For example, with a addnodelist of gssio1,gssio2 and prefix A- and suffix -ib node names A-gssio1-ib and A-gssio2-ib is used to form the actual node names. The node name must be resolvable. Use = between **--suffix** and value if the value starts with -.

--accept-license

Provides the **--accept-license** indicating that the applicable licensing terms are accepted. If not provided user will be prompted for license acceptance after addition of the node.

--no-fw-update

This option skips SAS adapter, storage enclosure and drive firmware update after addition of the nodes into the cluster.

--contact-node *ContactNode*

Provides a node name where the command is run. The contact node must be reachable over ssh without password if it is not member of a xCAT group.

-h | --help

Displays usage information about this command and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the **gssaddnode** command.

Examples

1. This example shows EMS node `ems1` being added to the cluster. In the first example, group name of the existing cluster nodes is provided. In the second example a node of the existing cluster is provided.

```
gssaddnode -N ems1 --cluster-node-group gss_ppc64 --nodetype ems --accept-license  
gssaddnode -N ems1 --cluster-node gssio1 --nodetype ems --accept-license
```

2. This example shows the new I/O server nodes `gssio3` and `gssio4` being added to the existing cluster. In the first example, group name of the existing cluster nodes is provided. In the second example, a node of the existing cluster is provided. While adding I/O server node, make sure to use `--nodetype` as `gss`. The `gss` node type indicates that the newly added node as an NSD server I/O server node.

```
gssaddnode -N gssio3 --cluster-node-group gss gss_ppc64 --nodetype gss --accept-license  
gssaddnode -N gssio4 --cluster-node gssio1 --nodetype gss --accept-license
```

See also

See also the following *IBM Spectrum Scale: Command and Programming Reference* topics:

- **mmaddnode**
- **mmchconfig**
- **mmchlicense**
- **mmcrnodeclass**
- **mmstartup**

Location

`/opt/ibm/gss/tools/bin`

gsscallhomeconf script

Performs ESS HW call home configuration

Synopsis

```
gsscallhomeconf { [ -N NODE-LIST | -G NODE-GROUP ] } [ --show ] [ --prefix PREFIX ]  
[ --suffix SUFFIX ] -E ESA-AGENT [ --register { node,all } ] [ --no-swcallhome ] [ --crvpd ]  
[ --serial SOLN-SERIAL ] [ --model SOLN-MODEL ] [ --verbose ] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

You can use the **gsscallhomeconf** script to configure ESS for call home event generation using the Electronic Service Agent (ESA). This script can be run only on the ESS models with ppc64 architecture. With IBM Spectrum Scale RAID running on the ESS nodes, callback events about disk failures are generated on ppc64 architecture in the attached enclosure requiring replacement. In response to the callback events, **gsscallhomeconf** or **gsscallhomeevent** initiates a call home request to ESA over REST API. ESA is installed in the EMS node, and call home events when initiated by the **gsscallhomeconf** or **gsscallhomeevent**. The **gsscallhomeconf** and **gsscallhomeevent** scripts run on I/O server nodes and the EMS node. This release supports disk replacement events for attached enclosures only.

gsscallhomeconf also configures software call home along with hardware call home unless **--no-swcallhome** is specified. If you do not want to use software call home along with hardware call home, use the **--no-swcallhome** option. For more information, see Software call home documentation.

Parameters

-N *NODE-LIST*

Provides a list of nodes to configure.

-G *NODE-GROUP*

Provides the name of node group.

--prefix *PREFIX*

Provides the hostname prefix. Use = between **--prefix** and value if the value starts with -.

--suffix *SUFFIX*

Provides the hostname suffix. Use = between **--suffix** and value if the value starts with -.

-E *ESA-AGENT*

Provides the node name on which the ESA agent is installed and running.

--register {*node,all*}

Registers the endpoints (nodes, enclosure or all) with ESA.

--no-swcallhome

Specifies to not configure software call home while configuring hardware call home. The **gsscallhomeconf** command configures software call home with hardware call home by default unless this option is specified.

--crvpd

Creates the solution vital product data (VPD) file.

--serial *SOLN-SERIAL*

Provides the ESS solution serial number in the VPD file.

--model *SOLN-MODEL*

Provides the ESS model.

--verbose

Provides the verbose output.

-h | --help

Displays usage information about this script and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the **gsscallhomeconf** script.

Example

The following examples show configuring call home:

```
# ./gsscallhomeconf -G gss_ppc64 -E ems1 --suffix=-ib --register node
2018-05-28T02:40:35.422902 Generating node list...
2018-05-28T02:40:36.129896 nodelist: gssiol gssio2
2018-05-28T02:40:36.129933 suffix used for endpoint hostname: -ib
End point gssiol-ib registered successfully with systemid 3ecc9e664c8ce85d6bba3f6c7c3ae523
End point gssio2-ib registered successfully with systemid da4b3bd9d127d9285a4f73a55c232ecd
ESA configuration for ESS Call home is complete.
Started configuring software callhome
Checking for ESA is activated or not before continuing.
Fetching customer detail from ESA.
Customer detail has been successfully fetched from ESA.
Setting software callhome customer detail.
Successfully set the customer detail for software callhome.
Enabled daily schedule for software callhome.
Enabled weekly schedule for software callhome.
Direct connection will be used for software calhome.
Successfully set the direct connection settings for software callhome.
Enabled software callhome capability.
Creating callhome automatic group
Created auto group for software call home and enabled it.
Software callhome configuration completed.

# gsscallhomeconf -E ems3 -N ems3,gss_ppc64 --suffix=-te --register=all --no-swcallhome
2017-01-23T05:34:42.005215 Generating node list...
2017-01-23T05:34:42.827295 nodelist: ems3 essio31 essio32
2017-01-23T05:34:42.827347 suffix used for endpoint hostname: -te
End point ems3-te registered successfully with systemid 37e5c23f98090750226f400722645655
End point essio31-te registered successfully with systemid 35ae41e0388e08fd01378ae5c9a6ffef
End point essio32-te registered successfully with systemid 9ea632b549434d57baef7c999dbf9479
End point enclosure SV50321280 registered successfully with systemid 600755dc0aa2014526fe5945981b0e08
End point enclosure SV50918672 registered successfully with systemid 92aa6428102b44a4a1c9a293402b324c
ESA configuration for ESS Callhome is complete.
```

Location

/opt/ibm/gss/tools/bin

gsscallhomeevent command

Use the **gsacallhomeevent** command to send call home events to the Electronic Service Agent (ESA).

Synopsis

```
gsscallhomeevent [ --systemid EVENTSID ]  
[ --event {postRGTakeover | postRGRelinquish | rgOpenFailed | rgPanic | pdFailed |  
pdRecovered | pdReplacePdisk | pdPathDown | daRebuildFailed} ]  
[ --eventName EVENTNAME ] [ --myNode MYNODE ]  
[ --rgName RGNAME ] [ --rgErr RGERR ]  
[ --rgReason RGREASON ] [ --daName DANAME ]  
[ --pdName PDNAME ] [ --pdLocation PDLOCATION ]  
[ --pdFru PDFRU ] [ pdWwn PDWWN ] [ --pdState PdState ]  
[ --daRemainingRedundancy DAREMAININGDEDUNDANCY ]  
[ --verbose ] [ --show ] [ --upload UPLOADFILES ]  
[ --collector COLLECTOR ]  
[ --heartbeatnode HEARTBEATNODE ]  
[ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

You can use the **gsscallhomeevent** command to send call home events to the ESA agent.

Parameters

-h | **--help**

Specifies the help message and exit.

--systemid *EVENTSID*

Specifies the system ID for the event.

--event {*postRGTakeover*,*postRGRelinquish*,*rgOpenFailed*,*rgPanic*,*pdFailed*,*pdRecovered*,
pdReplacePdisk,*pdPathDown*,*daRebuildFailed*}

Specifies the call home event.

--eventName *EVENTNAME*

Specifies the event name.

--myNode *MYNODE*

Specifies the node that generates the event.

--rgName *RGNAME*

Specifies the recovery group (RG) name.

--rgErr *RGERR*

Specifies the RG error information.

--rgReason *RGREASON*

Specifies the RG reason.

--daName *DANAME*

Specifies the declustered array (DA) name that is associated with the event.

--pdName *PDNAME*

Specifies the physical disk (pdisk) name.

--pdLocation *PDLOCATION*

Specifies the pdisk location.

- pdFru** *PDFRU*
Specifies the pdisk field replaceable unit (FRU).
- pdWwn** *PDWWN*
Specifies the pdisk unique worldwide name (WWN).
- pdState** *PDSTATE*
Specifies the physical disk pdisk state.
- deRemainingRedundancy** *DAREMAININGREDUNDANCY*
Specifies the remaining redundancy of the associated DA.
- verbose**
Specifies the verbose output.
- show**
Specifies the endpoint and event details.
- upload** *UPLOADFILES*
Specifies the list of upload files and their location for the event.
- collector** *COLLECTOR*
Specifies the command or script to generate the upload file. There is no default value.
- heartbeatnode** *HEARTBEATNODE*
Specifies the heartbeat that is sent from the heartbeat node to ESA agent.

Security

You must have root authority to run the **gsscallhomeevent** command.

Exit status

- 0** Successful completion.
- nonzero** A failure occurs.

Example

This examples shows how to send call home events to the ESA agent.

```
gsscallhomeevent --eventName ReplaceDisk --event
pdReplacePdisk --systemid 37a6259cc0c1dae299a7866489dff0bd
```

Location

/opt/ibm/gss/tools/samples

gsscheckdisks command

Checks attached disks for errors under various I/O operations.

Synopsis

```
gsscheckdisks { -N NODE-LIST | -G NODE-GROUP }  
  [ --prefix Prefix ] [ --suffix Suffix ]  
  { --enclosure-list Enclosure-List | --disk-list DiskList | --show-enclosure-list }  
  [ --iotest Io-Test ] [ --ioengine Io-Engine ]  
  [ --batch-size Batch-Size ] [ --duration Test-Duration ]  
  [ --write-enable ] [ --iopath Io-Path ] [ --local ]  
  [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The **gsscheckdisks** command checks the attached drives for disk errors under various I/O operations before configuring (part of recovery group) them. At the end of the test, the number of errors (and net performance) per device is reported. This test is typically run on the EMS node. It can also be run on the I/O server node with the **--local** option. If any disk shows error (non-zero error count), it needs to be addressed before adding the drive in the recovery group. This is not a performance tool and performance statistics shown are approximate and relative to other disks in the same batch of the run. When run from the EMS node, summary results are saved in the **/temp/checkdisk** directory of the EMS node.

Parameters

-N *NODE-LIST*

Specifies a list of nodes on which to run the test.

You must specify the **-N** or **-G** parameter.

-G *NODE-GROUP*

Specifies the xCAT node group name of the I/O server nodes where the test is run. Either *NODE-LIST* or *NODE-GROUP* must be provided. This parameter cannot be used in conjunction with the **--local** parameter.

--prefix *Prefix*

Specifies the host name prefix. *Prefix* is used with the *NODE-LIST* to create the node name for the test. Use = between **--prefix** and value if the value starts with -.

--suffix *Suffix*

Specifies the node name suffix. *Suffix* is used with *NODE-LIST* to create the node name for the test.

For example, with a node list of *gssio1,gssio2* and prefix *A-* and suffix *-ib* node names *A-gssio1-ib* and *A-gssio2-ib* is used to run the test. The newly formed node name must be resolvable. Use = between **--suffix** and value if the value starts with -.

--enclosure-list { *EnclosureList* }

Specifies a list of enclosures to be tested. Specify **all** to test all attached enclosures.

--disk-list *DiskList*

Specifies a list of disks to be tested, for example: **sdh, sdx, sdm**.

Disks are not checked for potential write to local disks such as *sda* when **--write-enable** option is provided. **--enclosure-list** and **--disk-list** are mutually exclusive.

--show-enclosure-list

Displays a list of enclosures attached to I/O server nodes.

This list, or a subset of it can be used to provide input to `--enclosure-list`. Program exits after displaying the list.

--iotest *IoTest*

Specifies a comma-separated list of I/O operations for testing. Valid values are **r** (sequential read), **w** (sequential write), **R** (random read), **W** (random write), and **a** (all). The default is **r** (sequential read).

I/O tests are run in the sequence they are provided. For example, `r,w,W,R,w,r`. If **a** is provided alone or as a part of a sequence, the given sequence is replaced with `r,w,R,W`.

| --ioengine *Io-Engine*

| Specifies the I/O engine to use.

| Valid values are **g** for `gfsperf`, **d** for `dd`, or **s** for `shred`.

--batch-size *Batch-Size*

Specifies the batch size (number of disks tested at a time) of the test. Specify 0 for all attached drive in a single batch. Default batch size is 60. Maximum batch size must be limited to 120.

--duration *Test-Duration*

Specifies the duration of the test in seconds. Default is 30 seconds. Specify 0 to start and exit **gsscheckdisks** while the I/O engine is running. It can be used to sequentially write to the end of the disk.

--write-enable

Enables read-write I/O operations to the drive. Default is read-only. For any test sequence that involves write operation, `--write-enable` must be specified.

--io-path *io-path*

Specifies the comma separated list of I/O paths for test. The default is `0,1` indicating both paths.

--local

Runs the I/O test locally on the I/O server node. The command can only be called from an I/O server node. When invoked with `--local` option at a I/O server node, `-N` must be provided with host name of the I/O server node.

-h | --help

Displays usage information about this command and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Restrictions

This command must be run on a system where there is no GPFS cluster configured.

This command must not be run on a system that has GPFS recovery groups.

Security

You must have root authority to run the **gsscheckdisks** command.

Examples

This example shows **gsscheckdisks** command running on node `gssi01` and performing all tests (including write) in drives in all enclosures. Run:

```
gsscheckdisks -N gssi01 --encl all --iotest a --write-enable
```

The system displays output similar to this:

```
gsscheckdisks -N gssiol --encl all --iotest a --write-enable
```

```
gsscheckdisks -G gss_ppc64 --encl all --iotest a --write-enable
2014-12-04T21:28:32.577341 Start running check disks
2014-12-04T21:28:33.801643 nodelist: gssiol gssio2
2014-12-04T21:28:34.104659 Running checkdisk on node gssiol
gssiol: 2014-12-04T21:28:35.121970 Start running check disks
gssiol: List of Enclosures found
gssiol: SV32300072
gssiol: SV24819545
gssiol: Taking inventory of disks in enclosure SV32300072.
gssiol: Taking inventory of disks in enclosure SV24819545.
gssiol: 2014-12-04T21:32:37.637318 Starting r test for 118 of 118 disks. Path: 0, duration 30 secs
gssiol: 2014-12-04T21:33:14.678887 Check disk analysis for r test Complete
gssiol: 2014-12-04T21:33:14.680854 Starting w test for 118 of 118 disks. Path: 0, duration 30 secs
gssiol: 2014-12-04T21:33:51.743056 Check disk analysis for w test Complete
gssiol: 2014-12-04T21:33:51.745072 Starting R test for 118 of 118 disks. Path: 0, duration 30 secs
gssiol: 2014-12-04T21:34:28.903142 Check disk analysis for R test Complete
gssiol: 2014-12-04T21:34:28.905101 Starting W test for 118 of 118 disks. Path: 0, duration 30 secs
gssiol: 2014-12-04T21:35:06.042941 Check disk analysis for W test Complete
gssiol: 2014-12-04T21:35:06.044872 Starting r test for 118 of 118 disks. Path: 1, duration 30 secs
gssiol: 2014-12-04T21:35:43.109214 Check disk analysis for r test Complete
gssiol: 2014-12-04T21:35:43.111221 Starting w test for 118 of 118 disks. Path: 1, duration 30 secs
gssiol: 2014-12-04T21:36:20.174434 Check disk analysis for w test Complete
gssiol: 2014-12-04T21:36:20.176328 Starting R test for 118 of 118 disks. Path: 1, duration 30 secs
gssiol: 2014-12-04T21:36:57.343535 Check disk analysis for R test Complete
gssiol: 2014-12-04T21:36:57.345505 Starting W test for 118 of 118 disks. Path: 1, duration 30 secs
gssiol: 2014-12-04T21:37:34.498058 Check disk analysis for W test Complete
```

```
2014-12-04T21:37:34.555089 Running checkdisk on node gssio2
gssio2: 2014-12-04T21:37:35.585243 Start running check disks
gssio2: List of Enclosures found
gssio2: SV32300072
gssio2: SV24819545
gssio2: Taking inventory of disks in enclosure SV32300072.
gssio2: Taking inventory of disks in enclosure SV24819545.
gssio2: 2014-12-04T21:41:45.408912 Starting r test for 118 of 118 disks. Path: 0, duration 30 secs
gssio2: 2014-12-04T21:42:22.495101 Check disk analysis for r test Complete
gssio2: 2014-12-04T21:42:22.496999 Starting w test for 118 of 118 disks. Path: 0, duration 30 secs
gssio2: 2014-12-04T21:42:59.598634 Check disk analysis for w test Complete
gssio2: 2014-12-04T21:42:59.600548 Starting R test for 118 of 118 disks. Path: 0, duration 30 secs
gssio2: 2014-12-04T21:43:36.889743 Check disk analysis for R test Complete
gssio2: 2014-12-04T21:43:36.891564 Starting W test for 118 of 118 disks. Path: 0, duration 30 secs
gssio2: 2014-12-04T21:44:14.163045 Check disk analysis for W test Complete
gssio2: 2014-12-04T21:44:14.165026 Starting r test for 118 of 118 disks. Path: 1, duration 30 secs
gssio2: 2014-12-04T21:44:51.283537 Check disk analysis for r test Complete
gssio2: 2014-12-04T21:44:51.285473 Starting w test for 118 of 118 disks. Path: 1, duration 30 secs
gssio2: 2014-12-04T21:45:28.386629 Check disk analysis for w test Complete
gssio2: 2014-12-04T21:45:28.388583 Starting R test for 118 of 118 disks. Path: 1, duration 30 secs
gssio2: 2014-12-04T21:46:05.681164 Check disk analysis for R test Complete
gssio2: 2014-12-04T21:46:05.682977 Starting W test for 118 of 118 disks. Path: 1, duration 30 secs
gssio2: 2014-12-04T21:46:42.957917 Check disk analysis for W test Complete
```

When **gsscheckdisks** command runs, it collects information about the tests including disk performance and error counters. Error information collected during the test identifies a failing disk or path(s) to a disk. Test results are summarized and stored in the checkdisk directory of the EMS node. A directory with timestamps is created for each run. For each test run two files are created. They are *<node>diskana0.csv* and *<node>diskana1.csv*, and contains summary results of disk IO throughout of each device every second as well one line summary of each device showing throughput and error count. Name of the *node* *<node>* where the test is running is prefixed to the output files. Each Disk summary line looks similar to this.

```
2015-01-03T19:38:05.783338 Disk: sdbx loc SV12616682:2-4 ST32000444SS path 0(sg61) Op w elapsed time: 30
total sector read 0 read-tput 0.00 MB/sec, elapsed time 30 total sector write 2021376 write-tput 32.90 MB/sec
devname sdbx ioreq 1016 iodone 1017 ioerr 0
2015-01-03T19:38:05.788092 Disk: sdcB loc SV12616682:2-8 ST32000444SS path 0(sg61) Op w elapsed time: 30
```

```
total sector read 0 read-tput 0.00 MB/sec, elapsed time 30 total sector write 1996800 write-tput
32.50 MB/sec devname sdc b ioreq 1090 iodone 1091 ioerr 0
2015-01-03T19:38:05.792839 Disk: sdbi loc SV12616682:1-1 ST32000444SS path 0(sg61) Op w elapsed time: 30
total sector read 0 read-tput 0.00 MB/sec, elapsed time 30 total sector write 1984512 write-tput
32.30 MB/sec devname sdbi ioreq 998 iodone 999 ioerr 0
```

Here *loc* is the location of the disk in the enclosure, drawer-slot format. *sgxx* device shown in the path (within parenthesis), *rep* represents ESM accessing a disk. The *topsummary* program (e.g., `mmgetpdisktopology|topsummary`) output shows *sg* address of ESM in the storage enclosure. Number of *ioreq*, *iodone* and *ioerr* are sampled from the `/sys/block/<Disk>/device` directory. They are sampled at the beginning and at the end of the test. They are otherwise not correlated and number of *ioreq* and *iodone* may not match. The key objective of this test is to determine if error free IO operations can be done on a disk. In addition to the performance and error summary following files are created in the `/tmp` directory of each IO server node.

diskiostat.csv: It stores samples of the `/proc/iostat` for every second during the test run and with following format:

- ****col1:**** time epoch,
- ****col2:**** node where test is run
- ****col3:**** device

The rest of the 11 columns are dumps of `/proc/iostat`. *deviceerr.csv*: It sores number of drive error count and sampled once every second.

- ****col1:**** time epoch
- ****col2:**** node where run
- ****col3:**** device
- ****col4:**** io issued

Location

`/opt/ibm/gss/tools/bin`

gsschenv command

Modifies the Elastic Storage Server (ESS) environment settings.

Synopsis

```
gsschenv [ -m | --modify ] --show [-r | --reboot] [-V | --version]
[ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

Use the **gsschenv** command to modify the ESS environment settings. You can modify the system environment settings, including:

- IP address
- Host name
- Subnet name
- Domain name

By default, the system ships with default IP address, host name and domain, etc. In the case of a customer environment, the customer might want to change the ESS IP address, host name, etc. This utility can be used to achieve this.

The **gsschenv** command can be used as part of the Fusion mode for ESS deployment.

Remember: Contact your system administrator before you select a new IP address, host name, subnet name, and domain name. Any invalid values cause **gsschenv** the command to hang and the nodes become unreachable.

In order to use this utility, you must modify environment settings in the `gsschenv.cfg` file, which can be found in the `/opt/ibm/gss/tools/conf` folder. You must modify the settings in this file before you specify the **gsschenv** command. The contents of `gsschenv.cfg` are similar to this example:

```
# -----
# /opt/ibm/gss/tools/conf/gsschenv.cfg
# -----
# Modify the following
# HOSTNAME_ORIG = Original hostnames in your xCAT ESS environment.
# IP_ADDR_ORIG = Original IPs in your xCAT ESS environment want (1 to 1 with HOSTNAME_ORIG)
# HOSTNAME_NEW = The new hostnames (1 to 1 with the HOSTNAME_ORIG)
# IP_ADDR_NEW = The new IPs you want (1 to 1 with HOSTNAME_NEW/ORIG)
# NETMASK = The new netmask associated with the IPs
# DOMAIN = The new domain associated with the IPs

HOSTNAME_ORIG=(ems1 gssio1 gssio2)
IP_ADDR_ORIG=(192.168.202.20 192.168.202.21 192.168.202.22)
HOSTNAME_NEW=(modems1 modgssio1 modgssio2)
IP_ADDR_NEW=(192.168.202.20 192.168.202.21 192.168.202.22)
NETMASK="255.255.255.0"
DOMAIN="gpfs.net"
```

Parameters

-h | --help
Specifies the help message and exit.

-m | --modify

Specifies that the following values be changed:

- IP address
- Host name
- Subnet name
- Domain name

-r | --reboot

Specifies an I/O node reboot after you modify the environment.

-v | --version

Specifies the version number and exit.

Security

You must have root authority to run the **gsschenv** script.

Exit status

0 Successful completion.

nonzero

A failure occurs.

Example

This example shows how to change the ESS environment to newer values specified in the `gsschenv.cfg` file after you have set the correct configuration in `gsschenv.cfg` file.

```
gsschenv --modify /opt/ibm/gss/tools/conf/gsschenv.cfg --reboot
```

After running this command, the system reboots automatically. After the reboot and in order to log in and use the system, you must use the new:

- IP address
- Host name
- Subnet name
- Domain name

Location

`/opt/ibm/gss/tools/samples`

gsscrchxml command

Creates an XML file for call home.

Synopsis

```
gsscrchxml [ -N NODE-LIST | -GNODE-GROUP ]  
[ --prefix PREFIX ] [ --suffix SUFFIX ]  
[ --callhomexml CALLHOME-XML ] [ --crvpd ]  
[ --serial SOLN-SERIAL ] [ --model SOLN-MODEL ]  
[ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

You can use the **gsscrchxml** command to create an XML file for call home.

Parameters

-h | **--help**

Specifies the help message and exit.

-N *NODE-LIST*

Specifies the list of nodes on which to run tests.

-G *NODE-GROUP*

Specifies the node group name.

--prefix *PREFIX*

Specifies the host name prefix.

Note: Use an equal sign (=) between **--prefix** and *PREFIX* if *PREFIX* starts with a hyphen (-).

--suffix *SUFFIX*

Specifies the host name suffix.

Note: Use an equal sign (=) between **--suffix** and *SUFFIX* if *PREFIX* starts with a hyphen (-).

--callhomexml *CALLHOME-XML*

Specifies the Elastic Storage Server (ESS) call home XML file name and location. The default value is */tmp/essch<serial>.xml*, where *<serial>* is the ESS solution serial number (see *SolnModel*).

--crvpd

Creates the vital product data (VPD) file.

--serial *SOLN-SERIAL*

Specifies the ESS solution serial number.

--model *SOLN-MODEL*

Specifies the ESS model.

--onscreen

Shows the XML file on stdout.

Security

You must have root authority to run the **gsscrchxml** script.

Exit status

0 Successful completion.

nonzero

A failure occurs.

Example

This example shows how to generate a call home XML file:

```
gsscrchxml -N ems1,gss_ppc64
```

Location

```
/opt/ibm/gss/tools/samples
```

gssfindmissingdisks command

Checks the disk paths and cabling connectivity.

Synopsis

```
gssfindmissingdisks { -N NodeList | -G NodeGroup }  
    [ --prefix Prefix ] [ --suffix Suffix ] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The **gssfindmissingdisks** command checks for incorrect and missing cable connections between I/O server nodes and storage enclosures. The command checks for:

- Paths to disks
- Proper cable connections

Parameters

-N *NodeList*

Specifies a comma-separated list of nodes to run the check.

You must specify the **-N** or **-G** parameter.

-G *NodeGroup*

Specifies the name of the xCAT node group.

You must specify the **-N** or **-G** parameter.

--prefix *Prefix*

Specifies the hostname prefix. Use *Prefix* with *NodeList* to generate node names when you run the check. Use = between **--suffix** and value if the value starts with -.

--suffix *Suffix*

Specifies the hostname suffix. Use *Suffix* with *NodeList* to generate node names when you run the check. For example:

- Node list *gssi01,gssi02*
- Prefix *A-*
- Suffix *-ib*
- Node names *A-gssi01-ib* and *A-gssi02-ib*

The newly formed node name must be resolvable to corresponding IP address. Use = between **--suffix** and value if the value starts with -.

-h | **--help**

Displays usage information about this command and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the **gssfindmissingdisks** command.

Examples

In this example, there are no missing drive paths; however, cables are not connected properly. Run:

```
gssfindmissingdisks -G gss_ppc64
```

Following example shows gssfindmissingdisks running on node gssio1

```
gssfindmissingdisks -N gssio1
```

The system displays output similar to this:

```
[root@ems1 deploy]# gssfindmissingdisks -G gss_ppc64
```

```
gssfindmissingdisks -G gss_ppc64
2015-06-10T19:17:28.058805 Start find missing disk paths
2015-06-10T19:17:29.905278 nodelist:   gssio1 gssio2
2015-06-10T19:17:29.905351 May take long time to complete search of all drive paths
2015-06-10T19:17:29.905384 Checking missing disk paths from node  gssio1
gssio1 Enclosure SV45221140 (number 1):
gssio1 Enclosure SV45222733 (number 2):
gssio1: GSS configuration: 2 enclosures, 2 SSDs, 2 empty slots, 118 disks total, 6 NVRAM partitions
2015-06-10T19:17:48.272489 Checking missing disk paths from node  gssio2
gssio2 Enclosure SV45221140 (number 1):
gssio2 Enclosure SV45222733 (number 2):
gssio2: GSS configuration: 2 enclosures, 2 SSDs, 2 empty slots, 118 disks total, 6 NVRAM partitions
2015-06-10T19:18:04.740198 Finish search for missing disk paths. Number of missing disk paths: 0
```

Location

```
/opt/ibm/gss/tools/bin
```

gssgencluster command

Creates a GSS cluster from a node-list or node-group

Synopsis

```
gssgencluster -C ClusterName { -N NodeList | -G NodeGroup }  
    [ --subnet SubnetList ] [ --prefix Prefix ] [ --suffix Suffix ]  
    [--accept-license] [--no-fw-update] -add-gss-callback  
    [ --change-configuration ChangeConfig ] [--delete-cluster] [ --contact-node ContactNode ]  
    [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

Use the **gssgencluster** to create a GSS cluster containing the servers listed in NODE-LIST or NODE-GROUP. This command creates the cluster, apply applicable license acceptance, apply configuration changes for GSS application, create a node-class with the group name (xCAT group) of the nodes, update host-adapter, storage enclosure and disk firmware. At the successful completion the member nodes are also started.

The ESS cluster is configured (by default) with a callback script for any GNR events. This command specifies when any of the following events occurs:

- postRGTakeover
- postRGRelinquish
- rgOpenFailed
- rgPanic
- pdFailed
- pdRecovered
- pdReplacePdisk
- pdPathDown
- daRebuildFailed

See the **mmaddcallback** command for more details on callback events. A brief description of callback events are recorded in the `/var/log/messages` file. The `gnrcallback350.sh` file located in the `/opt/ibm/gss/tools/samples` directory can be customized for additional actions (such as sending an email). The callback information can be deleted by using the **mmdelcallback** command.

Parameters

-C *ClusterName*

Specifies the name of the cluster. You must specify this parameter.

-N *NodeList*

Specifies a comma-separated list of nodes for the disk path check.

You must specify the **-N** or **-G** parameter.

-G *NodeGroup*

Specifies an xCAT node group.

You must specify the **-N** or **-G** parameter.

--subnet *SubnetList*

Specifies one or more subnet names in a comma-separated list. This parameter is used to populate the GPFS subnet parameter.

--prefix *Prefix*

Specifies the hostname prefix. This prefix value is applied to the node names that are provided in *NodeList* or *NodeGroup*. The combined node name is used as the host name for any GPFS cluster creation. Use an equal sign (=) between **--prefix** and *Prefix* if the prefix begins with a hyphen (-).

--suffix *Suffix*

Specifies the hostname suffix. This suffix value is applied to the node names that are provided in *NodeList* or *NodeGroup*. The combined node name is used as the host name for any GPFS cluster creation. For example a *NodeList* value of gssio1,gssio2, with a prefix value of A- and a suffix value of -ib, and with host names A-gssio1-ib and A-gssio2-ib can be used to create the GPFS cluster. The newly formed node name must be resolvable. Use an equal sign (=) between **--suffix** and *Suffix* if the suffix begins with a hyphen (-).

--accept-license

Indicates that you accept the applicable licensing terms. The license acceptance prompt is suppressed. If this option is not provided you must accept the license after you create the cluster.

--no-fw-update

Indicates that you do not want to apply firmware updates for the host adapter, storage enclosures, and drives. This option skips the SAS adapter, the storage enclosure, and the drive FW update that occurs after the GPFS cluster creation.

--add-gss-callback

Specifies the addition of a GSS callback with the identifier *gnrcallback*. The *gnrcallback.sh* script is supplied in */usr/lpp/mmfs/samples/vdisk* directory. This script is invoked when any of the following events occurs:

- postRGTakeover
- postRGRelinquish
- rgOpenFailed
- rgPanic
- pdFailed
- pdRecovered
- pdReplacePdisk
- pdPathDown
- daRebuildFailed.

(See the **mmaddcallback** command for further details on callback events. A brief description of any callback events are recorded in the *gnrcallback.log* file located in the */var/mmfs/tmp/gnrcallback.log*. The *gnrcallback.sh* file can be customized for additional actions (such as sending an email). The callback information can be deleted by using the **mmdelcallback** command.

--change-configuration *ChangeConfig*

Provides additional change cluster configuration options. The default is None. Enclose with quotation marks (' or ") if there are spaces within the configuration options. See the **mmchconfig** command for available configuration options.

--delete-cluster

Specifies the deletion of one node at a time from the cluster starting with the last node that is listed in the node list and node group. If the nodes are part of a GPFS cluster when this command is executed the nodes are deleted from the existing cluster before the new cluster is created.

Note: This option might not be able to delete nodes in all situations.

If the command fails, the nodes can be manually deleted using the **mmdelnode** command.

--contact-node *ContactNode*

Specifies a node name when the command is run. The contact node must be reachable over SSH without a password if that node is not member of an xCAT group.

-h | --help

Displays usage information about this command and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the **gssgencluster** command.

Examples

This example shows generation of cluster test01 using node group gss_ppc64. Here the node list for the cluster creation is obtained from the xCAT group gss_ppc64. If the gss_ppc64 contains nodes gssio1 and gssio2 then cluster will be formed using these nodes.

```
gssgencluster -C test01 -G gss_ppc64
```

The node names in the NODE-GROUP gss_ppc64 are typically used for management tasks. High speed network such an Infiniband or 10Gbit Ethernet is used for the data/clustering network. In such a case you can create the cluster as follows:

```
gssgencluster -C test01 -G gss_ppc64 --suffix=-10g
```

Here node names gssio1-10g and gssio2-10g is used for the cluster creation. Where gssio1-10g and gssio2-10g must be resolvable into a valid IP addresses in the high speed network.

```
gssgencluster -C test01 -G gss_ppc64 --suffix=-10g --accept-license --change-configuration verbsRdmaSend=no
```

Here license is accepted and verbsRdmsSend is disabled.

See also

See also the following *IBM Spectrum Scale: Command and Programming Reference* topics:

- **mmchconfig**

Location

/opt/ibm/gss/tools/bin

gssgenclusterrgs command

Creates recovery groups, NSDs, and file systems.

Synopsis

```
gssgenclusterrgs { -N NodeList | -G NodeGroup }  
[--prefix Prefix ] [ --suffix Suffix ] [ --verify Verify ]  
[ --rgversion RgVersion ] [--use-rgstanza] [--use-cur-recoverygroups]  
[--no-log-vdisk] [--verbose] [ --contact-node ContactNode ]  
[--legacy LegacyOptions] [--convert] [--resize] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The **gssgenclusterrgs** command can be used to create:

- Recovery groups
- Declustered arrays (DAs)
- Associated log vdisks for logtip, logbackup and loghome

Use this command to create data and metadata vdisks, NSDs, and file systems.

In case of hybrid enclosures, two DAs are created. One DA for HDDs and the other one for SSDs.

Note: By default, **gssgenclusterrgs** does not use mmvdisk-controlled vdisks. If you want to create mmvdisk-controlled RGs, use the **--legacy no** option. After RGs are created with mmvdisk, they cannot be converted back to legacy RGs.

gssgenclusterrgs also supports conversion of legacy RGs to mmvdisk-controlled RGs.

Restriction: As of now, **gssgenclusterrgs** does not allow custom recovery group names.

Recovery group names are generated by this command using the following logic:

Adds “rg_” at the beginning of the recovery group name followed by the prefix if provided, followed by the I/O server host name, followed by the suffix if provided. For example:

```
$ gssgenclusterrgs -N gssio1,gssio2 --prefix=su_ --suffix=-10g
```

The recovery group names generated for this command are:

- rg_su_gssio1_10g
- rg_su_gssio2_10g

If you are using “-” in a string, it is replaced by “_”.

Parameters

-N *NodeList*

Specifies a list of nodes for recovery group creation.

You must specify the **-N** or **-G** parameter.

-G *NodeGroup*

Specifies the name of the node group for recovery group creation.

You must specify the **-N** or **-G** parameter.

- prefix *Prefix***
Specifies the host name prefix. Use an equal sign (=) between **--prefix** and *Prefix* if the prefix begins with a hyphen (-).
- suffix *Suffix***
Specifies the host name suffix. Use an equal sign (=) between **--suffix** and *Suffix* if the suffix begins with a hyphen (-).
- verify *Verify***
Verifies whether the pdisk was formatted previously. Valid values are: yes, no. The default is yes.
- rgversion *RgVersion***
Specifies the RG version string. Specify a version string or LATEST.
- use-rgstanza**
Uses the stanza files provided for RG creation. The stanza files must be located in the /tmp directory of the first I/O server of the corresponding building block. The names of the stanza files must follow this naming convention: xxxxR.stanza, yyyyL.stanza.
- use-cur-recoverygroups**
Uses the current (existing) recovery groups. This is useful if you rerun the command and do not want to delete and recreate the RGs.
- no-log-vdisk**
Specifies that log vdisks are not created.
- verbose**
Provides more details.
- contact-node *ContactNode***
Specifies the name of the node where the command is run, **--contact-node gssio1**, for example.
- multi-da**
Creates multiple DAs in a recovery group.
- compat**
Provides compatibility mode of loghome vdisk.
- legacy {yes | no}**
Generates the cluster recovery groups by using **mmcrrecoverygroup**. If this parameter is set to no then recovery group is created using **mmvdisk**. As of now **--legacy** is set to yes by default for any RGs being created.
- convert**
Convert legacy RGs to mmvdisk-controlled RGs. Once converted to mmvdisk-controlled, an RG cannot be reverted back to a legacy RG.
- resize**
Support for MES deployment. Convert your GL2->GL4->GL6 etc. See mmvdisk MES documentation for more information.
- h | --help**
Displays usage information about this command and exits.

Exit status

- 0** Successful completion.
- nonzero**
A failure has occurred.

Security

You must have root authority to run the **gssgenclusterrgs** command.

Examples

- This example shows how to create recovery groups and log vdisks.

```
gssgenclusterrgs -G gss_ppc64
```

The system displays output similar to this:

```
gssgenclusterrgs -G gss_ppc64
```

```
2014-11-26T07:18:20.905224 Determining peer nodes
2014-11-26T07:18:22.339222 nodelist:          gssio1 gssio2
2014-11-26T07:18:22.339340 Getting pdisk topology from node to create partner list gssio1
2014-11-26T07:19:45.969626 Getting pdisk topology from node to create partner list gssio2
2014-11-26T07:21:07.882685 Getting pdisk topology from node for recoverygroup creation. gssio1
2014-11-26T07:22:30.264392 Getting pdisk topology from node for recoverygroup creation. gssio2
2014-11-26T07:23:59.845833 Stanza files for node pairs
gssio1 gssio2 /tmp/SV24819545L.stanza /tmp/SV24819545R.stanza

2014-11-26T07:23:59.845947 Creating recovery group gssio1
2014-11-26T07:24:28.115152 Creating recovery group gssio2
2014-11-26T07:24:58.174974 Creating log vdisks in recoverygroup gssio1
2014-11-26T07:26:00.542820 Creating log vdisks in recoverygroup gssio2
2014-11-26T07:30:42.694659 Task complete.
```

- This example shows how to create RGs without log vdisks.
gssgenclusterrgs -G gss_ppc64 --no-log-vdisk
- This example shows how to create only log vdisks if the RGs already exist.
gssgenclusterrgs -G gss_ppc64 --use-cur-recoverygroups
- This example shows how to create recovery groups using an existing stanza file.
gssgenclusterrgs -G gss_ppc64 --use-rgstanza

See also

- “gssgenvdisks command” on page 30

See also the following *IBM Spectrum Scale: Command and Programming Reference* topics:

- **mmcrfs** command

Location

```
/opt/ibm/gss/tools/bin
```

gssgennetworks command

Create a bonded ESS network

Synopsis

```
gssgennetworks { -N Node-List | -G Node-Group } [ --prefix Prefix ] [ --suffix Suffix ]  
[ --interface Interface ] [ --create-bond | --add-slave ] [ --gateway Gateway ] [ --bond Bond ]  
[ --mode { balance-rr,active-backup,balance-xor,broadcast,802.3ad,balance-tlb,balance-alb } ]  
[ --hash-policy {layer2+3,layer3+4} ] [ --netmask Crid ]  
[ --ipoib ] [ --mtu {1500,2048,4092,9000} ]  
[ --contact-node CONTACT-NODE ] [ --verbose ] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

Use **gssgennetworks** to create a high speed network by using **nmcli**. This command is applicable to any network with Ethernet and InfiniBand interfaces. This command also analyzes the `/etc/hosts` file in the EMS node and (based on the input that is provided in that file) it creates a bonded connection on the high speed interfaces that are in the target node. If new slave interfaces are added (or up) after the bond is created, they can be added to the bond.

By default, the:

- `Miimon` is set to 100
- Bonding mode is set to 802.3ad (LACP)
- `Xmit_hash_policy` is set to `layer2+3`

Any other bond options contain the default value, which includes the `lacp_rate` (the default value is `slow`). For proper network operation, the Ethernet switch and InfiniBand switch settings in the networking infrastructure must match the I/O server node interface bond settings.

Parameters

-N *Node-List*

Specifies a list of nodes to run the check.

-G *Node-Group*

Specifies the name of the node group to run the check.

--prefix *Prefix*

Specifies the hostname prefix.

Use an equal sign (=) between **--prefix** and *Prefix* if *Prefix* starts with a hyphen (-).

--suffix *Suffix*

Specifies the hostname suffix.

Use an equal sign (=) between **--suffix** and *Suffix* if *Suffix* starts with a hyphen (-).

--interface *Interface*

Specifies a list of interfaces for bonding. If the list is not provided, by default all high-speed interfaces are taken.

--create-bond

Creates a bonded interface.

--add-slave

Adds the slave interfaces to an existing bond. This is useful when more high-speed links are up or added since the bond creation.

--gateway *Gateway*

Specifies a gateway for the network.

--bond *Bond*

Specifies the name of the bond. The default is bond0.

--mode

Specifies mode for the bonded interface. The default is 802.3ab (recommended). The bonding option `xmit_hash_policy` is set to `layer2+3` when 802.3ab or `balance-xor` is selected.

--hash-policy {*layer2+3,layer3+4*}

Specifies the xmit hash policy for 802.3ad and `balanced-xor`. The default value is `layer2+3`.

| --netmask *Cidr*

| Specifies the CIDR or network mask for the interface. The default is /24. The network mask should
| be specified in the format of CIDR.

--verbose

Provides more verbose output. The default is false.

--ipoib

Provides IP over Infiniband services.

| --mtu {*1500,2048,4092,9000*}

| Provides the MTU of the bond network. For Ethernet, 1500 or 9000 MTU is allowed (Default is 1500).
| For Infiniband, 2048 or 4092 MTU is allowed (Default is 2048).

--contact-node *CONTACT-NODE*

Provides the contact node.

-h | --help

Displays usage information about this script and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the `gssgennetworks` script.

Examples

1. This example shows how to run the `gssgennetworks` to obtain information of high-speed interfaces IP address assigned to the bond:

```
[root@ems1 ~]# gssgennetworks -G gss_ppc64 --suffix=-te0 --verbose
2016-01-25T16:05:14.184397 Starting network generation
2016-01-25T16:05:15.782794 nodelist:  gssio1 gssio2
2016-01-25T16:05:15.782846 suffix used for network hostname: -te0
mlx4_0 port 1 ==> enp1s0 (Down)
mlx4_0 port 2 ==> enp1s0d1 (Down)
mlx4_1 port 1 ==> enP4p1s0 (Up)
mlx4_1 port 2 ==> enP4p1s0d1 (Down)
mlx4_2 port 1 ==> enP9p1s0 (Down)
mlx4_2 port 2 ==> enP9p1s0d1 (Down)

Interface list for node  gssio1
Down interface enp1s0
Down interface enp1s0d1
Up interface enP4p1s0
Down interface enP4p1s0d1
Down interface enP9p1s0
```

```

Down interface enP9p1s0d1
mlx4_0 port 1 ==> enp1s0 (Down)
mlx4_0 port 2 ==> enp1s0d1 (Down)
mlx4_1 port 1 ==> enP4p1s0 (Up)
mlx4_1 port 2 ==> enP4p1s0d1 (Down)
mlx4_2 port 1 ==> enP9p1s0 (Down)
mlx4_2 port 2 ==> enP9p1s0d1 (Down)

```

```

Interface list for node gssio2
Down interface enp1s0
Down interface enp1s0d1
Up interface enP4p1s0
Down interface enP4p1s0d1
Down interface enP9p1s0
Down interface enP9p1s0d1
Node: gssio1-te0, IP Address: 11.1.202.13
Node: gssio2-te0, IP Address: 11.1.202.14

```

2. This example shows how to create a bond in the I/O server node gssio1 using the information provided in the /etc/hosts file:

```

[root@ems1 bin]# gssgennetworks -N gssio1 --suffix=-te0 --create
2016-01-25T14:19:42.615008 Starting network generation
2016-01-25T14:19:44.132500 nodelist: gssio1
2016-01-25T14:19:44.132551 suffix used for network hostname: -te0
mlx4_0 port 1 ==> enp1s0 (Down)
mlx4_0 port 2 ==> enp1s0d1 (Down)
mlx4_1 port 1 ==> enP4p1s0 (Up)
mlx4_1 port 2 ==> enP4p1s0d1 (Down)
mlx4_2 port 1 ==> enP9p1s0 (Down)
mlx4_2 port 2 ==> enP9p1s0d1 (Down)

```

```

Interface list for node gssio1
Down interface enp1s0
Down interface enp1s0d1
Up interface enP4p1s0
Down interface enP4p1s0d1
Down interface enP9p1s0
Down interface enP9p1s0d1
Node: gssio1-te0, IP Address: NA
gssio1: Connection 'bond-bond0' (aaf9ff6c-2cb4-4cd8-9912-96a27da5d86c) successfully added.

```

```

[WARN] gssio1: Bond created with one slave interface
nmcli c add type bond-slave ifname enP4p1s0 master bond0
gssio1: Connection 'bond-slave-enP4p1s0' (4ce0e384-4044-4675-b6a6-51588e30efad)
successfully added.

```

```

nmcli c up bond-slave-enP4p1s0
gssio1: Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/11)

```

```

nmcli c up bond-bond0
gssio1: Connection successfully activated (master waiting for slaves)
(D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/12)

```

```

nmcli d sh bond0
gssio1: GENERAL.DEVICE: bond0
gssio1: GENERAL.TYPE: bond
gssio1: GENERAL.HWADDR: F4:52:14:FD:58:92
gssio1: GENERAL.MTU: 1500
gssio1: GENERAL.STATE: 100 (connected)
gssio1: GENERAL.CONNECTION: bond-bond0
gssio1: GENERAL.CON-PATH: /org/freedesktop/NetworkManager/ActiveConnection/12
gssio1: IP4.ADDRESS[1]: 11.1.202.13/24
gssio1: IP4.GATEWAY: 11.1.202.1

```

```
nmcli c
```

```

gssiol: NAME                UUID                TYPE                DEVICE
...
gssiol: bond-bond0         aaf9ff6c-2cb4-4cd8-9912-96a27da5d86c  bond                bond0
gssiol: enP4pls0          6068e853-30c4-44e0-82b5-658d439fd37b  802-3-ethernet      --
...

```

```

nmcli d
gssiol: DEVICE            TYPE            STATE            CONNECTION
...
gssiol: bond0             bond            connected        bond-bond0
gssiol: enP4pls0          ethernet        connected        bond-slave-enP4pls0
...

```

Bond creation compete

1. To obtain information of high speed interface IP address assigned to the bond for Ethernet interfaces, run:

```

# gssgennetworks -G gss_ppc64 --suffix=-10g --verbose
2018-02-28T01:33:51.764194 Starting network generation
2018-02-28T01:33:52.432630 nodelist:  gssiol gssio2
2018-02-28T01:33:52.432669 suffix used for network hostname: -10g
Interface(s) available on node gssiol-10g
Up interface:  enP2pls0
Up interface:  enP2pls0d1
Up interface:  ib0
Down interface:  ib1
Up interface(s) of node gssiol-10g considered for bonding are ['enP2pls0', 'enP2pls0d1']
Interface(s) available on node gssio2-10g
Up interface:  enP2pls0
Up interface:  enP2pls0d1
Up interface:  ib0
Down interface:  ib1
Up interface(s) of node gssio2-10g considered for bonding are ['enP2pls0', 'enP2pls0d1']
gssiol-10g: Current IP Address: IP not assigned
gssio2-10g: Current IP Address: IP not assigned

```

2. To create a bond (for Ethernet interfaces where the bond name is bond0) in an I/O server node by using information that is provided in the /etc/hosts file:

```

# gssgennetworks -G gss_ppc64 --suffix=-10g --bond bond0 --create-bond
2018-02-28T01:41:11.413341 Starting network generation
2018-02-28T01:41:12.072727 nodelist:  gssiol gssio2
2018-02-28T01:41:12.072765 suffix used for network hostname: -10g
Interface(s) available on node gssiol-10g
Up interface:  enP2pls0
Up interface:  enP2pls0d1
Up interface:  ib0
Down interface:  ib1
Up interface(s) of node gssiol-10g considered for bonding are ['enP2pls0', 'enP2pls0d1']
Interface(s) available on node gssio2-10g
Up interface:  enP2pls0
Up interface:  enP2pls0d1
Up interface:  ib0
Down interface:  ib1
Up interface(s) of node gssio2-10g considered for bonding are ['enP2pls0', 'enP2pls0d1']
gssiol-10g: Current IP Address: IP not assigned
gssio2-10g: Current IP Address: IP not assigned
gssiol-10g: IP Address assigned to bond: 172.31.250.17
gssio2-10g: IP Address assigned to bond: 172.31.250.18
Bond creation complete

```

3. To obtain information of high speed interface IP address that are assigned to the bond for Infiniband interfaces, run:

```

# gssgennetworks -G gss_ppc64 --suffix=-ib --ipoib
2018-02-28T01:46:52.785492 Starting network generation
2018-02-28T01:46:53.452440 nodelist:  gssiol gssio2
2018-02-28T01:46:53.452475 suffix used for network hostname: -ib
Interface(s) available on node gssiol-ib

```



```

Up interface: enP2pls0
Up interface: enP2pls0d1
Up interface: ib0
Down interface: ib1
Up interface(s) of node gssio1-ib considered for bonding are ['ib0']
Interface(s) available on node gssio2-ib
Up interface: enP2pls0
Up interface: enP2pls0d1
Up interface: ib0
Down interface: ib1
Up interface(s) of node gssio2-ib considered for bonding are ['ib0']
gssio1-ib: Current IP Address: IP not assigned
gssio2-ib: Current IP Address: IP not assigned

```

4. To create a bond (for Ethernet interfaces where the bond name is bond1) in an I/O server node by using information that is provided in the /etc/hosts file:

```

# gssgennetworks -G gss_ppc64 --suffix=-ib --ipoib --create-bond
2018-02-28T01:48:31.769301 Starting network generation
2018-02-28T01:48:32.432582 nodelist: gssio1 gssio2
2018-02-28T01:48:32.432633 suffix used for network hostname: -ib
Interface(s) available on node gssio1-ib
Up interface: enP2pls0
Up interface: enP2pls0d1
Up interface: ib0
Down interface: ib1
Up interface(s) of node gssio1-ib considered for bonding are ['ib0']
Interface(s) available on node gssio2-ib
Up interface: enP2pls0
Up interface: enP2pls0d1
Up interface: ib0
Down interface: ib1
Up interface(s) of node gssio2-ib considered for bonding are ['ib0']
gssio1-ib: Current IP Address: IP not assigned
gssio2-ib: Current IP Address: IP not assigned
gssio1-ib: IP Address assigned to bond: 172.31.250.1
[WARN] gssio1-ib: Bond created with one slave interface
gssio2-ib: IP Address assigned to bond: 172.31.250.2
[WARN] gssio2-ib: Bond created with one slave interface
Bond creation complete

```

Location

/opt/ibm/gss/tools/bin

gssgenvdisks command

Generates vdisk stanza files and creates vdisks, NSDs, and file systems.

Synopsis

```
gssgenvdisks [ --contact-node ContactNode ] [ --recovery-group RecoveryGroup ]  
[ --create-stanza-only ] [ --vdisk-stanza VdiskStanza ] [ --vdisk-suffix VdiskSuffix ]  
[ --create-vdisk] [ --create-nsds] [ --create-filesystem] [ --filesystem-name Device ]  
[ --filesystem-mount MountPoint ] [ --filesystem-options Options ] [ --system SystemPool ]  
[ --use-only-da USE-DA ] [ --num-of-metadata-nsds NumOfMetadataNsds ] [ --num-of-nsds NumOfNsds ]  
[ --num-of-data-nsds NumOfDataNsds ] [ --metadata-vdisk-size MetadataVdiskSize ]  
[ --data-vdisk-size DataVdiskSize ] [ --vdisk-size VdiskSize ]  
[ --data-blocksize DataBlockSize ] [ --blocksize BlockSize ]  
[ --metadata-blocksize MetadataBlockSize ] [ --metadata-percent MetadataPercent ]  
[ --reserved-space-percent ReservedSpacePercent ] [ --raid-code RaidCode ]  
[ --failure-group FailureGroup ] [ --reserved-space-percent ReservedSpacePercent ] [ --crcesfs ]  
[ --verbose ] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The **gssgenvdisks** command generates vdisk stanza files and creates NSDs, and file systems.

gssgenvdisks provides a default placement policy in case hybrid enclosures are used. In case of hybrid enclosures, any data vdisk goes to a DA which is composed of HDDs and any metadata vdisk goes to a DA which is composed of SDDs.

The **gssgenvdisks** command can detect the hybrid enclosures.

You can override the default vdisk placement policy used in case of a hybrid enclosure system, by using the **--use-only-da** option. The **--use-only-da** option allows data and metadata vdisks on only the specified DA. It ignores other DAs available in the recovery group.

Parameters

-h | **--help**

Displays usage information about this command and exits.

--contact-node *ContactNode*

Specifies the name of the node where the command is run (for example: **--contact-node gssio1**).

Ensure that the node name is reachable by the system

--recovery-group *RecoveryGroup*

Specifies a list of recovery groups. If specific recovery groups are not provided all recovery groups are used to generate vdisk stanza.

--create-stanza-only

Creates only the stanza file in the /tmp folder using the name specified in *VdiskStanza* without creating a vdisk, nsd, or filesystem.

--vdisk-stanza *VdiskStanza*

Specifies the path name of the vdisk stanza file to create. The default is /tmp/vdisk1.cfg. The vdisk stanza file is saved in the contact node when provided.

--vdisk-suffix *VdiskSuffix*

Specifies the suffix for the vdisk names. The valid characters that can be used in the suffix are: a to z, A to Z, 0 to 9, and _ (underscore).

- create-vdisk**
Creates the vdisks. Without this option, only the vdisk stanza is created.
- create-nsds**
Creates NSDs from vdisks.
- create-filesystem**
Creates a filesystem using NSDs.
- filesystem-name *Device***
Specifies the file system name. The default is gpfs0.
- filesystem-mount *MountPoint***
Specifies the file system mount point. The default is /gpfs.
- filesystem-options *Options***
Specifies other file system creation options. The default is None. Enclose with quotation marks (' or ") if there are blank spaces within the options.

See the **mmcrfs** command description in the *IBM Spectrum Scale: Command and Programming Reference* for valid file system creation options.
- system *SystemPool* {yes | no}**
Creates the vdisk to store data and metadata in the same vdisk under the system pool.
- use-only-da *USE-DA***
Provides the DA name that must be considered while creating vdisks.
- num-of-metadata-nsds *NumOfMetadataNsds***
Specifies the number of metadata NSDs per DA. The default values are:
- 1 for GS1, GS2, and GL2 systems
 - 2 for GS4 and GL4 systems
 - 3 for GS6 and GL6 systems
- The space available for the metadata is equally divided between the metadata NSDs.
- num-of-data-nsds *NumOfDataNsds***
Specifies the number of data NSDs per DA. The default values are:
- 1 for GS1, GS2, and GL2 systems
 - 2 for GS4 and GL4 systems
 - 3 for GS6 and GL6 systems
- The space available for the data is equally divided between the data NSDs.
- num-of-nsds *NumOfNsds***
- metadata-vdisk-size *MetadataVdiskSize***
Specifies the size of the metadata vdisks in GiB.

When specified, this option is used (instead of the **--metadata-percent** option) to calculate the vdisk size. If no data vdisks are being configured (the value of *NumberOfDataNsds* is 0), *DataVdiskSize* should be set to a non-zero number (for example: 1000) to set the metadata vdisk size correctly. Otherwise, the metadata vdisk size is set to 0 in the vdisk stanza file. To work around this, a non-zero *DataVdiskSize* should be provided (with a *NumberOfDataNsds* value of 0) when metadata vdisks only are configured.
- data-vdisk-size *DataVdiskSize***
Specifies the size of the data vdisks in GiB. If **--data-vdisk-size** is zero or not provided, the vdisk size is determined from the available space after removing the reserved space.
- vdisk-size *VdiskSize***

--metadata-blocksize *MetadataBlockSize*

Specifies the block size of the metadata NSDs. The default is 1M.

--data-blocksize *DataBlockSize*

Specifies the block size of the data NSDs. The default is 8M.

--blocksize *BlockSize*

Specifies the block size of data and metadata NSD. The default is 16M.

--metadata-percent *MetadataPercent*

Specifies the metadata NSD capacity as a percentage of the overall usable capacity. The default is 5. The estimate is based on usable capacity (after redundancy overhead, for example: 8+2p for data and 3wayreplicated for metadata). When **metadata-vdisk-size** is provided this option is ignored.

--raid-code *RaidCode*

Specifies the RAID code. The default is 8+2p. Acceptable raid-codes are 8+2p and 8+3p. The 3wayreplicated code is used for metadata vdisks. The raid code for data vdisks is 8+2p. The 4wayreplicated code is used for metadata vdisk. The raid code for data vdisks is 8+3p.

--failure-group *FailureGroup*

Specifies the NSD base failure group. All NSDs in a building block are provided with the same failure group. If two or more building blocks are present, NSDs in each building block are assigned increasing failure group numbers, starting with the base failure group number. The default base failure group is 30.

--reserved-space-percent *ReservedSpacePercent*

Specifies the percentage of total space to reserve. The default is 1.

--crcesfs

Creates the filesystem for the Cluster Export Services (CES) shared root.

--verbose

Specifies the more details.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the **gssgenvdisks** command.

Examples

This example shows **gssgenvdisks** command creating vdisks, nsds and filesystem using all recoverygroups in the cluster. Run:

```
gssgenvdisks --create-vdisk --create-filesystem --raid-code 8+3p
```

The system displays output similar to this:

```
2015-03-07T06:03:42.415436 Start creating vdisk stanza
2015-03-07T06:03:42.415517 No contact node provided. Using current node.  ems1
vdisk stanza saves in ems1:/tmp/vdisk1.cfg
2015-03-07T06:04:02.907108 Generating vdisks for nsd creation
2015-03-07T06:06:15.762349 Creating nsds
2015-03-07T06:06:28.904591 Creating filesystem
Filesystem successfully created. Verify failure group of nsds and change as needed.
2015-03-07T06:06:44.765517 Applying data placement policy
2015-03-07T06:06:50.135838 Task complete.
```

See also

See also the following *IBM Spectrum Scale: Command and Programming Reference* topics:

- **mmcrfs** command

Location

/opt/ibm/gss/tools/bin

gssinstallcheck command

Performs ESS install check.

Synopsis

```
gssinstallcheck { -N NODE-LIST | -G NODE-GROUP } [ --prefix PREFIX ]  
[ --get-version ] [ --suffix SUFFIX ] [ --syslog ] [ --phy-mapping ] [ --srv-events ]  
[ --hmc-username HSCUSERNAME ] [ --platform-events EVENTLIST ]  
[ --close-platform-events CLOSEEVENTLIST ] [ --net-errors ] [ --errthld ERROR-THRESHOLD ]  
[ --dropthld DROP-THRESHOLD ] [ --monitor ] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The `gssinstallcheck` command checks various aspects of the installation.

Parameters

-N *NODE-LIST*

Provides a list of nodes to run the check.

-G *NODE-GROUP*

Provides a name of the node group to run the check.

--prefix *PREFIX*

Provides a host name prefix. Use = between --suffix and value if the value starts with -.

--get-version

Provides the ESS package version.

--suffix *SUFFIX*

Provides the host name suffix. Use = between --suffix and value if the value starts with -.

--syslog

Logs output to syslog (/var/log/messages). Default no logging to syslog.

--phy-mapping

Check only phy mapping. When selected only this option is run.

--srv-events

Shows the serviceable events. When selected, only this option is run.

--hmc-username *HSCUSERNAME*

Provides the HMC root user name for PPC BE deployments.

--platform-events *EVENTLIST*

Provides platform event details.

--close-platform-events *CLOSEEVENTLIST*

Closes the provided platform event details.

--net-errors

Checks for the network error counts. When selected, only this option is run.

--errthld *ERROR-THRESHOLD*

Provides packet error threshold in percent during net-errors check.

--dropthld *DROP-THRESHOLD*

Provides packet drop threshold in percent during net-errors check.

--monitor

In this mode, outputs are only logged in to syslog. The stdout is turned off.

-h | --help

Displays usage information about this command and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the **gssinstallcheck** command.

Examples

1. This example shows running install check to obtain installed package version.

```
# gssinstallcheck -G ems1,gss_ppc64 --get-version
```

A sample output is as follows:

Start of install check

```
nodelist:      ems1 gssio1 gssio2
```

```
Node: ems1          Installed version:      5.3.0-20180221T135444Z_ppc64_datamanagement
Node: gssio1        Installed version:      5.3.0-20180221T142513Z_ppc641e_datamanagement
Node: gssio2        Installed version:      5.3.0-20180221T142513Z_ppc641e_datamanagement
```

2. This example shows running install check on an I/O server node in the **gss_ppc64** group.

```
# gssinstallcheck -G gssio2
```

A sample output is as follows:

```
===== Summary of node: gssio2 =====
Getting system firmware level. May take a long time...
Getting system profile setting.

Installed version:      5.3.0-20180221T142513Z_ppc641e_datamanagement
[OK] Linux kernel installed: 3.10.0-514.41.1.el7.ppc64le
[OK] Systemd installed: 219-42.el7_4.7.ppc64le
[OK] Networkmgr installed: 1.8.0-11.el7_4.ppc64le
[OK] OFED level:      MLNX_OFED_LINUX-4.1-4.1.6.1
[OK] IPR SAS FW:      17518300
[OK] ipraid RAID level: 10
[OK] ipraid RAID Status: Healthy
[OK] IPR SAS queue depth: 64
[OK] System Firmware : FW860.42 (SV860_138)
[OK] System profile setting: scale
[OK] Host adapter driver: 13.100.00.00
[OK] Kernel sysrq level is: kernel.sysrq = 1
Spectrum Scale RAID is not active, can not get gpfs version
Spectrum Scale RAID is not active. Can not check GNR configuration data
Performing Spectrum Scale RAID configuration check.
ems nodeclass not found. Parameter checking as IO Server node.
Spectrum Scale RAID is not active. Configuration settings can not be checked
[ERROR] Can not obtain GNR configuration parameters
[ERROR] Can not obtain GNR callback settings
[ERROR] GNR Callback is not set correctly
[OK] New disk prep script: /usr/lpp/mmfs/bin/tspreparenewpdiskforuse
[OK] Network adapter MT4115 firmware: 12.20.1010, net adapter count: 2
[OK] Network adapter MT26448 firmware: 2.9.1326, net adapter count: 1
Obtaining storage firmware versions from IO nodes. May take a long time...
```

```
Can not obtain firmware level of storage elements
[ERROR] Unsupported or incorrect firmware found in storage subsystem
Phy mapping check skipped for LE systems
End of install check
```

See also

- “gssdeploy script” on page 125
- “gssinstall script” on page 130

Location

/opt/ibm/gss/tools/bin

gssnettest command

Performs ESS Network test.

Synopsis

```
gssnettest { -N NODE-LIST | -G NODE-GROUP } [ --prefix PREFIX ] [ --suffix SUFFIX ]  
[ --contact-node CONTACT-NODE ] [ --duration TEST-DURATION ] [ --buffersize BUFFER-SIZE ] [ --errthld  
ERROR-THRESHOLD ] [ --dropthld DROP-THRESHOLD ] [ --syslog ] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The **gssnettest** script helps in running the network workload to test network for proper operations. The **nsdperf** workload generator is the underlying tool to generate the workload.

Parameters

-N *NODE-LIST*

Provides a list of nodes to run the network performance test.

-G *NODE-GROUP*

Provides the name of node group.

--prefix *PREFIX*

Provides the host name prefix. Use = between --prefix and value if the value starts with -.

--suffix *SUFFIX*

Provides the host name suffix. Use = between --suffix and value if the value starts with -.

--contact-node *CONTACT-NODE*

Provides the contact node.

--duration *TEST-DURATION*

Provides the test run time in seconds. Default test duration is 30 seconds.

--buffersize *BUFFER-SIZE*

Provides the buffer size in bytes. Default is 4194304 (4 MiB).

--errthld *ERROR-THRESHOLD*

Provides the packet error threshold in percent.

--dropthld *DROP-THRESHOLD*

Provides the packet drop threshold in percent.

--syslog

Logs the packet error and drop percent to syslog (/var/log/messages). Default no logging to syslog.

-h | **--help**

Displays usage information about this script and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the **gssnettest** script.

Example

This example shows running the network test on the management server node (EMS) and I/O server nodes.

```
GSENV=TEST gssnettest -N ems1,gss_ppc64 --suffix=-hs
```

Location

```
/opt/ibm/gss/tools/bin
```

gssnodedetails command

Obtains node details.

Synopsis

```
gssnodedetails { -N NODE-LIST | -G NODE-GROUP }  
                [ --prefix Prefix ] [ --suffix Suffix ]  
                [--fspdiscover][--identify] [--iprange IPRANGE]  
                [--pass PASSWORD] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The **gssnodedetails** command can be used to obtain node details.

Parameters

- N *NODE-LIST***
Specifies a comma separated list of nodes to get the node details.
- G *NODE-GROUP***
Specifies a node group.
- prefix *Prefix***
Provides a host name prefix. Use = between --prefix and value if the value starts with a -.
- suffix *Suffix***
Provides a host name suffix. Use = between --suffix and value if the value starts with -.
- fspdiscover**
Specifies to discover nodes on the FSP network.
- identify**
Specifies to identify the node.
- iprange *IPRANGE***
Provides the IP range use to discover nodes.
- pass *PASSWORD***
Specifies the IPMI password of FSP. Default password is used when not provided.
- h | --help**
Displays usage information about this command and exits.

Exit status

- 0** Successful completion.
- nonzero**
A failure has occurred.

Security

You must have root authority to run the **gssstress** command.

Example

This example shows how to obtain the node details of the `gssi01` node.

```
# gssnodedetails -N gssi01
```

A sample output is as follows:

2018-02-28T03:30:07.688876 Generating node list for the test...

```
2018-02-28T03:30:08.352377 nodelist:  gssiol
Node type   : 8247-22L
Node serial  : 2159F3A
  Static hostname: gssiol.gpfs.net
    Icon name: computer
    Machine ID: 213721d94084446a8b7e5f7bdf2ad1e
    Boot ID: cf8f091260ac4a9d92a48ca7f9f730be
  Operating System: Red Hat Enterprise Linux Server 7.4 (Maipo)
    CPE OS Name: cpe:/o:redhat:enterprise_linux:7.4:GA:server
    Kernel: Linux 3.10.0-693.33.1.el7.ppc64le
    Architecture: ppc64-le
```

IP Address:

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
   inet 127.0.0.1/8 scope host lo
   --
2: enP3p9s0f0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP qlen 1000
   inet 192.168.45.21/24 brd 192.168.45.255 scope global enP3p9s0f0
Kernel       : Linux gssiol.gpfs.net 3.10.0-693.33.1.el7.ppc64le
#1 SMP Wed Jan 17 10:35:28 EST 2018 ppc64le ppc64le ppc64le GNU/Linux
```

```
OFED          : MLNX_OFED_LINUX-4.1-4.1.6.1:
gssiol: PCI Adapters
0001:05:00.0 Serial Attached SCSI controller:
LSI Logic / Symbios Logic SAS2308 PCI-Express Fusion-MPT SAS-2 (rev 05)

0001:0b:00.0 Serial Attached SCSI controller:
LSI Logic / Symbios Logic SAS2308 PCI-Express Fusion-MPT SAS-2 (rev 05)

0001:0c:00.0 RAID bus controller: IBM PCI-E IPR SAS Adapter (ASIC) (rev 02)

0005:03:00.0 Serial Attached SCSI controller:
LSI Logic / Symbios Logic SAS2308 PCI-Express Fusion-MPT SAS-2 (rev 05)

0005:09:00.0 Serial Attached SCSI controller:
LSI Logic / Symbios Logic SAS2308 PCI-Express Fusion-MPT SAS-2 (rev 05)

0006:03:00.0 Serial Attached SCSI controller:
LSI Logic / Symbios Logic SAS2308 PCI-Express Fusion-MPT SAS-2 (rev 05)
```

```
0006:09:00.0 Serial Attached SCSI controller:
LSI Logic / Symbios Logic SAS2308 PCI-Express Fusion-MPT SAS-2 (rev 05)
gssiol: Memory
          total        used         free      shared  buff/cache   available
Mem:      123G          2.9G         119G         26M         1.0G         119G
Swap:      7.8G           0B          7.8G
```

```
gssiol: Highspeed Interface
mlx4_0 port 1 ==> enP2p1s0 (Up)
mlx4_0 port 2 ==> enP2p1s0d1 (Up)
mlx5_0 port 1 ==> ib0 (Up)
mlx5_1 port 1 ==> ib1 (Down)
```

Kernel Interface table

Iface	MTU	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
enP2p1s0	1500	0	0	0	0	0	0	0	0	BMRU
enP2p1s0	1500	0	0	0	0	0	0	0	0	BMRU
enP3p9s0	1500	595	0	0	0	4884	0	0	0	BMRU
enP3p9s0	1500	0	0	0	0	0	0	0	0	BMU
enP3p9s0	1500	0	0	0	0	0	0	0	0	BMU
enP3p9s0	1500	0	0	0	0	0	0	0	0	BMU
ib0	2044	129	0	0	0	0	0	0	0	BMRU
ib1	4092	0	0	0	0	0	0	0	0	BMU
lo	65536	4	0	0	0	4	0	0	0	LRU

gssiol: IP RAID Adapters Status

Name	PCI/SCSI Location	Description	Status
sda	0001:0c:00.0/0:2:0:0	RAID 10 Array	Optimized

Location

/opt/ibm/gss/tools/bin

gssprecheck command

Performs the ESS install or upgrade precheck.

Synopsis

```
gssprecheck {-N NODE-LIST | -G NODE-GROUP} {--install | --upgrade}  
{--file CONFIG_FILE} [--syslog] [--monitor] [--verbose] [--pre] [-h | --help]
```

Availability

Available with the Elastic Storage Server.

Description

The **gssprecheck** command checks for common errors upon new installs or upgrades.

Parameters

- N *NODE-LIST***
Provides a list of node to run the check.
- G *NODE-GROUP***
Provides the name of the node group to run the check.
- file *CONFIG_FILE***
Provides the location of `gssdeploy.cfg` file for parsing.
- install**
Prechecks for install related items only.
- upgrade**
Prechecks for upgrade related items only.
- syslog**
Logs the output to syslog (`/var/log/messages`). Default no logging to syslog.
- monitor**
In this mode outputs are only logged in to syslog. The `stdout` is turned off.
- verbose**
Indicates verbose mode.
- pre**
Does the initial check prior to running the **gssdeploy -x**.
- h --help**
Displays usage information about this command and exits.

Exit status

- 0** Successful completion.
- nonzero**
A failure has occurred.

Security

You must have root authority to run the **gssprecheck** command.

Examples

This example shows running precheck check on the management server node (EMS) prior to running the **gssdeploy -d** command:

```
[root@ems1 precheck]# ./gssprecheck -N ems1 --install --file /var/tmp/gssdeploy.cfg
```

A sample output is as follows:

```
2016-12-02T10:09:37.252363 >>>ESS500_BETA_3<<< Start of pre-install check
2016-12-02T10:09:37.252413 This may take a few minutes. Please be patient

===== Summary of EMS node =====
[OK] Parsing configuraton file
[OK] Checking xCAT version
[OK] Checking xCAT site table
[OK] Checking for general repo errors
[OK] Checking for enabled external subscriptions
[OK] Checking kernel repo
[ERROR] DHCP server check
    >>>[HINT] Check dhcp is running and the leases file exists.<<<
    >>>[HINT] Run makedhcp -q <group> to make sure mac->IP are setup.<<<
[OK] Checking FSP iface
[OK] DNS server check
[OK] /etc/hosts advanced checks
[OK] Manifest check
[OK] Checking /etc/hosts exists
[OK] Checking deploy iface
[OK] Checking correct redhat version
[OK] Checking correct Endian type
[OK] No high CPU % processes found
[OK] Root FS space check
[OK] tmp FS space check
[OK] Var FS space check
[OK] var log FS space check
[OK] Timezone consistency check

2016-12-02T10:10:11.541241 Checking nodes.

2016-12-02T10:10:11.541301 >>>ESS500_BETA_3<<< End of pre-install check
```

This example shows running precheck prior to upgrading an ESS cluster

```
[root@ems1 precheck]# ./gssprecheck -N ems1 --upgrade --file /var/tmp/gssdeploy.cfg
2016-12-02T10:11:38.504845 >>>ESS500_BETA_3<<< Start of pre-install check
2016-12-02T10:11:38.504903 This may take a few minutes. Please be patient
2016-12-02T10:11:39.430609 nodelist:      ems1

===== Summary of EMS node =====
[OK] Parsing configuraton file
[OK] Checking for heavy mm commands
[OK] Checking xCAT version
[OK] Checking xCAT site table
[OK] Checking xdsh connectivity
[ERROR] Bonded link check
    >>>[HINT] One or more network bond links down. Run...<<<
    >>>[HINT] Run cat /proc/net/bonding/bond0 | grep MII on each node and fix<<<
[OK] Spectrum Scale lock check
[OK] Checking deploy iface
[OK] Timezone consistency check
[OK] Universal time consistency check
[OK] Quorum node check
[OK] long waiters check
[ERROR] mmhealth health check
    >>>[HINT] Run mmhealth node show -N all and investigate.<<<
[ERROR] mmhealth eventlog check
    >>>[HINT] Run mmhealth node eventlog and investigate.<<<
```

```
[ERROR] resolv.conf valid and matches all nodes
>>>[HINT] Make sure each node in the Building Block have /etc/resolv.conf<<<
>>>[HINT] and the nameserver points back to the EMS mgt IP<<<
[OK] DNS server check
>>Running gnrhealthcheck...This will take a few moments<<
[ERROR] GNR health check
>>>[HINT] GNR health check detected errors Investigate before proceeding.<<<
[OK] Manifest check
[OK] Checking FSP iface
[OK] Checking /etc/hosts exists
[OK] /etc/hosts same on all nodes
[OK] /etc/hosts advanced checks
[OK] Checking for general repo errors
[OK] Checking for enabled external subscriptions
[OK] Checking kernel repo
[OK] Checking correct redhat version
[OK] Checking correct Endian type
[OK] No high CPU % processes found
[OK] Root FS space check
[OK] tmp FS space check
[OK] Var FS space check
[OK] var log FS space check
[OK] Checking that tracing is disabled
[OK] Active Node Check
[OK] Checking for deadlocks

2016-12-02T10:13:31.499906 Checking nodes.

2016-12-02T10:13:31.499972 >>>ESS500_BETA_3<<< End of pre-install check
```

See also

See also the following *Deploying the Elastic Storage Server* information:

- *Elastic Storage Server: Quick Deployment Guide.*

Location

/opt/ibm/gss/tools/bin

gssruntask command

Runs specific tasks on a node.

Synopsis

```
gssruntask { -N NodeList | -G NodeGroup }  
            [ --prefix Prefix ] [ --suffix Suffix ]  
            --task Task [ --mode Mode ]  
            [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The **gssruntask** command can be used to run specific tasks on a node.

Table 2. Supported tasks for gssruntask

Task ID	Task
scsi-sg	lsscsi -sg
scsi-disk	lsscsi grep disk
mmlsenclosure-all	mmlsenclosure all
mmlsfirmware-driv	mmlsfirmware --type drive
ping	ping -c 3
mmlsenclosure-all-L	mmlsenclosure all -L
mmlsenclosure-all-not-ok	mmlsenclosure all --not-ok
ssh	ssh
mmlspdisk-not-ok	mmlspdisk all --not-ok
scsi	lsscsi
mmlsfirmware-adapt	mmlsfirmware --type host-adapter
mmlsfirmware-encl	mmlsfirmware --type storage-enclosure
scsi-encl	lsscsi grep enclosure

Parameters

-N *NodeList*

Specifies a comma separated list of nodes to get the node details.

-G *NodeGroup*

Specifies a node group.

--prefix *Prefix*

Provides a host name prefix. Use = between --prefix and value if the value starts with a -.

--suffix *Suffix*

Provides a host name suffix. Use = between --suffix and value if the value starts with -.

--task *Task*

Provides the task with options to run on the *NodeList* or *NodeGroup*.

--mode *Mode*

Provides the run mode of the task. The default is local.

-h | --help

Displays usage information about this command and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the **gssstress** command.

Example

This example shows how to run the **scsi** task on the **gssiol** node.

```
# gssruntask -N gssiol --task scsi-sg
```

A sample output is as follows:

```
2018-02-28T03:35:35.540298 Generating node list for the test...
```

```
2018-02-28T03:35:36.202471 nodelist: gssiol
```

```
Running task lsscsi -sg node gssiol
```

```
Success: Running task lsscsi -sg on node gssiol
```

```
gssiol: [0:0:0:0] storage IBM AL14SEB060N 6506 - /dev/sg1 -
gssiol: [0:0:1:0] storage IBM AL14SEB060N 6506 - /dev/sg2 -
gssiol: [0:0:2:0] enclosu IBM VSBPD14M1 6GSAS 03 - /dev/sg5 -
gssiol: [0:0:3:0] cd/dvd IBM. RMB00140572 IP02 /dev/sr0 /dev/sg6 -
gssiol: [0:0:4:0] enclosu IBM PSBPD14M1 6GSAS 1508 - /dev/sg7 -
gssiol: [0:0:5:0] enclosu IBM PSBPD14M1 6GSAS 1508 - /dev/sg8 -
gssiol: [0:1:0:0] no dev IBM IPR-10 68C38500 - /dev/sg3 -
gssiol: [0:2:0:0] disk IBM IPR-10 68C38500 /dev/sda /dev/sg4 571GB
gssiol: [0:3:0:0] no dev IBM 57DC001SISIOA 0150 - /dev/sg0 -
gssiol: [2:0:0:0] disk IBM MTFDJAK400MBS 502E /dev/sdb /dev/sg9 400GB
gssiol: [2:0:1:0] disk IBM MTFDJAK400MBS 502E /dev/sdc /dev/sg10 400GB
gssiol: [2:0:2:0] disk IBM MTFDJAK400MBS 502E /dev/sdd /dev/sg11 400GB
gssiol: [2:0:3:0] disk IBM MTFDJAK400MBS 502E /dev/sde /dev/sg12 400GB
gssiol: [2:0:4:0] disk IBM MTFDJAK400MBS 502E /dev/sdf /dev/sg13 400GB
....
....
```

Location

/opt/ibm/gss/tools/bin

gssstoragequickcheck command

Quickly checks the attached configuration.

Synopsis

```
gssstoragequickcheck { -N NodeList | -G NodeGroup }  
                    [ --component ComponentList ] [ --prefix Prefix ] [ --suffix Suffix ]  
                    [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The `gssstoragequickcheck` command is used to perform a high level check of the storage connectivity and configuration. It is run from the management server node (EMS).

Parameters

-N *NodeList*

Specifies a comma separated list of nodes.

-G *NodeGroup*

Specifies a xCAT node group name. Either a *NodeList* or a *NodeGroup* must be provided.

--component *ComponentList*

Provides a component list to specify the scope of check. The options are as follows:

- server - to check I/O server node
- adapter - to check installed network and storage adapters
- storage - to check attached enclosures and disks

--prefix *Prefix*

Provides a host name prefix. *Prefix* is used with *NodeList* to generate node names where the check is run. Use = between --prefix and value if the value starts with a -.

--suffix *Suffix*

Provides a host name suffix. *Suffix* is used with *NodeList* to generate node names where the check is run. For example, with a node list of `gssio1,gssio2` and prefix `A-` and suffix `-ib`, node names `A-gssio1-ib` and `A-gssio2-ib` will be used to run the test. The newly formed node name must be resolvable to the corresponding IP address. Use = between --suffix and value if the value starts with -.

-h | --help

Displays usage information about this command and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the `gssstoragequickcheck` command.

Example

This example shows **gssstoragequickcheck** running on nodes defined in the node group `gss_ppc64`.
`gssstoragequickcheck -G gss_ppc64`

This example shows **gssstoragequickcheck** running on the node `gssio1` and for the SCSI component.
`gssstoragequickcheck -N gssio1 --comp scsi`

A sample output is as follows:

```
2014-12-02T17:33:15.826648 Start of storage quick configuration check
 2014-12-02T17:33:17.518018 nodelist:  gssio1 gssio2

gssio1: Machine Type: 8247-22L
gssio2: Machine Type: 8247-22L
gssio1: Valid SAS Adapter Configuration. Number of Adapter(s) found 3
gssio1: Valid Network Adapter Configuration. Number of Adapter(s) found: 3
gssio2: Valid SAS Adapter Configuration. Number of Adapter(s) found 3
gssio2: Valid Network Adapter Configuration. Number of Adapter(s) found: 3
gssio1: Enclosure DCS3700 found 2
gssio1: Disk ST2000NM0023 found 116
gssio1: Total disk found 116, expected 116
gssio1: SSD SG9XCA2G200GEIBM found 2
gssio1: Total SSD found 2, expected 2
gssio2: Enclosure DCS3700 found 2
gssio2: Disk ST2000NM0023 found 116
gssio2: Total disk found 116, expected 116
gssio2: SSD SG9XCA2G200GEIBM found 2
gssio2: Total SSD found 2, expected 2

2014-12-02T17:33:26.985323 End of storage quick configuration check
```

Location

`/opt/ibm/gss/tools/bin`

gsssnap script

Collects the snapshot of the ESS deployment

Synopsis

```
gsssnap [ { -N | --nodes } NodeList ] [ -g | --gpfs ] [ -V | --version ]  
[ -i | --installcheck ] [ -t | --tarch ] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The **gsssnap** script creates an informational system snapshot at a single point in time. This system snapshot consists of cluster configuration, disk configuration, network configuration, network status, ESS logs, dumps, and traces. Use the **gsssnap** script as one of the main tools to gather preliminary data when an ESS problem is encountered, such as a hung ESS script. The information that is gathered with the **gsssnap** script can be used in conjunction with other information (for example, ESS internal dumps, traces, and kernel thread dumps) to solve an ESS problem.

By default, the **gsssnap** script collects snapshot information from the management server node.

Parameters

- g | --gpfs**
Takes a GPFS snapshot.
- i | --installcheck**
Capture the **gssinstallcheck** output.
- t | --tarch**
Specifies the target architecture of deployment. ppc64 or ppc64le.
- { -N | --nodes } *NodeList***
Specifies a comma-separated list of nodes from which to collect snapshot information.
- V, --version**
Displays the program's version number and exits.
- h | --help**
Displays usage information about this script and exits.

Exit status

- 0** Successful completion.
- nonzero**
A failure has occurred.

Security

You must have root authority to run the **gsssnap** script.

Example

This example shows how to collect the snapshot information from I/O server node `gssi01`:

```
gsssnap -N gssi01
```

The system displays output similar to this:

```
# gsssnap -N gssio1,gssio2

gsssnap [INFO]: Collecting xcat snap
gsssnap [INFO]: Collecting sosreports for node(s): ems1,gssio1,gssio2
gsssnap [INFO]: Collecting GSS snap
#####
gsssnap tar file: /tmp/gsssnap.20151013T213843Z.tgz
MD5 file: /tmp/gsssnap.20151013T213843Z.tgz.md5
Please provide tar file to IBM service
#####
```

Location

/opt/ibm/gss/xcat/bin/gsssnap

gssstress command

Runs a stress test on the file system with different options.

Synopsis

```
gssstress [ -t Threads ] [ -i Iterations ] [ -b BlockSize ]  
           [ -s FileSize ] [ -o OperationList ] [ -p IoPattern ] [ -f FileName ]  
           TargetPath Node1 [Node2...NodeN] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The **gssstress** command can be invoked from the management server node (EMS) to run stress tests on the file system with different options.

Parameters

-t *NumThreads*

Specifies the number of simultaneous threads. The default is 16.

-i *Iterations*

Specifies the number of iterations of read-write operations to be performed on the file system. The default is 20.

-b *BlockSize*

Specifies the block size to be used for the stress test. The default is 16M.

-s *FileSize*

Specifies the size of the file to be used for the stress test. The default is 200G.

-o *OperationList*

Specifies the I/O operation type to be performed while running the stress test. Valid values are: create, read, write, read. The default is create.

-p *IoPattern*

Specifies the I/O pattern to be used for the stress test. Valid values are: seq (for sequential), rand (for random access). The default is seq.

-f *FileName*

Specifies the base file name used in the stress test. The default is stressFile. The file names for the tests are generated as: *FileName.Iteration.NodeWhereRunning*. For example: stressFile.1.gssio1, stressFile.2.gssio1,... for iterations 1 and 2 on node gssio1.

TargetPath

Specifies the GPFS file system mount point on which the stress test is performed.

Node1 [Node2...NodeN]

Is a space-separated list of nodes where the stress test is run. Do not use commas between the node names. For example:

```
gssio1 gssio2
```

The node must have file system mounted on it before performing any stress test.

-h | **--help**

Displays usage information about this command and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the **gssstress** command.

Example

This example shows how to run the stress test on GPFS file system `/gpfs/gpfs0`.

```
# mmmount gpfs0 -a  
# gssstress /gpfs/gpfs0 gssio1 gssio2
```

In this example, **gssstress** is invoked on the management server node. It is run on I/O server nodes `gssio1` and `gssio2` with `/gpfs/gpfs0` as the target path. By default, **gssstress** runs for 20 iterations, which can be adjusted using the `-i` option.

Location

`/opt/ibm/gss/tools/bin`

gssutils command

Installs and deploys ESS

Synopsis

```
gssutils {-N NODE-LIST | -G NODE-GROUP}  
[--prefix PREFIX] [--suffix SUFFIX] [-h | --help ]  
[--config CONFIG_FILE] [--customize]  
[--ems-node-nameEMS_NODE] [--io-node-one-nameIO_NODE1]  
[--io-node-two-nameIO_NODE2] [--prefix-namePREFIX_NAME] [--suffix-nameSUFFIX_NAME]
```

Availability

Available with the Elastic Storage Server.

Description

gssutils is the ESS installation and deployment toolkit. It is a collection of tools and utilities to facilitate SSR, installation and deployment, and upgrade tasks. **gssutils** can be invoked with or without any optional arguments.

It provides a set of task menus related to install and deploy activities. When a task is selected from the menu a command is issued to the system for that task. This toolkit requires a minimum of 80 x 24 character window to operate.

Restriction: Make sure that you use a console of adequate height and width to specify long options for the command during “C” button customization . An 80 x 24 console size might lead to an unexpected error, if the command option is very long.

At the bottom of the screen, the command for the task with the options is displayed. To change the options, you can press c. A dialog window opens to enter new or modified options. Select back or press Esc key to close the screen. You can select the corresponding menu item using 1 to 9 number keys. If you want to select a menu item mapped to a number that is greater than 9, you must scroll down to that menu item.

gssutils can be run on the ESS management server (EMS) node only. If no node name, node list, or group name are specified, `ems1` and `gss_ppc64` are considered as the default name for EMS and I/O server nodes respectively.

You can customize **gssutils** according to a specific environment such that parameters such as EMS and I/O server node names, and node name suffix and prefix are saved in the **gssutils** configuration file. As a result, you do not need to specify these parameters every time that you use **gssutils**. You can generate the **gssutils** configuration file specific to your environment using `--customize` option and that configuration file can be used for subsequent **gssutils** operations.

Parameters

-N *NODE-LIST*

Provides a list of nodes. If node list or group name is not provided, `-N localhost` is assumed.

-G *NODE-GROUP*

Provides the name of node group that the nodes are a member of. Nodes in the *NODE-LIST* are members of the *NODE-GROUP*.

--prefix *PREFIX*

Provides the host name prefix. Use = between `--prefix` and value if the value starts with -.

--suffix *SUFFIX*

Provides the host name suffix. Use = between --suffix and value if the value starts with -.

--config *CONFIG_FILE*

Provides the configuration file for **gssutils** for a specific environment.

--customize

Customizes the EMS host name and I/O server node names and generates the **gssutils** configuration file. This file can be used with --config to run **gssutils** specific to an environment.

--prefix *PREFIX_NAME*

Specifies the custom prefix name to populate the **gssutils** configuration file.

--suffix *SUFFIX_NAME*

Specifies the custom suffix name to populate the **gssutils** configuration file.

--ems-node-name *EMS_NODE*

Specifies the EMS host name to populate the **gssutils** configuration file.

--io-node-one-name *IO_NODE1*

Specifies the I/O node 1 host name to populate the **gssutils** configuration file.

--io-node-two-name *IO_NODE2*

Specifies the I/O node 2 host name to populate the **gssutils** configuration file.

-h | --help

Displays usage information about this script and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the **gssutlis** script.

Examples

- This example shows how to run **gssutils** with default values.

```
# gssutils
```

- This example shows how to run **gssutils** for **ems2** and **gss_ppc64** I/O node group, with the suffix **-10g**.

```
# gssutils -G ems2,gss_ppc64 --suffix=-10g
```

- This example shows how to run **gssutils** with the prefix **sk-** and with the suffix **-ib**.

```
# gssutils --prefix sk- --suffix=-ib
```

- This example shows how to customize **gssutils** for a specific environment. In this example environment **ems** node is **ems2** and I/O server node names are **io3** and **io4**. The suffix to be used is **-40g**.

```
# gssutils --customize --config /var/tmp/env2 --ems-node-name ems2 \  
--io-node-one-name io3 --io-node-two-name io4 --suffix=-40g
```

Successfully generated the customization configuration file.

In this example, the user has generated the configuration file specific to their environment. The --config option can be used to load **gssutils** with the values specific to the user environment.

```
# gssutils --config /var/tmp/env2
```

Location

```
/opt/ibm/gss/tools/bin
```

gssutils main menu

ESS Installation and Deployment Toolkit (gssutils) main menu

Synopsis

1. Help
2. SSR Tools >
3. Plug n Play and Fusion >
4. Install >
5. Upgrade >
6. Validation checks >
7. View/Collect service data (snaps) >
8. Exit

Availability

Available with the Elastic Storage Server.

Description

The `gssutils` main menu allows you to perform end-to-end operations for an ESS system.

Menu options

1. Help

Opens this manual page.

2. SSR Tools

Checks ESS for proper hardware installation, system configuration, and SAS connectivity. Using this menu item, users can perform the basic validation of the system once it arrives from manufacturing.

- a) Shows node details.
- b) Perform check of various install parameters.
- c) Perform a brief check of installed adapters and connected disks in the system.
- d) Perform storage cable connection and topology check. This test is meaningful when run on I/O server nodes.
- e) Perform I/O operations to the attached disks.
- f) Ping all nodes in the node list.
- g) Check ssh to all nodes in the node list.
- h) Run `lsscsi` from all nodes to show attached scsi devices.
- i) Show open serviceable events pending in the node.

3. Plug n Play and Fusion

Allows customers to build a cluster, file system and begin sampling the GUI as soon as possible.

Plug n Play

The goal of the Plug-N-Play mode is to allow customers to build a cluster, file system, and begin sampling the GUI as soon as possible. The stated goal is for this to be achieved in under an hour after Lab-Based Services (LBS) starts working on the system. Manufacturing now ships EMS with xCAT preconfigured with default settings.

Prerequisites:

- Unpacking and basic power connectivity are completed.
- FSP and xCAT networks are set up in documented ports and they are connected to proper VLANs.
- SSRs have done validation checks using `gssutils` to ensure correct disk placement, cabling, networking, and server health.
- Access to EMS is available over SSH for LBS.

Fusion Mode

The goal of the Fusion mode is to no longer require that Elastic Storage Server (ESS) systems be rediscovered or re deployed at a customer site. The end goal of this mode is to greatly reduce the time and the complexity in bringing up an ESS system. There are several tasks that you no longer have to perform:

- No need for **gssdeploy -x**: No need to install and rediscover the nodes through xCAT
- No need for **gssdeploy -d**: No need to reinstall the I/O nodes with Red Hat Enterprise Linux

Everything is treated as an upgrade and the amount of time saved significantly goes up if the system was shipped with the latest levels. This is achieved by shipping xCAT pre-configured out of manufacturing and providing a new tool (**gsschenv**) which automatically changes your IPs, host names, domain, and netmask.

This mode is called Fusion because it mixes parts of the upgrade and installation flows. The flow is all upgrade until the cluster creation. After cluster creation, it turns into installation because the cluster, file system, and GUI, etc. need to be set up.

4. Install

Allows customer to perform installation and deployment tasks of ESS system. It can be used for deploying a new system including EMS and I/O server nodes and creating GPFS file system after deployment.

5. Upgrade

Allows customer to perform installation and deployment tasks of ESS system. It can be used to upgrade the existing ESS deployment to latest version including upgrade of ESS and I/O server nodes along with GPFS cluster version upgrade.

6. Validation check

Performs stress test of the ESS hardware after installation or upgrade. Users can perform operations such as:

- See detailed properties of the attached storage.
- Perform network stress test.
- Run file system stress test.

7. View/Collect service data (snaps)

Collects various snaps of ESS system and GPFS such as:

- a) Run SAS adapter utility tool for miscellaneous service tasks of IP RAID adapter.
- b) Run IBM Power RAID adapter report generator for support.
- c) Collect gsssnap from the ESS Management Server (EMS) node.
- d) Collect gpfs.snap for service.
- e) Collect SOS report for service.
- f) Generate call home xml config file for call home setup.

8. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Install/Deploy menu

ESS Installation and Deployment Toolkit (gssutils) Install/Deploy menu

Synopsis

1. Help
2. Install and prepare software on ESS management server >
3. Deploy ESS software on IO server nodes >
4. Create GPFS cluster >
5. Back

Availability

Available with the Elastic Storage Server.

Description

The Install/Deploy menu allows customers to perform installation and deployment tasks of ESS system. It can be used for deploying a new system including EMS and I/O server nodes, and creating GPFS file system after deployment.

Menu options

1. Help

Opens this manual page.

2. Install and prepare software on ESS management server

Deploys ESS software on EMS server node. This is the first step towards the deployment of the ESS system. EMS node must be deployed first before deploying any other node in a new ESS cluster. Make sure you know IP address, host name and other network parameters of the ESS infrastructure for successfully deployment of the EMS node.

Here, users need to change the `gssdeploy.cfg` according to their environment. For more information, see the following sample content of `gssdeploy.cfg`.

Users also need to make sure that the `/etc/hosts` file is updated with the correct host name and IP address. ESS deployment is highly dependent on the `/etc/hosts` file. Make sure that this file is in the Linux format.

For PPC64LE deployments, you must know the server serial number for deployment. Users can run the `gssdeploy -f FSP_NETWORK` command to scan the list of systems in the FSP network. During scan, the server serial number is shown which can be used for the PPC64LE deployment.

Once all correct parameter have been set into `gssdeploy.cfg`, users can run the `gssdeploy` command along with `-x` or `-b` and `-x` combination. For more information, see *gssdeploy command*.

After successful EMS node deployment, users must back up the xCAT database. The xCAT database contains vital information which must be preserved and in case of an EMS node corruption, this xCAT database backup can be used to restore the EMS xCAT.

```
-----  
gssdeploy.cfg file  
-----  
#[RHEL]  
# Set to Y if RHEL DVD is used otherwise iso is assumed.  
RHEL_USE_DVD="N"  
  
#[RHEL]  
# Device location of RHEL DVD used instead of iso  
RHEL_DVD="/dev/cdrom"  
  
# Mount point to use for RHEL media.  
RHEL_MNT="/opt/ibm/gss/mnt"
```

```

# Directory containing ISO.
RHEL_ISODIR="/opt/ibm/gss/iso"

#[EMS]
# Hostname of EMS (ems1 is a default hostname from manufacturing)
EMS_HOSTNAME="ems1"

# Network interface for xCAT management network
EMS_MGTNETINTERFACE="enP3p9s0f0"

#[IOSERVERS]
# Default userid of IO Server.
SERVERS_UID="root"

# Default password of IO Server.
#
# To generate an encrypted password at the highest level supported
# by the crypt module use:
#
# python -c 'import crypt,getpass; print crypt.crypt(getpass.getpass())'
#
# Note: Password must be contained in single quotes.
SERVERS_PASSWD='cluster'

# Array of IO servers to provision and deploy.
# You can get serial numbers of the nodes using
# -f option of the gssdeploy. You can also use -i to
# identify the server once you know the IP address
# of FPSs using -f option.
#-----
# Applicable for LE systems only
#-----
SERVERS_SERIAL=(2159F3A 2159F4A)

# Array of IO servers to provision and deploy.
SERVERS_NODES=(gssio1 gssio2)

#[DEPLOY]
# OSIMAGE stanza to deploy to IO servers.
# NOTE: Architecture (e.g. ppc64le or pcc64be) stated in the
# DEPLOY_OSIMAGE is used to determine
# target architecture for deployment.
# Name ISO file
RHEL_ISO="RHEL-7.4-Server-ppc64**<LE>or<BE>***-dvd1.iso"
DEPLOY_OSIMAGE="rhels7.4-ppc64**<LE>or<BE>***-install-gss"

#-----
# Applicable for LE systems only
#-----
#[FSP or BMC]
FSP_MGTNETINTERFACE="enP3p9s0f3"
FSP_PASSWD="PASSWORD"
FSP_USERID=""

#-----
# Applicable for BE systems only
#-----
#[HMC]
HMC_HOSTNAME="hmc1"
HMC_ROOTUID="hscroot"
HMC_PASSWD="abc123"

# End of customization

```

3. Deploy ESS software on IO server nodes

Deploys ESS software on I/O server nodes. This task must be done after the deployment of the EMS management server is successfully completed.

Before deploying the I/O server node, users must run the necessary precheck. Failing to do so might result in failed deployment of I/O server nodes.

Nodes are deployed using **gssdeploy -d**. For more information, see *gssdeploy script*.

Deployment of I/O server node is a time taking process. Even after **gssdeploy -d** ends, the I/O server node deployment continues in the background with activities such as OS installation, post script execution, NIC initialization, etc. Users must check for the status of I/O server nodes before proceeding further.

```
$ nodestat gss_ppc64
gssiol: sshd
gssio2: sshd
```

Users must look into `/var/log/xcat/xcat.log` on all I/O server nodes to make sure all post scripts exited with the return status 0.

Once deployment of the I/O server node is complete, `/etc/hosts` from the I/O server node must be copied to all I/O server nodes.

```
$ xdcp gss_ppc64 /etc/hosts /etc/hosts
```

After copying the `/etc/hosts` file, I/O server nodes must be rebooted. Once reboot is complete, run post check to make sure I/O server nodes deployment completed correctly.

4. Create GPFS cluster

Creation of the GPFS cluster must be attempted only if the node deployment of EMS and I/O server nodes completed successfully. You must run `installcheck` and `prechecks` before proceeding with the GPFS cluster creation.

One more consideration users must take into account is of the high speed network deployment for the GPFS cluster. Consult your system admin to ensure that a high speed network (like InfiniBand) network is available for the GPFS cluster use.

Set up the high-speed network that is used for the cluster data communication. With the Ethernet high-speed network, you can use the **gssgennetworks** script to create a bonded Ethernet or Infiniband interface over active (up) high-speed network interfaces.

Ensure that either the `CONNECTED_MODE=no` statement exists or this line is commented out in the corresponding slave-bond interface scripts located in `/etc/sysconfig/network-scripts` directory of the management server and I/O server nodes. These scripts are created as part of the IP over IB bond creation. An example of the slave-bond interface with the modification is as follows.

```
TYPE=Infiniband <= change from Ethernet to Infiniband
NAME=bond-slave-ib0 <= bond-slave-ib0 is the slave connection
UUID=86c0af63-4b6c-475c-a724-0fb074dc9092
DEVICE=ib0 <= slave interfaceONBOOT=yes
MASTER=bond0 <= master bond interface
SLAVE=yes
CONNECTED_MODE=no <= This must be either set to no or commented out for ESS 5.2.0 onwards
NM_CONTROLLED=yes <= add this line
```

Note: The MTU must also be modified for the IB bond links to become active on reboot. This is hardware specific but 2044 or 4092 MTU are standard options.

Once the high-speed network is set up, stress test the high-speed network as follows:

```
$ GSSENV=TEST gssnettest -N ems1,gss_ppc64 --suffix=-hs
```

Once network configuration is completed, users can use the **gsscluster** command to create a new cluster and later on they can use **gssgenclusterrgs** to create recovery groups and **gssgenvdisks** to create vdisk, NSD, and file systems. For more information, see *gssgencluster*, *gssgenclusterrgs*, and *gssgenvdisks* manual pages.

After the creation of the GPFS cluster, users can consider adding the management server node as a client node to the GPFS cluster with the **gssaddnode** command. For more information, see *gssaddnode command*.

The GPFS GUI is an important piece of software that must be installed on the EMS node.

5. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Create GPFS cluster menu

ESS Installation and Deployment Toolkit (gssutils) Create GPFS cluster menu

Synopsis

1. Help
2. Manage high-speed network >
3. Create and manage the GPFS cluster >
4. Install the ESS GUI >
5. Back

Availability

Available with the Elastic Storage Server.

Description

This menu can be used for the creation of the GPFS cluster. This must be attempted only if the node deployment of EMS and I/O server nodes completed successfully. You must run and installcheck and pre-checks before proceeding with the GPFS cluster creation.

Menu options

1. Help

Opens this manual page.

2. Manage high-speed network

One important consideration users must take into account is of the high speed network deployment for the GPFS cluster. Consult your system admin to ensure that a high speed network (like InfiniBand) network is available for the GPFS cluster use.

Set up the high-speed network that is used for the cluster data communication. With the Ethernet high-speed network, you can use the **gssgennetworks** script to create a bonded Ethernet or Infiniband interface over active (up) high-speed network interfaces.

Ensure that either the `CONNECTED_MODE=no` statement exists or this line is commented out in the corresponding slave-bond interface scripts located in `/etc/sysconfig/network-scripts` directory of the management server and I/O server nodes. These scripts are created as part of the IP over IB bond creation. An example of the slave-bond interface with the modification is as follows.

```
TYPE=Infiniband <= change from Ethernet to Infiniband
NAME=bond-slave-ib0 <= bond-slave-ib0 is the slave connection
UUID=86c0af63-4b6c-475c-a724-0fb074dc9092
DEVICE=ib0 <= slave interfaceONBOOT=yes
MASTER=bond0 <= master bond interface
SLAVE=yes
CONNECTED_MODE=no <= This must be either set to no or commented out for ESS 5.2.0 onwards
NM_CONTROLLED=yes <= add this line
```

Note: The MTU must also be modified for the IB bond links to become active on reboot. This is hardware specific but 2044 or 4092 MTU are standard options.

Once the high-speed network is set up, stress test the high-speed network as follows:

```
$ GSSENV=TEST gssnettest -N ems1,gss_ppc64 --suffix=-hs
```

3. Create and manage the GPFS cluster

Once network configuration is completed, users can use the **gsscluster** command to create a new cluster and later on they can use **gssgenclusterrgs** to create recovery groups and **gssgenvdisks** to create vdisk, NSD, and file systems. For more information, see *gssgencluster*, *gssgenclusterrgs*, and *gssgenvdisks* manual pages.

After the creation of the GPFS cluster, users can consider adding the EMS node as a client node to the GPFS cluster with the **gssaddnode** command. For more information, see *gssaddnode command*.

4. Install the ESS GUI

The GPFS GUI is an important piece of software that must be installed on the EMS node.

5. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Create and manage GPFS cluster menu

ESS Installation and Deployment Toolkit (gssutils) Create and manage GPFS cluster menu

Synopsis

1. Help
2. Create the GPFS cluster
3. Create recovery group
4. Create Virtual Disks, NSDs & File System
5. Add EMS node into cluster
6. Cluster post creation task
7. Post validation of GPFS cluster
8. Back

Availability

Available with the Elastic Storage Server.

Description

This menu can be used to create the cluster, recovery groups, and file systems.

Menu options

1. Help

Opens this manual page.

2. Create the GPFS cluster

Create the GPFS cluster using I/O server nodes. For more information, see *gssgencluster command*.

Press *c* to change the EMS host name and I/O server nodes group name or I/O server node host name.

```
$ gssgencluster -G gss_ppc64 -C test01 --accept-license
```

3. Create recovery group

Creates the recovery group after creating the cluster. For more information, see *gssgenclusterrgs command*.

Press *c* to change the EMS host name and I/O server nodes group name or I/O server node host name.

```
$ gssgenclusterrgs -G gss_ppc64
```

4. Create Virtual Disks, NSDs & File System

Creates vdisk, NSD and file system after creating the recovery group. For more information, see *gssgenvdisks command*.

Press *c* to change the command.

```
$ gssgenvdisks --create-vdisk --create-nsds --create-filesystem --contact-node gssi01
```

5. Add EMS node into cluster

Adds the EMS node into the newly created cluster as a client node. For more information, see *gssaddnode command*.

Press *c* to change the command.

```
$ gssaddnode -N ems1 --cluster-node gssi01 --accept-license --no-fw-update
```

6. Cluster post creation task

After the cluster creation, there are some system restarts required. This menu can be used to perform those necessary restarts.

7. Post validation of GPFS cluster

Validates the cluster post successful creation.

8. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Install the ESS GUI menu

ESS Installation and Deployment Toolkit (gssutils) Install the ESS GUI menu

Synopsis

1. Help
2. Configure Zimon Collectors on on Mgmt Node
3. Setup Performance monitor on Mgmt and IO Nodes
4. Start Performance monitor sensor
5. Enable capacity and fileset quota monitoring
6. Start performance collector
7. Enable GPFS Mgmt GUI Service
8. Start GPFS Mgmt GUI
9. Back

Availability

Available with the Elastic Storage Server.

Description

This menu can be used to install the ESS GUI after after cluster creation.

Menu options

1. Help

Opens this manual page.

2. Configure Zimon Collectors on Mgmt Node

Generates the performance collector on the management server node by running the following command. The management server node must be part of the ESS cluster and the node name must be the node name used in the cluster (For example, ems1-hs).

```
$ mmperfmon config generate --collectors ems1-hs
```

Press c to change the EMS host name.

3. Setup Performance monitor on Mgmt and IO Nodes

Sets up the nodes in the ems node class and gss_ppc64 node class for performance monitoring by running the following command.

```
$ mmchnode --perfmon -N ems,gss_ppc64
```

4. Start Performance monitor sensor

Starts the performance monitoring sensors by running the following command

```
$ xdsh ems1,gss_ppc64 "systemctl start pmsensors"
```

5. Enable capacity and fileset quota monitoring

Capacity and fileset quota monitoring is not enabled in the GUI by default. You must correctly update the values and restrict collection to the management server node only.

1. To modify the GPFS Disk Capacity collection interval, run the following command:

```
$ mmperfmon config update GPFSDiskCap.restrict=EMSNodeName GPFSDiskCap.period=PeriodInSeconds
```

The recommended period is 86400 so that the collection is done once per day.

2. To restrict GPFS Fileset Quota to run on the management server node only, run the following command:

```
$ mmperfmon config update GPFSFilesetQuota.period=600 GPFSFilesetQuota.restrict=EMSNodeName
```

Note: To enable quota, the file system quota checking must be enabled. Refer **mmchfs -Q** and **mmcheckquota** commands in *IBM Spectrum Scale: Command and Programming Reference*.

Verify that the values are set correctly in the performance monitoring configuration by running the **mmperfmon config show** command on the management server node. Make sure that `GPFSDiskCap.period` is properly set, and `GPFSFilesetQuota` and `GPFSDiskCap` are both restricted to the management server node only.

Note: If you are moving from manual configuration to auto configuration then all sensors are set to default. Make the necessary changes using the **mmperfmon** command to customize your environment accordingly. For information on how to configure various sensors using **mmperfmon**, see the topic *Manually installing IBM Spectrum Scale GUI* in IBM Spectrum Scale documentation.

6. Start performance collector

Starts the performance collector on the management server node.

```
$ systemctl start pmcollector
```

7. Enable GPFS Mgmt GUI Service

Starts the management GUI service.

```
$ systemctl enable gpfsgui.service
```

8. Start GPFS Mgmt GUI

Starts the management GUI.

```
$ systemctl enable gpfsgui
```

9. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Manage high-speed network menu

ESS Installation and Deployment Toolkit (gssutils) Manage high-speed network menu

Synopsis

1. Help
2. Verify /etc/hosts is valid on all IO Nodes
3. View high-speed network setup
4. Setup high-speed network setup
5. ESS network stress test
6. Back

Availability

Available with the Elastic Storage Server.

Description

Users must take into account the high speed network deployment for the GPFS cluster. Consult your system admin to ensure that a high speed network (like InfiniBand) network is available for the GPFS cluster use.

Set up the high-speed network that is used for the cluster data communication. With the Ethernet high-speed network, you can use the **gssgennetworks** script to create a bonded Ethernet or Infiniband interface over active (up) high-speed network interfaces.

Menu options

1. Help

Opens this manual page.

2. Verify /etc/hosts is valid on all IO Nodes

Verifies that the /etc/hosts file is correct and valid. It must have correct host name and IP address mapping for EMS and I/O server nodes. It must also have host name and IP address for the high speed network.

3. View high-speed network setup

Verifies the high speed network already exists (or already deployed).

```
$ gssgennetworks -N ems1,gss_ppc64
```

In case the high speed network is already deployed, the command shows the configured IP address. A sample output follows.

```
# gssgennetworks -N ems1,gss_ppc64
2018-03-05T23:43:48.849836 Starting network generation
2018-03-05T23:43:49.492574 nodelist:   ems1 gssio1 gssio2
Interface(s) available on node ems1
Up interface:  enP1p9s0
Down interface: enP1p9s0d1
Up interface:  ib0
Down interface: ib1
Up interface(s) of node ems1 considered for bonding are ['enP1p9s0']
Interface(s) available on node gssio1
Up interface:  enP2p1s0
Up interface:  enP2p1s0d1
Up interface:  ib0
Down interface: ib1
Up interface(s) of node gssio1 considered for bonding are ['enP2p1s0', 'enP2p1s0d1']
Interface(s) available on node gssio2
Up interface:  enP2p1s0
Up interface:  enP2p1s0d1
Up interface:  ib0
```



```

Down interface: ib1
Up interface(s) of node gssio2 considered for bonding are ['enP2p1s0', 'enP2p1s0d1']
ems1: Current IP Address: 192.168.45.20
gssio1: Current IP Address: 192.168.45.21
gssio2: Current IP Address: 192.168.45.22

```

Here the high speed network is already deployed on the EMS and I/O server nodes. In this case, users can skip the high speed network creation.

Users must verify that the configured high speed network is the intended network that they want to configure. For more information, see **gssgennetwork command**.

Press c to change the EMS host name and I/O server nodes group name or I/O server node host name.

4. Setup high-speed network

High speed network can be setup using the **gssgennetwork** command. It is a command that can work over Ethernet or InfiniBand networks. Users must be ensure that the network interfaces on the EMS and I/O server nodes are connected to correct switch.

Note: Wrong switch connection and bond creation over a wrong connection might result in node connection loss.

```
$ gssgennetworks -N ems1,gss_ppc64 --suffix=-10g --create-bond
```

```

2018-02-28T01:41:11.413341 Starting network generation
2018-02-28T01:41:12.072727 nodelist: gssio1 gssio2
2018-02-28T01:41:12.072765 suffix used for network hostname: -10g
Interface(s) available on node gssio1-10g
Up interface: enP2p1s0
Up interface: enP2p1s0d1
Up interface: ib0
Down interface: ib1
Up interface(s) of node gssio1-10g considered for bonding are ['enP2p1s0', 'enP2p1s0d1']
Interface(s) available on node gssio2-10g
Up interface: enP2p1s0
Up interface: enP2p1s0d1
Up interface: ib0
Down interface: ib1
Up interface(s) of node gssio2-10g considered for bonding are ['enP2p1s0', 'enP2p1s0d1']
gssio1-10g: Current IP Address: IP not assigned
gssio2-10g: Current IP Address: IP not assigned
gssio1-10g: IP Address assigned to bond: 172.31.250.17
gssio2-10g: IP Address assigned to bond: 172.31.250.18
Bond creation complete

```

Here a new bond0 has been created and valid IP address has been assigned to newly created bond interfaces. For more information, see **gssgennetwork command**.

Press c to change the EMS host name and I/O server nodes group name or I/O server node host name.

5. ESS network stress test

Performs network stress test. This task must only be run in a test environment as it can generate high load on the network.

```
$ GSSENV=TEST gssnettest -N ems1,gss_ppc64
```

For more information, see **gssnettest command**.

Press c to change the EMS host name and I/O server nodes group name or I/O server node host name.

6. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - EMS node deployment menu

ESS Installation and Deployment Toolkit (gssutils) EMS node deployment menu

Synopsis

1. Help
2. Customize gssdeploy script configuration file
3. View and edit /etc/hosts file
4. FSP Subnet Range Scan (For LE Only)
5. Physically identify the nodes (For LE Only)
6. Check for EMS node before Install
7. Install ESS software on management server
8. Discover server nodes and create xCAT Object
9. Back up the xCAT database
10. Update EMS node >
11. Back

Availability

Available with the Elastic Storage Server.

Description

The menu can be used for ESS management server node (EMS) deployment.

Menu options

1. Help

Opens this manual page.

2. Customize gssdeploy script configuration file

This is the first step towards upgrade of the ESS system. Make sure you know IP address, host name and other network parameters of the ESS infrastructure to successfully upgrade the EMS node.

Here users need to change the gssdeploy.cfg file according to their environment.

3. View and edit /etc/hosts file

Users need to make sure that the /etc/hosts file is updated with the correct host name and IP address. ESS deployment is highly dependent on the /etc/hosts file. Make sure that this file is in the Linux format.

4. FSP Subnet Range Scan (For LE Only)

If deploying on the PPC64LE platform, gather information for the gssdeploy.cfg configuration file using the following commands when you are in close proximity with the rack containing the nodes.

Scan the nodes in the FSP subnet range:

```
$ /var/tmp/gssdeploy -f FSP_Subnet_Range
```

FSP_Subnet_Range is the FSP management node interface subnet range. For example, 10.0.0.0/24.

Note:

- It is recommended to use the IP address 10.0.0.1 for the management interface, if possible.
- It is highly recommended that you use the /24 netmask because scanning of the subnet takes a considerable duration of time if a wider network range is used.
- The gssdeploy -f command first determines if a DHCP server is running on the network. If the DHCP server is not running, it prompts you to start one so that the I/O server nodes can obtain addresses. Select Y to start the DHCP server when prompted.

- This command scans the specified subnet range to ensure that only the nodes on which you want to deploy are available. These include I/O server nodes and management server node (EMS). This command also returns the following:
 - Serial numbers and FSP numbers of the nodes in the building block
 - Serial numbers and IP addresses of I/O server nodes in the building block

Important: Do not proceed to the next step until FSP IP addresses and serial numbers of all known nodes are visible using the `gssdeploy -f` script.

5. Physically identify the nodes (For LE Only)

Physically identify the nodes in the rack.

```
$ /var/tmp/gssdeploy -i
```

With the `-i` option, `Node_IP`, `Default_Password`, and `Duration` need to be provided as input, where:

- `Node_IP` is the returned FSP IPMI IP address of the node obtained by using the `gssdeploy -f` command.
- `Default_Password` is the default password of the node, which is `PASSWORD`.
- `Duration` is the time duration in seconds for which the LED on the node should blink.

After you issue this command, the LED blinks on the specified node for the specified duration. You can identify the node in the rack using the blinking LED. Depending on the order of a node in the rack, its corresponding entry is made in the `gssdeploy.cfg` file. For example, for the bottommost node in the rack, its corresponding entry is put first in `gssdeploy.cfg`.

6. Check for EMS node before Install

Perform precheck to detect any errors and address them before proceeding further:

```
$ /opt/ibm/gss/tools/samples/gssprecheck -N ems1 --pre --install --file /var/tmp/gssdeploy.cfg
```

Note: `gssprecheck` gives hints on ways to fix any discovered issues. It is recommended to review each found issue carefully though resolution of all might not be mandatory.

7. Install ESS software on management server

Perform actual deployment of EMS node:

```
$ gssdeploy -b
```

The `-b` option installs xCAT software and its supporting software on the EMS node. In case of PPC64LE, if discovery fails then users can only run `gssdeploy -g`. For more information, see `gssdeploy` manual page.

Note: Make sure that you log off and log in back after this step before proceeding with any further steps.

8. Discover server nodes and create xCAT Object

Discover the nodes and create an entry in the xCAT database.

Note: Make sure that you log off and log in back to run the `gssdeploy -o` command to set the xCAT path.

9. Back up the xCAT database

Back up the xCAT database and save it to a location not on the management server node.

```
$ dumpxCATdb -p /var/tmp/db
$ tar -zcvf xCATDB-backup.tar.gz /var/tmp/db
```

In case of PPC64LE, if discovery fails then users can only run `gssdeploy -g`. For more information, see `gssdeploy` manual page.

10. Update EMS node

Perform update of the EMS node after deployment. This required to update the kernel, OS patches, OFED, IPRAID, etc. on the node.

11. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - EMS node update menu

ESS Installation and Deployment Toolkit (gssutils) EMS node update menu

Synopsis

1. Help
2. Update Kernel Errata Repo (Optional)
3. Update OS with Patch (Optional)
4. Update ESS management server node
5. Reboot EMS node
6. Update OFED on the management server node
7. Update IP RAID adapter firmware
8. Reboot EMS node again
9. Back

Availability

Available with the Elastic Storage Server.

Description

The menu can be used to update the EMS node to apply latest OS patch and update firmware.

Menu options

1. Help

Opens this manual page.

2. Update Kernel Errata Repo (Optional)

Set up the kernel and update system kernel repository. It's an optional step and required if new kernel needs to be installed on the EMS node.

```
$ /var/tmp/gssdeploy -k /var/tmp/RHSA-2017-2437-73-BE-KERNEL.tar.gz
```

Make sure you use the correct kernel tar ball. Also make sure the files inside the kernel tar ball must be complete in terms of dependency resolution. **gssdeploy** checks for the file permission inside kernel tar ball and if there are any wrong file permissions, **gssdeploy** reports it. Make sure the files inside tar ball at least have 444 file permission.

For example: Make sure that all RPMs in the `/install` directory including the extracted files in the kernel directory (`/install/gss/otherpkgs/rhels7/<arch>/kernel`), have the correct read permission for user, group, and others (`chmod 644` files).

```
/install/gss/otherpkgs/rhels7/<arch>/kernel  
-rw-r--r-- 1 nobody nobody 39315448 Nov 29 17:59 kernel-3.10.0-514.35.1.el7.ppc64.rpm
```

Also make sure that the file is placed inside one level of directory. Nesting files under more than one level might result in kernel repo creation failure. **gssdeploy** can strip more than one level of nested directory. For more information, see **gssdeploy** manual page.

Press `c` to change the tar ball name.

3. Update OS with Patch (Optional)

Set up the operating system patch yum repository. It's an optional step and required if new operating system patch needs to be installed on the EMS node.

```
$ /var/tmp/gssdeploy -p /var/tmp/systemd-RHBA-2017-1311-73-BE.tar.gz,/var/ \  
tmp/netmgr-RHBA-2017-1317-73-BE.tar.gz
```

gssdeploy -p creates a patch repository where users can give more than one patch tar ball in a comma-separated list.

Make sure you use the correct patch tar ball. Also make sure the files inside the patch tar ball must be complete in terms of dependency resolution. **gssdeploy** checks for the file permission

inside patch tar ball and if there are any wrong file permissions, **gssdeploy** reports it. Make sure the files inside tar ball at least have 444 file permission.

For example: Make sure that all RPMs in the /install directory including the extracted files in the patch directory (/install/gss/otherpkgs/rhels7/<arch>/patch), have the correct read permission for user, group, and others (chmod 644 files).

```
/install/gss/otherpkgs/rhels7/<arch>/patch
-rw-r--r-- 1 nobody nobody 5412240 Nov 29 12:02 systemd-219-42.e17_4.4.ppc64.rpm
-rw-r--r-- 1 nobody nobody 1785872 Nov 29 12:35 NetworkManager-1.8.0-11.e17_4.ppc64.rpm
```

Also make sure that the file is placed inside one level of directory. Nesting files under more than one level might result in patch repository creation failure. **gssdeploy** can strip more than one level of nested directory. For more information, see **gssdeploy** manual page.

Press c to change the EMS host name.

4. Update ESS management server node

Update the management server node. Here ems1 is the xCAT host name. This step installs the kernel, uninstalls OFED, installs IBM Spectrum Scale, and applies the IBM Spectrum Scale profile.

```
$ updatenode ems1 -P gss_updatenode
```

Make sure you reboot the node using menu item 5 and return on this menu to rerun the **updatenode** command again. The **updatenode** command must be run twice.

Press c to change the EMS host name.

5. Reboot EMS node

Use **systemctl** reboot to reboot the management server node and run menu item 4 again. This additional step rebuilds OFED for new kernel and builds GPFS™ portability layer (GPL) for IBM Spectrum Scale.

6. Update OFED on the management server node

Update OFED on the management server node:

```
$ updatenode ems1 -P gss_ofed
```

Press c to change the EMS host name.

7. Update IP RAID adapter firmware

Update the IP RAID Adapter firmware on the management server node:

```
$ updatenode ems1 -P gss_ipraid
```

Press c to change the EMS host name.

8. Reboot EMS node again

Final reboot of the EMS node after updating the binaries and firmware on the EMS node.

9. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Deploy I/O nodes menu

ESS Installation and Deployment Toolkit (gssutils) Deploy I/O nodes menu

Synopsis

1. Help
2. Run precheck before deploying IO node >
3. Deploy ESS software on IO Server nodes
4. Check deploy status
5. Check for Post script finish on IO Nodes before reboot
6. Copy /etc/hosts to all IO Nodes
7. Reboot IO nodes
8. Run post check >
9. Back

Availability

Available with the Elastic Storage Server.

Description

This menu can be used to deploy ESS software on I/O server nodes.

Menu options

1. Help

Opens this manual page.

2. Run precheck before deploying IO node

Run precheck to make sure I/O server nodes deployment will succeed. This task runs **gssinstallcheck** on the EMS node to make sure that the EMS server is functioning properly for successful the deployment of the I/O server nodes.

```
$ gssinstallcheck -N ems1
```

This script is used to verify IBM Spectrum Scale profile, OFED, and kernel, etc. Check for any error with the following:

- Installed packages
- Linux kernel release
- OFED level
- IPR SAS FW
- IPR SAS queue depth
- System firmware
- System profile setting
- Host adapter driver

Ignore other errors that might be flagged by the **gssinstallcheck** script. They go away after the remaining installation steps are completed. For more information, see *gssinstallcheck command*.

Run the **gssprecheck** script in full install mode and address any errors.

```
$ /opt/ibm/gss/tools/samples/gssprecheck -N ems1 --install --file /var/tmp/gssdeploy.cfg
```

Note: **gssprecheck** gives hints on ways to fix any discovered issues. It is recommended to review each found issue carefully though resolution of all might not be mandatory. For more information, see *gssprecheck command*.

3. Deploy ESS software on IO Server nodes

Deploy ESS software in the IO Server nodes from the ESS management node. Make sure that ESS management server deployed successfully before initiating this step.


```
$ /var/tmp/gssdeploy -d
```

For more information, see *gssdeploy script*.

4. Check deploy status

Check for deployment completion. Node status changes over time from `noping` to `sshd`. When **nodestate** displays `sshd` as the status for all I/O server nodes, this indicates that the installation is complete. Deployment may take up to 30 minutes to complete.

Run **nodestate** command again and again until system status becomes `sshd`.

```
$ nodestat gss_ppc64
```

After running the command, the output displays the OS image name or packages being installed. For example:

PPC64LE installations:

```
node: rhels7.4-ppc64le-install-gss
node: rhels7.4-ppc64le-install-gss
```

PPC64BE installations:

```
node: rhels7.4-ppc64-install-gss
node: rhels7.4-ppc64-install-gss
```

After about 30 minutes, the following output displays:

```
node: sshd
node: sshd
```

The installation is complete when **nodestat** displays `sshd` for all I/O server nodes. Here `gss_ppc64` is the xCAT node group containing I/O server nodes. To follow the progress of a node installation, you can tail the console log by using the following command:

```
$ tailf /var/log/ consoles/NodeName
```

where *NodeName* is the name of the node.

5. Check for Post script finish on IO Nodes before reboot

Once the **nodestat** command in step 5 shows `sshd`, it indicates that the deployment of the operating system is completed; however, xCAT post script installation starts. xCAT post scripts are very important for the successful configuration of the I/O server node after the operating system installation. Make sure this task does not display any active xCAT post script running.

```
$ xdsh gss_ppc64 "ps -eaf | grep -v grep | grep xcatpost"
```

If there are any processes still running, wait for them to complete.

6. Copy /etc/hosts to all IO Nodes

After the successful deployment and configuration of the I/O server node, you must copy the `/etc/hosts` to the I/O server nodes. It is very important that the I/O server nodes know about the host name and IP address for the upcoming GPFS cluster deployment.

```
$ xdcp gss_ppc64 /etc/hosts /etc/hosts
```

7. Reboot IO nodes

Reboot the I/O server nodes after successful deployment.

8. Run post check

Run various post checks on the deployed I/O server nodes to make sure that the I/O server node deployment completed successfully.

9. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - I/O nodes precheck menu

ESS Installation and Deployment Toolkit (gssutils) I/O nodes precheck menu

Synopsis

1. Help
2. Check and validate various install parameters for EMS
3. Check for EMS node before install
4. Back

Availability

Available with the Elastic Storage Server.

Description

This menu can be used to run precheck on EMS for I/O server nodes deployment.

Menu options

1. Help

Opens this manual page.

2. Check and validate various install parameters for EMS

Run precheck to make sure I/O server nodes deployment will succeed. This task runs **gssinstallcheck** on the EMS node to make sure that the EMS server is functioning properly for successful the deployment of the I/O server nodes.

```
$ gssinstallcheck -N ems1
```

This script is used to verify IBM Spectrum Scale profile, OFED, and kernel, etc. Check for any error with the following:

- Installed packages
- Linux kernel release
- OFED level
- IPR SAS FW
- IPR SAS queue depth
- System firmware
- System profile setting
- Host adapter driver

Ignore other errors that might be flagged by the **gssinstallcheck** script. They go away after the remaining installation steps are completed. For more information, see *gssinstallcheck command*.

3. Check for EMS node before install

Run the **gssprecheck** script in full install mode and address any errors.

```
$ /opt/ibm/gss/tools/samples/gssprecheck -N ems1 --install --file /var/tmp/gssdeploy.cfg
```

Note: **gssprecheck** gives hints on ways to fix any discovered issues. It is recommended to review each found issue carefully though resolution of all might not be mandatory. For more information, see *gssprecheck command*.

4. Back

Exit to the previous menu.

Exit status

- 0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - I/O nodes post check menu

ESS Installation and Deployment Toolkit (gssutils) I/O nodes post check menu

Synopsis

1. Help
2. Quick storage configuration check
3. Check enclosure cabling and paths to disks
4. Check disks for IO operations
5. Check for any serviceable event
6. Back

Availability

Available with the Elastic Storage Server.

Description

This menu can be used to run post check on I/O server nodes after successful deployment.

Menu options

1. Help

Opens this manual page.

2. Quick storage configuration check

Perform a brief check of installed adapters and connected disks in the system. For more information, see **gssstoragequickcheck command**.

```
/opt/ibm/gss/tools/bin/gssstoragequickcheck -G gss_ppc64
```

Press **c** to change the EMS host name and I/O server nodes group name or I/O server node host name.

3. Check enclosure cabling and paths to disks

Perform storage cable connection and topology check. This test is meaningful when run on I/O server nodes. For more information, see **gssfindmissingdisks command**.

```
/opt/ibm/gss/tools/bin/gssfindmissingdisks -G gss_ppc64
```

Press **c** to change the EMS host name and I/O server nodes group name or I/O server node host name.

4. Check disks for IO operations

Perform I/O operations to the attached disks. This test should only be run in a test environment. The node name must match the hostname of the node. For more information, see **gsscheckdisks command**.

```
GSSENV=INSTALL /opt/ibm/gss/tools/bin/gsscheckdisks -G gss_ppc64  
--enclosure-list all --iotest a --write-enable
```

Press **c** to change the EMS host name and I/O server nodes group name or I/O server node host name.

5. Check for any serviceable event

Show open serviceable events pending in the node. Serviceable events might be generated during deployment of nodes (see known issues). Address all open serviceable events. For more information, see **gssinstallcheck command**.

```
/opt/ibm/gss/tools/bin/gssinstallcheck -N ems1,gss_ppc64 --srv-event
```

Press **c** to change the EMS host name and I/O server nodes group name or I/O server node host name.

6. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Plug N Play and Fusion mode deployment menu

ESS Installation and Deployment Toolkit (gssutils) Plug N Play and Fusion mode deployment menu

Synopsis

1. Help
2. SSR Tools
3. Create GPFS cluster (Option 1)
4. Change IP, Hostname and Create GPFS cluster (Option 2)
5. Back

Availability

Available with the Elastic Storage Server.

Description

This menu can be used for Plug N Play and Fusion mode deployment of ESS system. It allows customers to build a cluster, file system and begin sampling the GUI as soon as possible.

Plug n Play

The goal of the Plug-N-Play mode is to allow customers to build a cluster, file system, and begin sampling the GUI as soon as possible. The stated goal is for this to be achieved in under an hour after Lab-Based Services (LBS) starts working on the system. Manufacturing now ships EMS with xCAT preconfigured with default settings.

Prerequisites:

- Unpacking and basic power connectivity are completed.
- FSP and xCAT networks are set up in documented ports and they are connected to proper VLANs.
- SSRs have done validation checks using **gssutils** to ensure correct disk placement, cabling, networking, and server health.
- Access to EMS is available over SSH for LBS.

Fusion Mode

The goal of the Fusion mode is to no longer require that Elastic Storage Server (ESS) systems be rediscovered or re deployed at a customer site. The end goal of this mode is to greatly reduce the time and the complexity in bringing up an ESS system. There are several tasks that you no longer have to perform:

- No need for **gssdeploy -x**: No need to install and rediscover the nodes through xCAT
- No need for **gssdeploy -d**: No need to reinstall the I/O nodes with Red Hat Enterprise Linux

Everything is treated as an upgrade and the amount of time saved significantly goes up if the system was shipped with the latest levels. This is achieved by shipping xCAT pre-configured out of manufacturing and providing a new tool (**gsschenv**) which automatically changes your IPs, host names, domain, and netmask.

This mode is called the Fusion mode because it mixes parts of the upgrade and installation flows. The flow is all upgrade until the cluster creation. After cluster creation, it turns into installation because the cluster, file system, and GUI, etc. need to be set up.

Menu options

1. Help

Opens this manual page.

2. SSR Tools

Checks ESS for proper hardware installation, system configuration, and SAS connectivity. Using this menu item, users can perform the basic validation of the system once it arrives from manufacturing.

3. Create GPFS cluster (Option 1)

The primary option is to build a very generic environment to allow the customer to preview their working Elastic Storage Server (ESS) system as fast as possible with the assumption that the final customizations are coming later. This gives the customers an opportunity to see their storage subsystem working right away. They start to get familiar with the installation process and the available file system space, start deciding on file system and block sizes, and become familiar with the GUI.

Create the GPFS cluster using default (1Gb network, default host name, default cluster/file system settings) to show the customer how fast we can bring the system and begin using it.

Requirement for option #1:

- Default manufacturing host name, IPs, user IDs, passwords
- Networking over the 1Gb (provisioning) only.

4. Change IP, Hostname and Create GPFS cluster (Option 2)

Change system IP, host name, etc. and create the GPFS cluster.

The secondary option is to start the process quickly to move the system into an actual installation state. There are several upfront items that need to be decided upon to choose this option. The result is a system that already has the actual host names, IPs, domain, netmasks, and potentially the high-speed connections. The disadvantage of going with option #2 is that you might not have all this information. Since the main goal of the Plug-N-Play mode is speed, the primary mode must be option #1 which allows the customer to start using ESS as fast as possible.

Requirement for option #2:

- All customer host name, IPs, netmasks, domain name must be known
- Optional: The high-speed network items must be known and connected properly to the switch. The switch must be configured correctly for bonding.

5. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Fusion mode deployment menu

ESS Installation and Deployment Toolkit (gssutils) Fusion mode deployment menu

Synopsis

1. Help
2. Destroy any previously deployed cluster >
3. Customize gsschenv.cfg with new node details
4. Apply changes using gsschenv.cfg file
5. Reboot EMS node
6. View and edit /etc/hosts file
7. Copy /etc/hosts to all IO Nodes
8. Compare the installed ESS version
9. Upgrade ESS Software on Management Node >
10. Upgrade the I/O server node (repeat for all nodes) >
11. Create GPFS cluster >
12. Back

Availability

Available with the Elastic Storage Server.

Description

This menu can be used for Fusion mode deployment of ESS system that includes changing the system networking, upgrading the binary (if required), and creating the GPFS cluster.

Menu options

1. Help

Opens this manual page.

2. Destroy any previously deployed cluster

Destroy any previously deployed cluster so that users can go ahead and change the system host name, IP address, etc. and recreate the GPFS cluster using the new host name and network.

3. Customize gsschenv.cfg with new node details

Customize your configuration file with the new and old information about your ESS system. For example:

```
# Modify the following
# HOSTNAME_ORIG = Original hostnames in your xCAT ESS environment
# IP_ADDR_ORIG = Original IPs in your xCAT ESS environment want (1 to 1 with HOSTNAME_ORIG)
# HOSTNAME_NEW = The new hostname (1 to 1 with the HOSTNAME_ORIG)
# IP_ADDR_NEW = The new IPs you want (1 to 1 with HOSTNAME_NEW/ORIG)
# NETMASK = The new netmask associated with the IPs
# DOMAIN = The new domain associated with the IPs

HOSTNAME_ORIG=(ems1 gssio1 gssio2)
IP_ADDR_ORIG=(192.168.202.20 192.168.202.21 192.168.202.22)
HOSTNAME_NEW=(modems1 modgssio1 modgssio2)
IP_ADDR_NEW=(192.168.202.20 192.168.202.21 192.168.202.22)
NETMASK="255.255.255.0"
DOMAIN="gpfs.net"
```

4. Apply changes using gsschenv.cfg file

Apply new changes to the system. The new changes to the system are applied depending on the configuration specified in the gsschenv.cfg file. I/O server nodes are rebooted automatically. However, the EMS node must be rebooted manually using menu item 5.

5. Reboot EMS node

Reboot the EMS node after applying the new host name, IP address, etc. Make sure you come to back to menu item 6 after the EMS node is rebooted.

6. View and edit /etc/hosts file

Users need to make sure that the /etc/hosts file is updated with the correct host name and IP address. ESS deployment is highly dependent on the /etc/hosts file. Make sure that the file is in the Linux format.

7. Copy /etc/hosts to all IO Nodes

After the successful deployment and configuration of I/O server nodes, you must copy the EMS /etc/hosts to I/O server nodes. This is very important because I/O server nodes must be aware of the host name and the IP address for the upcoming GPFS cluster deployment.

```
$ xdcp gss_ppc64 /etc/hosts /etc/hosts
```

8. Compare the installed ESS version

Compare the installed ESS version with the available new version of the ESS available on Fix Central. If any newer version is available at IBM FixCentral then make sure that you upgrade the ESS system using menu items 9 and 10.

9. Upgrade ESS Software on Management Node

Upgrade the ESS software on the EMS node.

10. Upgrade the I/O server node (repeat for all nodes)

Upgrade the ESS software on I/O server nodes. Make sure that you repeat all steps in this menu item for all nodes.

11. Create GPFS cluster

Create a new GPFS cluster.

12. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - SSR Tools menu

ESS Installation and Deployment Toolkit (gssutils) SSR Tools menu

Synopsis

1. Help
2. Show node details
3. Check and validate various install parameters
4. Quick storage configuration check
5. Check enclosure cabling and paths to disks
6. Check disks for IO operations
7. Ping all nodes
8. Check ssh to all nodes
9. Run lsscsi from all nodes
10. Check for open serviceable events
11. Back

Availability

Available with the Elastic Storage Server.

Description

The SSR Tools menu allows users to check ESS for proper hardware installation, system configuration, and SAS connectivity. Using this menu item, users can perform the basic validation of the system once it arrives from manufacturing.

Menu options

1. Help

Opens this manual page.

2. Show node details

Shows miscellaneous node information. For more information, see *gssnodedetails command*.

```
/opt/ibm/gss/tools/bin/gssnodedetails -N ems1,gss_ppc64
```

Press c to change the EMS host name and I/O server nodes group name or I/O server node host name.

3. Check and validate various install parameters

Performs check of various installation parameters. Address items identified as WARN or ERROR. Some parameters can not be checked until the system is fully deployed. For more information, see *gssinstallcheck command*.

```
/opt/ibm/gss/tools/bin/gssinstallcheck -N ems1,gss_ppc64
```

Press c to change the EMS host name and I/O server nodes group name or I/O server node host name.

4. Quick storage configuration check

Performs a brief check of installed adapters and connected disks in the system. For more information, see *gssstoragequickcheck command*.

```
/opt/ibm/gss/tools/bin/gssstoragequickcheck -N gss_ppc64
```

Press c to change the EMS host name and I/O server nodes group name or I/O server node host name.

5. Check enclosure cabling and paths to disks

Performs storage cable connection and topology check. This test is meaningful when run on I/O server nodes. For more information, see *gssfindmissingdisks command*.

```
/opt/ibm/gss/tools/bin/gssfindmissingdisks -G gss_ppc64
```

Press **c** to change the EMS host name and I/O server nodes group name or I/O server node host name.

6. Check disks for IO operations

Performs I/O operations to the attached disks. This test should only be run in a test environment. The node name must match the host name of the node. For more information, see *gsscheckdisks command*.

```
GSENV=INSTALL /opt/ibm/gss/tools/bin/gsscheckdisks -N ems1,gss_ppc64
--enclosure-list all --iotest a --write-enable --local
```

Press **c** to change the EMS host name and I/O server nodes group name or I/O server node host name.

7. Ping all nodes

Pings all nodes in the node list. For more information, see *gssruntask command*.

```
/opt/ibm/gss/tools/bin/gssruntask -N ems1,gss_ppc64 --task ping --mode ota
```

Press **c** to change the EMS host name and I/O server nodes group name or I/O server node host name.

8. Check ssh to all nodes

Checks SSH to all nodes in the node list. For more information, see *gssruntask command*.

```
/opt/ibm/gss/tools/bin/gssruntask -N ems1,gss_ppc64 --task ssh --mode ota
```

Press **c** to change the EMS host name and I/O server nodes group name or I/O server node host name.

9. Run lsscsi from all nodes

Runs **lsscsi** from all nodes to show attached scsi devices. For more information, see *gssruntask command*.

```
/opt/ibm/gss/tools/bin/gssruntask -N ems1,gss_ppc64 --task scsi
```

Press **c** to change the EMS host name and I/O server nodes group name or I/O server node host name.

10. Check for open serviceable events

Shows open serviceable events pending in the node. Serviceable events might be generated during deployment of nodes (see known issues). Address all open serviceable events.

```
/opt/ibm/gss/tools/bin/gssinstallcheck -N ems1,gss_ppc64 --srv-event
```

Press **c** to change the EMS host name and I/O server nodes group name or I/O server node host name.

11. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Upgrade menu

ESS Installation and Deployment Toolkit (gssutils) Upgrade menu

Synopsis

1. Help
2. Upgrade Pre-check for ESS System >
3. Upgrade ESS Software on Management Node >
4. Upgrade the I/O server node (repeat for all nodes) >
5. Update the enclosure firmware
6. Update the drive firmware
7. Upgrade the Management GUI on EMS node >
8. Back

Availability

Available with the Elastic Storage Server.

Description

The Install/Deploy menu allows customers to perform upgrade tasks of ESS system.

Prerequisites and supported upgrade paths

Before you begin the upgrade procedure, do the following:

- Ensure that the Red Hat Enterprise Linux (RHEL) 7.4 PPC64 or PPC64LE server ISO (for example, RHEL-7.4-20171119.0-Server-ppc64-dvd1.iso) is available in the /opt/ibm/gss/iso directory as specified in the gssdeploy.cfg configuration file.
- Disable the subscription manager and any external repositories by issuing the following commands on each node that you want to upgrade:

```
subscription-manager config --rhsm.manage_repos=0  
yum clean all
```
- Understand the implications of upgrading the release level to LATEST and upgrading the file system format version. After you complete the upgrade to the latest code level, you cannot revert to the previous code level. For more information, see [Completing the migration to a new level of IBM Spectrum Scale](#).

The following upgrade paths are supported:

- ESS version 5.2.x and 5.3.0 to version 5.3.1 on PPC64BE.
- ESS version 5.2.x and 5.3.0 to version 5.3.1 on PPC64LE.

Note: For upgrading to ESS 5.3.x from version 5.1.x or earlier (PPC64BE), you must contact IBM Support because direct upgrade to version 5.3.x from these versions is not supported. The available indirect upgrade paths are as follows.

- 3.5.5 (or earlier) > 4.5.2 > 5.1.x > 5.3.x
- 4.0.x > 5.0.x > 5.1.x (or 5.2.x) > 5.3.x
- 4.5.x (or 4.6.x) > 5.1.x > 5.3.x

Important: If you are not upgrading to ESS 5.3.x, it is recommended that you install ESS 5.2.2.1 to avoid system stability or functional issues.

Wait for any of these commands that are performing file system maintenance tasks to complete:

```
mmadddisk  
mmapplypolicy  
mmcheckquota
```

mmdeldisk
mmfsck
mmlssnapshot
mmrestorefs
mmrestripefile
mmrestripefs
mmrpldisk

Stop the creation and deletion of snapshots using **mmcrsnapshot** and **mmdelsnapshot** during the upgrade window.

Menu options

1. Help

Opens this manual page.

2. Upgrade Pre-check for ESS System

Run precheck for the entire ESS system to determine whether the system is ready for upgrade.

3. Upgrade ESS Software on Management Node

Upgrade is a two step process in which users must upgrade the EMS node first followed by I/O server nodes. Using this task user can perform an upgrade of the EMS node.

4. Upgrade the I/O server node (repeat for all nodes)

This is the second step of upgrade process in which after upgrading the EMS node users perform this task to upgrade the I/O server nodes. Make sure the steps in this subtask are repeated for all nodes one by one.

5. Update the enclosure firmware

Once the upgrade of the EMS and I/O server nodes is complete users must upgrade the enclosure firmware attached to I/O server nodes. To update the storage enclosure firmware, this task must be performed from one of the I/O server node of each building block.

```
$ xdsh gssi01 "mmchfirmware --type storage-enclosure"
```

Here the assumption is that for the first building block, the connected node is `gssi01`. Therefore, user must run **mmchfirmware --type storage-enclosure** from one of the I/O server nodes. In this case, it is being run only from the `gssi01` node.

Press `c` to change the I/O server node host name.

6. Update the drive firmware

Once the upgrade of EMS and I/O server nodes is complete, users must upgrade the drive firmware attached to I/O server nodes. Make sure that before upgrading drive firmware, the storage enclosure firmware is upgraded. To update the drive firmware, this task must be performed from one of the I/O server nodes of each building block.

```
$ xdsh gssi01 "mmchfirmware --type drive"
```

Here the assumption is that for the first building block, the connected node is `gssi01`. Therefore, user must run **mmchfirmware --type drive** from one of the I/O server nodes. In this case, it is being run only from the `gssi01` node.

Press `c` to change the I/O server node host name.

7. Upgrade the Management GUI on EMS node

Upgrade the management GUI running on EMS node.

8. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Upgrade precheck menu

ESS Installation and Deployment Toolkit (gssutils) EMS Upgrade precheck menu

Synopsis

11. Help
2. Perform GNR health check
3. Verify network connectivity
4. Verify GPFS system health
5. Check for any serviceable event
6. Check for deployment error
7. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to perform upgrade precheck tasks for the ESS system.

Menu options

1. Help

Opens this manual page.

2. Perform GNR health check

Perform GNR health check before going to upgrade. Only if GNR state is healthy, proceed with upgrade. All GNR problems and errors must be addressed before performing upgrade.

```
$ gnrhealthcheck
```

For more information, see *gnrhealthcheck command*.

3. Verify network connectivity

Verify network connectivity of the nodes connected in the existing deployed ESS system.

```
$ xdsh ems1,gss_ppc64 /usr/lpp/mmfs/bin/mmnetverify
```

Press *c* to change the EMS host name and I/O server nodes group name or I/O server node host name.

4. Verify GPFS system health

Verify GPFS system health of the nodes connected in the existing deployed ESS system.

```
$ /usr/lpp/mmfs/bin/mmhealth node show -N all
```

Press *c* to change the node name from "all" to a specific node.

5. Check for any serviceable event

Applicable for PPC64BE deployments. Any serviceable event on HMC must be addressed or acknowledged before going for upgrade. Serviceable events are generated by HMC in case of any hardware failure or if the system reaches the configured threshold.

```
$ gssinstallcheck -N ems1,gss_ppc64 --srv-event
```

Press *c* to change the EMS host name and I/O server nodes group name or I/O server node host name. For more information, see *gssinstallcheck command*.

6. Check for deployment error

Run precheck and check for any deployment errors. All precheck errors must be addressed before going for an upgrade.

```
$ gssinstallcheck -N ems1,gss_ppc64
```


Press `c` to change the EMS host name and I/O server nodes group name or I/O server node host name. For more information, see *gssinstallcheck* command.

7. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Upgrade EMS menu

ESS Installation and Deployment Toolkit (gssutils) Upgrade EMS menu

Synopsis

1. Help
2. Backup and Cleanup xCAT Database (optional)
3. Extract ESS Binaries >
4. Update ESS software repository
5. Customize gssdeploy script configuration file
6. Install ESS software on management server
7. Restore xCAT objects from backup (Optional)
8. EMS Pre upgrade tasks >
9. Perform EMS Upgrade >
10. EMS Post upgrade tasks >
11. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to upgrade the ESS system and install latest ESS software on EMS node. During the upgrade process if a step fails, it must be addressed before moving to the next step. Follow these steps to perform an upgrade of the ESS system.

Menu options

1. Help

Opens this manual page.

2. Backup and Cleanup xCAT Database (optional)

Perform cleanup and backup of xCAT DB. Only required if you are upgrading and xCAT or the Operating system changes. Perform the following steps if you are upgrading from ESS 5.2.x or 5.3.0.

If upgrading from ESS 5.2.x and 5.3.0, perform cleanup and save a backup copy of the xCAT database:

```
$ /opt/ibm/gss/install/rhel7/<arch>/samples/gssdeploy -c -r /var/tmp/xcatdb
```

For example:

On PPC64BE

```
$ /opt/ibm/gss/install/rhel7/ppc64/samples/gssdeploy -c -r /var/tmp/xcatdb
```

On PPC64LE

```
$ /opt/ibm/gss/install/rhel7/ppc64le/samples/gssdeploy -c -r /var/tmp/xcatdb
```

3. Extract ESS Binaries

Extract the latest ESS binaries to perform an upgrade.

4. Update ESS software repository

Update the ESS management server software repositories with new packages. Use the correct architecture of **gssinstall** (ppc64 or ppc64le). Make sure you regenerate the **gssutils** configuration file if using the customized environment file. Once complete, exit **gssutils** and re-run.

On PPC64BE

```
$ /usr/bin/python /var/tmp/gssinstall_ppc64 -u
```

On PPC64LE

```
$ /usr/bin/python /var/tmp/gssinstall_ppc64le -u
```

5. Customize gssdeploy script configuration file

This is the first step towards the upgrade of the ESS system. Make sure you know the IP address, host name, and other network parameters of the ESS infrastructure to successfully upgrade the EMS node.

Here users need to change the `gssdeploy.cfg` according to their environment. User also need to make sure that the `/etc/hosts` file is updated with the correct host name and IP address. ESS deployment is highly dependent on the `/etc/hosts` file. Make sure that the file is in the Linux format.

6. Install ESS software on management server

Perform actual deployment of the EMS node.

```
$ gssdeploy -b
```

The `-b` option installs xCAT software and its supporting software on the EMS node. In case if discovery fails on PPC64LE, you can only run **gssdeploy -g**.

Make sure that you log off and log in back after this step before proceeding with any further steps. For more information, see **gssdeploy command**.

7. Restore xCAT objects from backup (Optional)

Restore the xCAT object using the backed-up xCAT database. The backup was taken earlier for the older xCAT database. This step is required if either xCAT is of a newer version or the operating system changed.

Note: Make sure that you log off and log in back to run the **gssdeploy -o** command to set the xCAT path.

If upgrading from ESS 5.2.x or 5.3.0, install tools and xCAT and restore the xCAT database:

```
$ /var/tmp/gssdeploy -o -r /var/tmp/xcatdb
```

8. Pre upgrade tasks

Perform upgrade pre task to make sure the system is ready for upgrade. If upgrade pre task indicates any issue then it must be resolved before performing the upgrade.

Ignoring pre-check and forced upgrade might lead to failed deployment of ESS cluster nodes

9. Perform EMS Upgrade

Perform actual upgrade, if upgrade pre-check doesn't report any issues. Performing an upgrade shuts down the GPFS cluster on the EMS node until upgrade over.

10. EMS Post upgrade tasks

Perform post validation of the EMS node after successful ESS software upgrade.

11. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Upgrade EMS menu

ESS Installation and Deployment Toolkit (gssutils) Upgrade EMS menu

Synopsis

1. Help
2. Backup and Cleanup xCAT Database (optional)
3. Extract ESS Binaries >
4. Update ESS software repository
5. Customize gssdeploy script configuration file
6. Deploy new xCAT from backed-up xCAT DB (Optional)
7. Check for EMS node before upgrade
8. Perform EMS Upgrade >
9. Check and validate various install parameters
10. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to upgrade the ESS system and install latest ESS software on the EMS node. During the upgrade process if a step fails, it must be addressed before moving to the next step. Follow these steps to perform an upgrade of the ESS system.

Menu options

1. Help

Opens this manual page.

2. Backup and Cleanup xCAT Database (optional)

Perform cleanup and backup of xCAT DB. Only required if you are upgrading and xCAT or the Operatin system changes. Perform the following steps if you are upgrading from ESS 5.1.x or ESS 5.2.x.

If upgrading from ESS 5.1.x or ESS 5.2.x, perform cleanup and save a backup copy of the xCAT database:

```
$ /opt/ibm/gss/install/rhel7/<arch>/samples/gssdeploy -c -r /var/tmp/xcatdb
```

For example:

On PPC64BE

```
$ /opt/ibm/gss/install/rhel7/ppc64/samples/gssdeploy -c -r /var/tmp/xcatdb
```

On PPC64LE

```
$ /opt/ibm/gss/install/rhel7/ppc64le/samples/gssdeploy -c -r /var/tmp/xcatdb
```

3. Extract ESS Binaries

Extract the latest ESS binaries to perform an upgrade.

4. Update ESS software repository

Update the ESS management server software repositories with new packages. Use the correct architecture of **gssinstall** (ppc64 or ppc64le). Make sure you regenerate the **gssutils** configuration file if using the customized environment file. Once complete, exit **gssutils** and re-run.

On PPC64BE

```
$ /usr/bin/python /var/tmp/gssinstall_ppc64 -u
```

On PPC64LE

```
$ /usr/bin/python /var/tmp/gssinstall_ppc64le -u
```

5. Customize gssdeploy script configuration file

This is the first step towards the upgrade of the ESS system. Make sure you know the IP address, host name, and other network parameters of the ESS infrastructure to successfully upgrade the EMS node.

Here users need to change the `gssdeploy.cfg` according to their environment. User also need to make sure that the `/etc/hosts` file is updated with the correct host name and IP address. ESS deployment is highly dependent on the `/etc/hosts` file. Make sure that the file is in the Linux format.

6. Deploy new xCAT from backed-up xCAT DB (Optional)

Deploy new xCAT using the backed-up xCAT database. The backup was taken earlier for the older xCAT database. This step is required if either xCAT is of a newer version or the operating system changed.

If upgrading from ESS 5.1.x or ESS 5.2.x, install tools and xCAT and restore the xCAT database:

```
$ /var/tmp/gssdeploy -x -r /var/tmp/xcatdb
```

7. Check for EMS node before upgrade

Run pre-check on the EMS node to check for any error, before upgrade.

```
$ /opt/ibm/gss/tools/samples/gssprecheck -N ems1 --upgrade --file /var/tmp/gssdeploy.cfg
```

Note: `gssprecheck` gives hints on ways to fix any discovered issues. It is recommended to review each found issue carefully though resolution of all might not be mandatory.

8. Perform EMS Upgrade

Perform actual upgrade, if upgrade pre-check doesn't report any issues. Performing an upgrade shuts down the GPFS cluster on the EMS node until the upgrade is completed.

9. Check and validate various install parameters

Run install check and check for any deployment errors. All install check errors must be addressed before proceeding further.

```
$ gssinstallcheck -N ems1
```

For more information, see `gssinstallcheck` command.

Press `c` to change the EMS host name.

10. Back

Exit to the previous menu.

Exit status

`0` Successful completion.

`nonzero`

A failure has occurred.

Security

You must have root authority to run `gssutils`.

gssutils - Extract ESS binaries menu

ESS Installation and Deployment Toolkit (gssutils) Extract ESS binaries menu

Synopsis

1. Help
2. Extract ESS Installer Tar Ball
3. Check the MD5 checksum
4. Remove old ESS Installer
5. Extract ESS Installer
6. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to extract the latest ESS binaries for an upgrade. Users must use the ESS tar ball (.tgz) file to perform these tasks.

Restriction: Make sure that you use a console of adequate height and width to specify a large file name while extracting the tar ball. 80 x 24 console size might lead to an unexpected error. If you experience an error, these tasks can be performed from the shell prompt also.

Menu options

1. Help

Opens this manual page.

2. Extract ESS Installer Tar Ball

Unpack the ESS software archive (This is contained in ESS_STD_BASEIMAGE-5.x.x-ppc64-Linux.tgz).

```
$ tar -zxvf gss_install-5.x_ppc64_standard_20171215T203303Z.tgz
```

3. Check the MD5 checksum

Check the MD5 checksum for the extracted ESS installer.

```
$ md5sum -c gss_install-5.x_ppc64_standard_20171215T203303Z.md5
```

Make sure that you run this task from the /home/deploy directory. **gssutils** must be running from /home/deploy directory.

4. Remove old ESS Installer

Make sure the /opt/ibm/gss/install/rhel7/<ARCH> directory is clean:

```
$ /bin/sh gss_install-5.x_ppc64_standard_20171215T203303Z --remove
```

Note: If you are upgrading to 5.3.x from an earlier release, you might need to clean up the directory structure used in earlier releases. To do so, issue the following command:

```
/bin/sh gss_install-5.3.x_ppc64_standard_20171215T203303Z --remove --dir /opt/ibm/gss/install
```

5. Extract ESS Installer

Extract the ESS packages and accept the license as follows. By default, it is extracted to the /opt/ibm/gss/install directory:

```
$ /bin/sh gss_install-5.3.x_ppc64_standard_20171215T203303Z --text-only --silent
```

6. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Upgrade EMS detailed menu

ESS Installation and Deployment Toolkit (gssutils) Upgrade EMS detailed menu

Synopsis

1. Help
2. Update Kernel Errata Repo (Optional)
3. Update OS with Patch (Optional)
4. Update ESS management server node
5. Reboot EMS node
6. Update OFED on the management server node
7. Update IP RAID adapter firmware
8. Reboot EMS node again
9. Upgrade IBM Spectrum Scale RAID config
10. Upgrade IBM Spectrum Scale RAID config EMS
11. Re-start GPFS cluster
12. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to perform the actual upgrade of the EMS node and upgrade operating system, driver firmware, etc.

Menu options

1. Help

Opens this manual page.

2. Update Kernel Errata Repo (Optional)

Set up the kernel and update system kernel repository. It's an optional step and required if new kernel needs to be installed on the EMS node.

```
$ /var/tmp/gssdeploy -k /var/tmp/BE-KERNEL.tar.gz
```

Make sure you use the correct kernel tar ball. Also make sure the files inside the kernel tar ball must be complete in terms of dependency resolution. **gssdeploy** checks for the file permission inside kernel tar ball and if there are any wrong file permissions, **gssdeploy** reports it. Make sure the files inside tar ball at least have 444 file permission.

For example: Make sure that all RPMs in the /install directory including the extracted files in the kernel directory (/install/gss/otherpkgs/rhels7/<arch>/kernel), have the correct read permission for user, group, and others (chmod 644 files).

```
/install/gss/otherpkgs/rhels7/<arch>/kernel  
-rw-r--r-- 1 nobody nobody 39315448 Nov 29 17:59 kernel-3.10.0-xxx.xx.1.e17.ppc64.rpm
```

Also make sure that the file is placed inside one level of directory. Nesting files under more than one level might result in kernel repo creation failure. **gssdeploy** can strip more than one level of nested directory. For more information, see **gssdeploy** manual page.

Press c to change the tar ball name.

3. Update OS with Patch (Optional)

Set up the operating system patch yum repository. It's an optional step and required if new operating system patch needs to be installed on the EMS node.

```
$ /var/tmp/gssdeploy -p /var/tmp/systemd-BE.tar.gz,/var/ \  
tmp/netmgr-RHBA-BE.tar.gz
```


gssdeploy -p creates a patch repository where users can give more than one patch tar ball in a comma-separated list.

Make sure you use the correct patch tar ball. Also make sure the files inside the patch tar ball must be complete in terms of dependency resolution. **gssdeploy** checks for the file permission inside patch tar ball and if there are any wrong file permissions, **gssdeploy** reports it. Make sure the files inside tar ball at least have 444 file permission.

For example: Make sure that all RPMs in the /install directory including the extracted files in the patch directory (/install/gss/otherpkgs/rhels7/<arch>/patch), have the correct read permission for user, group, and others (chmod 644 files).

```
/install/gss/otherpkgs/rhels7/<arch>/patch
-rw-r--r-- 1 nobody nobody 5412240 Nov 29 12:02 systemd-xxx-xx.e17_4.4.ppc64.rpm
-rw-r--r-- 1 nobody nobody 1785872 Nov 29 12:35 NetworkManager-x.x.x-xx.e17_4.ppc64.rpm
```

Also make sure that the file is placed inside one level of directory. Nesting files under more than one level might result in patch repository creation failure. **gssdeploy** can strip more than one level of nested directory. For more information, see **gssdeploy** manual page.

Press c to change the EMS host name.

4. Update ESS management server node

Update the management server node. Here ems1 is the xCAT host name. This step installs the kernel, uninstalls OFED, installs IBM Spectrum Scale, and applies the IBM Spectrum Scale profile.

```
$ updatenode ems1 -P gss_updatenode
```

Make sure you reboot the node using menu item 5 and return on this menu to rerun the **updatenode** command again. The **updatenode** command must be run twice.

Press c to change the EMS host name.

5. Reboot EMS node

Use **systemctl** reboot to reboot the management server node and run menu item 4 again. This additional step rebuilds OFED for new kernel and builds GPFS™ portability layer (GPL) for IBM Spectrum Scale.

6. Update OFED on the management server node

Update OFED on the management server node:

```
$ updatenode ems1 -P gss_ofed
```

Press c to change the EMS host name.

7. Update IP RAID adapter firmware

Update the IP RAID Adapter firmware on the management server node:

```
$ updatenode ems1 -P gss_ipraid
```

Press c to change the EMS host name.

8. Reboot EMS node again

Final reboot of the EMS node after updating the binaries and firmware on the EMS node.

9. Upgrade IBM Spectrum Scale RAID config

Perform GNR callback settings.

```
$ /opt/ibm/gss/tools/samples/gssupg531.sh -b ems1,gss_ppc64
```

10. Upgrade IBM Spectrum Scale RAID config EMS

Set the EMS node config settings, optional node name or node class.

```
$ /opt/ibm/gss/tools/samples/gssupg531.sh -c
```

11. Re-start GPFS cluster

Start the GPFS cluster on all nodes.

12. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Actual EMS upgrade menu

ESS Installation and Deployment Toolkit (gssutils) Actual EMS upgrade menu

Synopsis

1. Help
2. Update Kernel Errata Repo (Optional)
3. Update OS with Patch (Optional)
4. Update ESS management server node
5. Reboot EMS node
6. Update OFED on the management server node
7. Update IP RAID adapter firmware
8. Reboot EMS node again
9. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to perform the actual upgrade of the EMS node and upgrade operating system, driver firmware, etc.

Menu options

1. Help

Opens this manual page.

2. Update Kernel Errata Repo (Optional)

Set up the kernel and update system kernel repository. It's an optional step and required if new kernel needs to be installed on the EMS node.

```
$ /var/tmp/gssdeploy -k /var/tmp/RHSA-2017-2437-73-BE-KERNEL.tar.gz
```

Make sure you use the correct kernel tar ball. Also make sure the files inside the kernel tar ball must be complete in terms of dependency resolution. **gssdeploy** checks for the file permission inside kernel tar ball and if there are any wrong file permissions, **gssdeploy** reports it. Make sure the files inside tar ball at least have 444 file permission.

For example: Make sure that all RPMs in the /install directory including the extracted files in the kernel directory (/install/gss/otherpkgs/rhels7/<arch>/kernel), have the correct read permission for user, group, and others (chmod 644 files).

```
/install/gss/otherpkgs/rhels7/<arch>/kernel  
-rw-r--r-- 1 nobody nobody 39315448 Nov 29 17:59 kernel-3.10.0-514.35.1.el7.ppc64.rpm
```

Also make sure that the file is placed inside one level of directory. Nesting files under more than one level might result in kernel repo creation failure. **gssdeploy** can strip more than one level of nested directory. For more information, see the **gssdeploy** manual page.

Press c to change the tar ball name.

3. Update OS with Patch (Optional)

Set up the operating system patch yum repository. It's an optional step and required if new operating system patch needs to be installed on the EMS node.

```
$ /var/tmp/gssdeploy -p /var/tmp/systemd-RHBA-2017-1311-73-BE.tar.gz,/var/ \  
tmp/netmgr-RHBA-2017-1317-73-BE.tar.gz
```

gssdeploy -p creates a patch repository where users can give more than one patch tar ball in a comma-separated list.

Make sure you use the correct patch tar ball. Also make sure the files inside the patch tar ball must be complete in terms of dependency resolution. **gssdeploy** checks for the file permission inside patch tar ball and if there are any wrong file permissions, **gssdeploy** reports it. Make sure the files inside tar ball at least have 444 file permission.

For example: Make sure that all RPMs in the /install directory including the extracted files in the patch directory (/install/gss/otherpkgs/rhels7/<arch>/patch), have the correct read permission for user, group, and others (chmod 644 files).

```
/install/gss/otherpkgs/rhels7/<arch>/patch
-rw-r--r-- 1 nobody nobody 5412240 Nov 29 12:02 systemd-219-42.e17_4.4.ppc64.rpm
-rw-r--r-- 1 nobody nobody 1785872 Nov 29 12:35 NetworkManager-1.8.0-11.e17_4.ppc64.rpm
```

Also make sure that the file is placed inside one level of directory. Nesting files under more than one level might result in patch repository creation failure. **gssdeploy** can strip more than one level of nested directory. For more information, see **gssdeploy** manual page.

Press c to change the EMS host name.

4. Update ESS management server node

Update the management server node. Here ems1 is the xCAT host name. This step installs the kernel, uninstalls OFED, installs IBM Spectrum Scale, and applies the IBM Spectrum Scale profile.

```
$ updatenode ems1 -P gss_updatenode
```

Make sure you reboot the node using menu item 5 and return on this menu to rerun the **updatenode** command again. The **updatenode** command must be run twice.

Press c to change the EMS host name.

5. Reboot EMS node

Use **systemctl** reboot to reboot the management server node and run menu item 4 again. This additional step rebuilds OFED for new kernel and builds GPFS™ portability layer (GPL) for IBM Spectrum Scale.

6. Update OFED on the management server node

Update OFED on the management server node:

```
$ updatenode ems1 -P gss_ofed
```

Press c to change the EMS host name.

7. Update IP RAID adapter firmware

Update the IP RAID Adapter firmware on the management server node:

```
$ updatenode ems1 -P gss_ipraid
```

Press c to change the EMS host name.

8. Reboot EMS node again

Final reboot of the EMS node after updating the binaries and firmware on the EMS node.

9. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - EMS pre-upgrade task menu

ESS Installation and Deployment Toolkit (gssutils) EMS pre-upgrade task menu

Synopsis

1. Help
2. Stop GPFS Management GUI
3. Backup PM Collector Configuration
4. Backup PM Sensor Configuration
5. Check for EMS node before upgrade
6. Stop GPFS Cluster
7. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to perform pre-upgrade tasks on the EMS node before upgrade.

Menu options

1. Help

Opens this manual page.

2. Stop GPFS Management GUI

Stop the management GUI.

```
$ systemctl stop gpfsogui.service
```

3. Backup PM Collector Configuration

On the management server node, stop GUI services, and save performance monitoring collector.

```
$ cp /opt/IBM/zimon/ZIMonCollector.cfg /var/tmp
```

4. Backup PM Sensor Configuration

On the management server node, stop GUI services, and save performance monitoring sensor configuration files.

```
$ cp /opt/IBM/zimon/ZIMonSensors.cfg /var/tmp
```

5. Check for EMS node before upgrade

Run pre-check on EMS node for any error, before upgrade.

```
$ /opt/ibm/gss/tools/samples/gssprecheck -N ems1 --upgrade --file /var/tmp/gssdeploy.cfg
```

Note: `gssprecheck` gives hints on ways to fix any discovered issues. It is recommended to review each found issue carefully though resolution of all might not be mandatory.

6. Stop GPFS Cluster

Stop GPFS cluster only on all nodes.

```
$ mmshutdown -a
```

7. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - EMS post-upgrade task menu

ESS Installation and Deployment Toolkit (gssutils) EMS post upgrade task menu

Synopsis

1. Help
2. Check and validate various install parameters
3. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to perform post upgrade task on the EMS node after upgrade.

Menu options

1. Help

Opens this manual page.

2. Check and validate various install parameters

Run precheck and check for any deployment error. All precheck errors must be addressed before going for an upgrade.

```
$ gssinstallcheck -N ems
```

For more information, see *gssinstallcheck command*.

Press c to change the EMS host name.

3. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Upgrade I/O server menu

ESS Installation and Deployment Toolkit (gssutils) Upgrade I/O server menu

Synopsis

1. Help
2. Prerequisite task before updating IO server node >
3. Update IO server node >
4. Post upgrade task after updating IO server node >
5. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to upgrade I/O server nodes one by one.

Menu options

- 1. Help**
Opens this manual page.
- 2. Prerequisite task before updating IO server node**
Prerequisite task that must be performed before updating I/O server nodes.
- 3. Update IO server node**
Perform actual upgrade of I/O server node binaries.
- 4. Post upgrade task after updating IO server node**
Post upgrade task that should perform after upgrading I/O server nodes.
- 5. Back**
Exit to the previous menu.

Exit status

- 0** Successful completion.
- nonzero** A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Upgrade I/O server menu

ESS Installation and Deployment Toolkit (gssutils) Upgrade I/O server menu

Synopsis

1. Help
2. Check for IO node before upgrade
3. Update IO server node >
4. Check for Physical Mapping (BE only)
5. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to upgrade I/O server nodes one by one.

Menu options

1. Help

Opens this manual page.

2. Check for IO node before upgrade

Run prechecks on I/O server node to make sure I/O server node is ready to upgrade.

```
$ /opt/ibm/gss/tools/samples/gssprecheck -N gssio1 --upgrade --file /var/tmp/gssdeploy.cfg
```

Note: `gssprecheck` gives hints on ways to fix any discovered issues. It is recommended to review each found issue carefully though resolution of all might not be mandatory.

For more information, see `gssprecheck` command. Press `c` to change the I/O server node host name.

3. Update IO server node

Perform actual upgrade of I/O server node binaries.

4. Check for Physical Mapping (BE only)

Check for physical mapping. Only applicable for PPC64BE deployments.

5. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run `gssutils`.

gssutils - Upgrade I/O server node detailed menu

ESS Installation and Deployment Toolkit (gssutils) Upgrade I/O server node detailed menu

Synopsis

1. Help
2. Update IO server node
3. Reboot IO node
4. Update OFED on the IO server node
5. Update IP RAID adapter firmware
6. Reboot IO node again
7. Update the SAS host adapter firmware
8. Update the node configuration
9. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to perform the actual upgrade of binaries, operating system, driver firmware, etc. on I/O server nodes.

Menu options

1. Help

Opens this manual page.

2. Update IO server node

Update the IO server node. Here gssio1 is the xCAT host name on IO node. This step installs the kernel, uninstalls OFED, installs IBM Spectrum Scale, and applies the IBM Spectrum Scale profile.

```
$ updatenode gssio1 -P gss_updatenode
```

Make sure you reboot the node using menu item 3 and return on this menu to rerun the **updatenode** command again. The **updatenode** command must be run twice.

Press c to change the I/O server node host name.

3. Reboot node

Use **systemctl** reboot to reboot the management server node and run menu item 2 again. This additional step rebuilds OFED for new kernel and builds GPFS™ portability layer (GPL) for IBM Spectrum Scale.

4. Update OFED on the IO server node

Update OFED on the I/O server node:

```
$ updatenode gssio1 -P gss_ofed
```

Press c to change the I/O server node host name.

5. Update IP RAID adapter firmware

Update the IP RAID Adapter firmware on the I/O server node :

```
$ updatenode gssio1 -P gss_ipraid
```

Press c to change the I/O server node host name.

6. Reboot IO node again

Final reboot of the I/O server node after updating the binaries and firmware.

7. Update the SAS host adapter firmware

After final reboot, update the SAS host adapter firmware for the upgraded I/O server node. Make sure you log in to I/O server node then upgrade the SAS host adapter firmware:

```
$ xdsh gssiol "mmchfirmware --type host-adapter"
```

Press `c` to change the I/O server node host name.

8. Update the node configuration

Update the updated I/O server node configuration by running this command from the EMS node.

```
$ /opt/ibm/gss/tools/samples/gssupg531.sh -s gssiol
```

Press `c` to change the I/O server node host name.

9. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Actual I/O server node upgrade menu

ESS Installation and Deployment Toolkit (gssutils) Actual I/O server node upgrade menu

Synopsis

1. Help
2. Update IO server node
3. Reboot IO node
4. Update OFED on the IO server node
5. Update IP RAID adapter firmware
6. Reboot IO node again
7. Update the SAS host adapter firmware
8. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to perform the actual upgrade of binaries, operating system, driver firmware, etc. on I/O server nodes.

Menu options

1. Help

Opens this manual page.

2. Update IO server node

Update the IO server node. Here gssio1 is the xCAT host name on IO node. This step installs the kernel, uninstalls OFED, installs IBM Spectrum Scale, and applies the IBM Spectrum Scale profile.

```
$ updatenode gssio1 -P gss_updatenode
```

Make sure you reboot the node using menu item 3 and return on this menu to rerun the **updatenode** command again. The **updatenode** command must be run twice.

Press c to change the I/O server node host name.

3. Reboot node

Use **systemctl** reboot to reboot the management server node and run menu item 2 again. This additional step rebuilds OFED for new kernel and builds GPFS™ portability layer (GPL) for IBM Spectrum Scale.

4. Update OFED on the IO server node

Update OFED on the I/O server node:

```
$ updatenode gssio1 -P gss_ofed
```

Press c to change the I/O server node host name.

5. Update IP RAID adapter firmware

Update the IP RAID Adapter firmware on the I/O server node :

```
$ updatenode gssio1 -P gss_ipraid
```

Press c to change the I/O server node host name.

6. Reboot IO node again

Final reboot of the I/O server node after updating the binaries and firmware.

7. Update the SAS host adapter firmware

After final reboot, update the SAS host adapter firmware for the upgraded I/O server node. Make sure you log in to I/O server node then upgrade the SAS host adapter firmware:

```
$ xdsh gssio1 "mmchfirmware --type host-adapter"
```

This command logs on to the node that was upgraded (gssio1) and invokes the SAS host adapter firmware command.

Press `c` to change the I/O server node host name.

8. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Pre-upgrade I/O server menu

ESS Installation and Deployment Toolkit (gssutils) Pre-upgrade I/O server menu

Synopsis

1. Help
2. Check for IO node before upgrade
3. Find the cluster and file system managers
4. Change the file system manager
5. Change the cluster manager
6. List the recovery groups
7. Move the recovery group
8. Unmount GPFS File System on Node
9. Shutdown GPFS on IO Node
10. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to perform prerequisite tasks before upgrading the binary of the I/O server node.

Menu options

1. Help

Opens this manual page.

2. Check for IO node before upgrade

Run precheck on I/O server node to make sure it is ready for upgrade.

```
$ /opt/ibm/gss/tools/samples/gssprecheck -N gssio1 --upgrade --file /var/tmp/gssdeploy.cfg
```

Note: `gssprecheck` gives hints on ways to fix any discovered issues. It is recommended to review each found issue carefully though resolution of all might not be mandatory.

For more information, see *gssprecheck command*.

Press `c` to change the I/O server node host name.

3. Find the cluster and file system managers

Find the cluster and file system managers.

```
mm1smgr
```

4. Change the file system manager

Move the file system manager role to another node if the current node is a file system manager.

```
$ mmchmgr gpfs0 gssio2-hs
```

Here `gssio2-hs` is another active GPFS node to which the file system manager role is going to transfer.

5. Change the cluster manager

Move the cluster manager role to another node if the current node is a cluster manager.

```
$ mmchmgr -c gssio2-hs
```

Here `gssio2-hs` is another active GPFS node to which the cluster manager role is going to transfer.

6. List the recovery groups

View the list of recovery groups.

```
$ mmlsrecoverygroup
```

To list the active server, primary server, and secondary server, run:

```
$ mmlsrecoverygroup rg_gssio1-hs -L | grep active -A2
```

Here `rg_gssio1-hs` is a recovery group which is running on the `gssio1-hs` node which is going to upgrade. The upgrade candidate node must not be a primary server for any recovery group, if user is planning to upgrade that node.

7. Move the recovery group

User must move the recovery group of going to upgrade node to another active GPFS node.

To move the recovery group from the current active I/O server node (`rg_gssio1-hs`) to the peer I/O server node (`gssio2-hs`) in the same building block, run:

```
$ mmchrecoverygroup rg_gssio1-hs --servers gssio2-hs,gssio1-hs
```

Here the `rg_gssio1-hs` recovery group is being moved to another I/O server node `gssio2-hs`. Now `gssio2-hs` is the primary server for `rg_gssio1-hs` recovery group.

8. Unmount GPFS File System on Node

Unmount the GPFS file system on the node which is going to be upgraded as the active recovery group have moved from current node to another node.

```
$ mmunmount all -N gssio1-hs
```

Here the file system is being unmounted from I/O server node `gssio1-hs`. As `gssio1-hs` is no longer the primary server of any of the recovery group the file system cannot be unmounted on this particular node.

9. Shutdown GPFS on IO Node

Once the file system has been unmounted on the node to be upgraded, shut down GPFS services on this node.

```
$ mmshutdown -N gssio1-hs
```

Here the GPFS service is being shut down on the `gssio1-hs` node as the file system is already unmounted.

10. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Post-upgrade I/O server menu

ESS Installation and Deployment Toolkit (gssutils) Post-upgrade I/O server menu

Synopsis

1. Help
2. Check for Physical Mapping (BE only)
3. Re-start GPFS cluster on IO Node
4. Back

Availability

Available with the Elastic Storage Server.

Description

Use this menu to perform post-upgrade tasks after updating the I/O server node.

Menu options

- 1. Help**
Opens this manual page.
- 2. Check for Physical Mapping (BE only)**
Check for physical mapping, only applicable for PPC64BE deployments.
- 3. Re-start GPFS cluster on IO Node**
Start the GPFS cluster on updated I/O server nodes.
- 4. Back**
Exit to the previous menu.

Exit status

- 0** Successful completion.
- nonzero** A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Validation and stress test menu

ESS Installation and Deployment Toolkit (gssutils) Validation and stress test menu

Synopsis

1. Help
2. Show storage details >
3. ESS network stress test
4. ESS filesystem stress test
5. Back

Availability

Available with the Elastic Storage Server.

Description

This menu can be used to perform stress test of the ESS hardware after installation or upgrade.

Menu options

1. Help

Opens this manual page.

2. Show storage details

View detailed properties of the attached storage.

- a) Show attached enclosures
- b) Show attached enclosures that are not-ok
- c) Show attached enclosure detail
- d) Show firmware details of SAS adapters
- e) Show firmware details of storage enclosures
- f) Show firmware details of installed drives
- g) Show pdisks that are not-ok

3. ESS network stress test

Perform network stress test. This task must only be run in a test environment as it can generate high load on the network.

```
$ GSSENV=TEST gssnettest -N ems1,gss_ppc64
```

For more information, see *gssnettest command*.

Press c to change the EMS host name and I/O server nodes group name or I/O server node host name.

4. ESS filesystem stress test

Run file system stress test. Inspect and address any errors logged in `/var/log/messages` on the EMS node and `/var/adm/ras/mmfs.log.latest` on I/O server nodes.

```
$ gssstress /gpfs/gpfs0 gssiol gssio2
```

For more information, see *gssstress command*.

Press c to change the file system name or the I/O server node host name.

5. Back

Exit to the previous menu.

Exit status

- 0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Storage subsystem details menu

ESS Installation and Deployment Toolkit (gssutils) Storage subsystem details menu

Synopsis

1. Help
2. Show attached enclosures
3. Show attached enclosures that are not-ok
4. Show attached enclosure detail
5. Show firmware details of SAS adapters
6. Show firmware details of storage enclosures
7. Show firmware details of installed drives
8. Show pdisks that are not-ok
9. Back

Availability

Available with the Elastic Storage Server.

Description

This menu can be used to view details of the storage subsystem including enclosure and disks.

Menu options

1. Help

Opens this manual page.

2. Show attached enclosures

View attached enclosures in the node list (**mm1senclosure all**). The node(s) must be member of an ESS cluster.

```
$ GSSSSH=ssh gssruntask -G gss_ppc64 --task mm1senclosure-all
```

For more information, see *gssruntask command*.

Press **c** to change the I/O server nodes group name or I/O server node host name.

3. Show attached enclosures that are not-ok

View attached enclosure in the node list needing service (**mm1senclosure all --not-ok**). The node(s) must be member of an ESS cluster.

```
$ GSSSSH=ssh gssruntask -G gss_ppc64 --task mm1senclosure-all-not-ok
```

For more information, see *gssruntask command*.

Press **c** to change the I/O server nodes group name or I/O server node host name.

4. Show attached enclosure detail

View attached enclosure details in the node list (**mm1senclosure all -L**). The node(s) must be member of an ESS cluster.

```
$ GSSSSH=ssh gssruntask -G gss_ppc64 --task mm1senclosure-all-L
```

For more information, see *gssruntask command*.

Press **c** to change the I/O server nodes group name or I/O server node host name.

5. Show firmware details of SAS adapters

View firmware of SAS adapters in the node list (**mm1sfirmware --type host-adapter**).

```
$ GSSSSH=ssh gssruntask -G gss_ppc64 --task mm1sfirmware-adapt
```

For more information, see *gssruntask command*.

Press **c** to change the I/O server nodes group name or I/O server node host name.

6. Show firmware details of storage enclosures

View firmware of attached enclosures in the node list (**mmlsfirmware --type storage-enclosure**).

```
$ GSSSSH=ssh gssruntask -G gss_ppc64 --task mmlsfirmware-enc1
```

For more information, see *gssruntask command*.

Press **c** to change the I/O server nodes group name or I/O server node host name.

7. Show firmware details of installed drives

View firmware details of installed disks in the node list (**mmlsfirmware --type drive**).

```
$ GSSSSH=ssh gssruntask -N gss_ppc64 --task mmlsfirmware-drv
```

For more information, see *gssruntask command*.

Press **c** to change the I/O server nodes group name or I/O server node host name.

8. Show pdisks that are not-ok

View attached pdisks that are not healthy in the node list (**mmlspdisk all --not-ok**).

```
$ GSSSSH=ssh gssruntask -N gss_ppc64 --task mmlspdisk-not-ok
```

For more information, see *gssruntask command*.

Press **c** to change the I/O server nodes group name or I/O server node host name.

9. Back

Exit to the previous menu.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

gssutils - Collect misc service data menu

ESS Installation and Deployment Toolkit (gssutils) Collect misc service data menu

Synopsis

1. Help
2. Run IBM Power RAID adapter tool (iprconfig)
3. Run IBM Power RAID adapter report generator (iprsos)
4. Collect ESS support data (gsssnap)
5. Collect gpfs.snap
6. Collect sos report
7. Create xml file for callhome
8. Back

Availability

Available with the Elastic Storage Server.

Description

This menu can be used to view and collect miscellaneous service data.

Menu options

1. Help

Opens this manual page.

2. Run IBM Power RAID adapter tool (iprconfig)

Run SAS adapter utility tool for miscellaneous service tasks of IP RAID adapter.

```
$ iprconfig
```

3. Run IBM Power RAID adapter report generator (iprsos)

Run IBM Power RAID adapter report generator for support.

```
$ iprsos
```

4. Collect ESS support data (gsssnap)

Collect **gsssnap** from the ESS Management Server (EMS) node. It must be run on the EMS node.

```
$ /opt/ibm/gss/xcat/bin/gsssnap -i
```

5. Collect gpfs.snap

Collect **gpfs.snap** for service. The snap is stored in `/tmp/mmfs`. The EMS node must be a member of an IBM Spectrum Scale cluster.

```
$ /usr/lpp/mmfs/bin/gpfs.snap
```

6. Collect sos report

Collect sos report for service.

```
$ sosreport
```

7. Create xml file for callhome

Generate callhome xml configuration file for the callhome setup. This command must be run on the EMS node.

```
$ gsscrchxml -N ems1,gss_ppc64 --onscreen
```

8. Back

Exit to the previous menu.

Exit status

- 0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run **gssutils**.

Chapter 2. ESS scripts

This section includes descriptions of the ESS scripts.

Descriptions of these ESS scripts follow:

- “gssdelvdisk script” on page 124
- “gssdeploy script” on page 125
- “gssinstall script” on page 130
- “gsssnap script” on page 49
- “gssupgrade.sh script” on page 134

ESS also includes the **mtuset** script in `/opt/ibm/gss/tools/samples` for changing the MTU.

For information about ESS commands, see Chapter 1, “ESS commands,” on page 1.

For information about IBM Spectrum Scale RAID commands and scripts, see *IBM Spectrum Scale RAID: Administration*.

For information about other IBM Spectrum Scale commands, see *IBM Spectrum Scale: Command and Programming Reference*.

gssdelvdisks script

Deletes the file system and the associated NSDs and vdisks.

Synopsis

```
gssdelvdisks {-c | -l | -s | -a | -h}
```

Availability

Available with the Elastic Storage Server.

Description

The **gssdelvdisks** script deletes the file system and the associated NSDs and vdisks. This script works only in case file system has been unmounted.

CAUTION:

Make sure that you understand the implications of running this command. It is a destructive command and you might lose the entire file system data. It is a non-recoverable process.

Parameters

- c** Specifies the contact node which is a part of the GPFS cluster.
- l** Specifies using a local system as the contact node.
- s**
Performs operation silently.
- a** Deletes the log vdisk also from recovery groups
- h | --help**
Displays usage information about this script and exits.

Exit status

- 0** Successful completion.
- nonzero**
A failure has occurred.

Security

You must have root authority to run the **gssdelvdisks** script.

Example

This example shows how to delete file system, NSD, vdisks including log vdisks.

```
$ gssdelvdisks -l -a
```

The **-a** option allows to delete the log vdisks also inside all recovery groups.

The **-l** option indicates using local node as the contact node.

Location

```
/opt/ibm/gss/install/samples
```

gssdeploy script

Primary tool for ESS deployment.

Synopsis

```
gssdeploy [config_file][ -b | --base ] [ -c | --clean ] [ -d | --deploy ]
[ -f | --find ] [ -g | --genesis ] [ -i | --identify ]
[ { -k | --kernel } TGZ-FILE { ppc64 | ppc64le } ] [ -l | --less ] [ -m | --getmacs ]
[ -o | --object ] [ { -p | --patch } TGZ-FILE1,TGZ-FILE2... [ { ppc64 | ppc64le } [PATCH-DIR]]
[ { -r | --restore } Directory ] [ -s | --silent ] [ -S | --Silent ]
[ -x | --xcat ] [ -V | --version ] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

gssdeploy is the primary tool for ESS system deployment. The **gssdeploy** deployment script is highly dependent on the parameter specified in the `gssdeploy.cfg` configuration file. Users must understand the entries in `gssdeploy.cfg` then proceed with the ESS deployment.

Update the `gssdeploy.cfg` file according to your requirements and the gathered information. Make sure you have edited the `gssdeploy.cfg` according to your environment before using `gssdeploy -b` or `-o`. The options that you can specify in the `gssdeploy.cfg` file include:

- Whether use DVD for installation: `RHEL_USE_DVD`
The default option is to use ISO.
- If DVD, then device location: `RHEL_DVD`
- Mount point to use for RHEL media: `RHEL_MNT`
- ISO location: `RHEL_ISODIR`
The default location is `/opt/ibm/gss/iso`.
- ISO file name: `RHEL_ISO`
- EMS host name: `EMS_HOSTNAME`
- Network interface for xCAT management network: `EMS_MGTNETINTERFACE`
- Network interface for FSP network: `FSP_MGTNETINTERFACE` [**Not applicable for PPC64BE**]
- FSP default IPMI password: `FSP_PASSWD` [**Not applicable for PPC64BE**]
- HMC host name: `HMC_HOSTNAME` [**Not applicable for PPC64LE**]
- HMC default user ID: `HMC_ROOTUID` [**Not applicable for PPC64LE**]
- HMC default password: `HMC_PASSWD` [**Not applicable for PPC64LE**]
- Type of deployment: `DEPLOYMENT_TYPE`
The default type of deployment is ESS. It can also be `ADD_BB`.
ESS: Deploys I/O server nodes.
ADD_BB: Adds new building block of I/O server nodes.
PPC64LE protocol nodes can be deployed using the CES or ADD_CES deployment types.
CES: Deploys protocol nodes.
ADD_CES: Adds new protocol nodes.
- I/O server user ID: `SERVERS_UID`
- I/O server default password: `SERVERS_PASSWD`
- I/O server serial numbers: `SERVERS_SERIAL` [**Not applicable for PPC64BE**]
- I/O server node names: `SERVERS_NODES`

For example, gssio1 gssio2

- Deployment OS image: DEPLOY_OSIMAGE

Note: For PPC64LE, there must be a one-to-one relationship between serial number and node in gssdeploy.cfg and for every node specified in gssdeploy.cfg, there must be a matching entry in /etc/hosts.

Parameters

config_file

Specifies the optional path for the customized gssdeploy.cfg file.

-b | --base

Performs base EMS xCAT and GSS package installation.

-c | --clean

Performs an interactive management server xCAT **dumpxCATdb** operation and cleanup of previous management server xCAT installation and ESS installed RPM packages.

-d | --deploy

Performs an interactive deployment of the configured I/O servers.

-f | --find

Finds node machine type and serial number attached on the FSP network.

-g | --genesis

Performs the node discovery (genesis).

-i | --identify

Identifies node with the FSP IP address.

{ -k | --kernel } TGZ-FILE [ppc64 | ppc64le]

Sets up the kernel repository. *TGZ-FILE* is the tgz file containing kernel update RPMs. Optionally target architecture can be specified. Default is the run time architecture of the EMS node. When more than one arguments are provided, they must be enclosed in quotes. For example, -k "kernel_update.tgz ppc64le".

Example: **gssdeploy -k RHSA-2017-2437-73-LE-KERNEL.tar.gz**

-l | --less

Shows brief output.

-m | --getmacs

Gets the MAC address of nodes.

-o | --object

Performs the required object creation and node discovery. This option requires base EMS xCAT and GSS packages to be already installed.

{ -p | --patch } TGZ-FILE1,TGZ-FILE2... [{ ppc64 | ppc64le } [PATCH-DIR]]

Sets up the EFIX/patch repository. *TGZ-FILEx* are the tgz files containing patch RPMs. Optionally target architecture can be specified. Default is the run time architecture of the EMS node. *PATCH-DIR* is the optional EFIX/patch directory. Default is patch. When more than one arguments are provided, they must be enclosed in quotes. For example, -p "patch_update.tgz ppc64le patch1".

Example: **gssdeploy -p systemd_LE.tar.gz,netmgr_LE.tar.gz**

{ -r | --restore } Directory

Specifies an xCAT database dump directory. When it is used in conjunction with the **--clean** (or **-c**) option, the xCAT database is saved to the specified directory. When it is used in conjunction with the **--xcat** (or **-x**) option, an xCAT database restore operation is performed using data from the specified directory.

- s | --silent**
Performs the selected operation non-interactively.
- S | --Silent**
Performs the selected operation non-interactively. Stops on error.
- x | --xcat**
Performs an interactive management server xCAT and ESS package installation. Creates the required objects and performs node discovery.
- V | --version**
Displays the program's version number and exits.
- h | --help**
Displays usage information about this script and exits.

Exit status

- 0** Successful completion.
- nonzero**
A failure has occurred.

Security

You must have root authority to run the **gssdeploy** script.

Examples

1. This example shows how to clean the current xCAT installation and associated configuration to remove any preexisting xCAT configuration, and then address any errors before proceeding:

```
$ /opt/ibm/gss/install/samples/gssdeploy -c
```

 This command cleans any previous ESS deployment file, repos, etc. from the system and makes the system ready for the new deployment.
2. This command scans the specified subnet range to ensure that only the nodes on which you want to deploy are available. These include I/O server nodes and management server node (EMS).

```
$ tmp/gssdeploy -f FSP_Subnet_Range
```

The **gssdeploy -f** command first determines if a DHCP server is running on the network. If the DHCP sever is not running, it prompts you to start one so that the I/O server nodes can obtain addresses. Select Y to start the DHCP server when prompted.

This command also returns the following:

- Serial numbers and FSP numbers of the nodes in the building block
- Serial numbers and IP addresses of I/O server nodes in the building block

Note: Do not proceed to the next step until FSP IP addresses and serial numbers of all known nodes are visible using the **gssdeploy -f** script.

3. This examples shows how to physically identify the nodes in the rack.

```
$ /var/tmp/gssdeploy -i
```

 With the **-i** option, *Node_IP*, *Default_Password*, and *Duration* need to be provided as input, where:
 - *Node_IP* is the returned FSP IPMI IP address of the node obtained by using the **gssdeploy -f** command.
 - *Default_Password* is the default password of the node, which is `PASSWORD`
 - *Duration* is the time duration in seconds for which the LED on the node should blink.
 After you issue this command, the LED blinks on the specified node for the specified duration. You can identify the node in the rack using the blinking LED.

Depending on the order of a node in the rack, its corresponding entry is made in the `gssdeploy.cfg` file. For example, for the bottommost node in the rack, its corresponding entry is put first in `gssdeploy.cfg`.

4. This example shows how perform a fresh deployment of an ESS system by running **gssdeploy** in two steps.

```
$ gssdeploy -b
```

The `-b` option does the base xCAT installation.

```
$ gssdeploy -o
```

The `-o` option creates the xCAT object.

The use of **gssdeploy -x** is not recommended, as it does the base xCAT installation every time you run with the `-x` option.

5. This example shows how to set up the kernel, systemd, and Network Manager errata repositories. Use the following command on PPC64BE systems:

```
| /var/tmp/gssdeploy -k /home/deploy/kernel-RHSA-2019-0202-BE.tar.gz -p \  
| /home/deploy/systemd-RHSA-2019-0204-BE.tar.gz,/home/deploy/netmgr-RHSA-2018-3665-BE.tar.gz --silent
```

Note: This command extracts the supplied tar zip files and builds the associated repository.

- `-k` option: Set up the kernel repository
- `-p` option: Set up the patch repository (For example: systemd, network manager). One or more patches might be specified at the same time separated by comma.
- Directory structure:

Kernel repository

```
/install/gss/otherpkgs/rhels7/<arch>/kernel
```

Patch repository

```
/install/gss/otherpkgs/rhels7/<arch>/patch
```

Important: Make sure that all RPMs in the `/install` directory including the extracted files in the kernel directory (`/install/gss/otherpkgs/rhels7/<arch>/kernel`), the patch directory (`/install/gss/otherpkgs/rhels7/<arch>/patch`), and xCAT RPMs, etc. have the correct read permission for user, group, and others (`chmod 644` files). For example:

```
/install/gss/otherpkgs/rhels7/<arch>/kernel  
-rw-r--r-- 1 nobody nobody 44023568 Feb 24 10:17 kernel-3.10.0-862.27.1.el7.ppc64.rpm  
  
/install/gss/otherpkgs/rhels7/<arch>/patch  
-rw-r--r-- 1 nobody nobody 5412240 Oct 05 12:02 systemd-219-57.el7_5.5.ppc64.rpm  
-rw-r--r-- 1 nobody nobody 1899284 Feb 24 10:49 NetworkManager-1.12.0-8.el7_6.ppc64.rpm
```

Wrong file permission will lead to node deployment failure.

6. This example shows how to deploy on the I/O server nodes using the customized deploy script.

```
$ /var/tmp/gssdeploy -d
```

This command starts deploying the I/O server nodes such as starting with disk partitioning, operating system installation, update driver and firmware on I/O server nodes, etc.

7. This examples shows how to use the `-g` option in case of PPC64LE deployment if the xCAT genesis discovery fails to discover the I/O nodes.

```
$ gssdeploy -g
```

This command performs only xCAT genesis discovery.

See also

- “`gssinstall` script” on page 130
- “`gssinstallcheck` command” on page 34

Location

`/opt/ibm/gss/install/samples`

gssinstall script

ESS software packages installer

Synopsis

```
gssinstall [ { -m | --manifest } Manifest ] ]  
          [ { -N | --nodes } NodeList ] [ -s | --silent ]  
          [ -v | --verbose ] [ -V [ -r | --repouupdate ] [ -u | --update ]  
          [ --version ] [ {-t | --tarch } TARCH ]  
          [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

gssinstall is the ESS software packages installer. The actual file name of **gssinstall** is **gssinstall_<arch>** where <arch> can be either ppc64 or ppc64le depending on the architecture of the binary packages you have extracted. It can be found in the /var/tmp directory.

Parameters

- { -m | --manifest } *Manifest***
Specifies a manifest file for version comparison. There is no default.
- { -N | --nodes } *NodeList***
Specifies a comma-separated I/O node list on which to operate. There is no default.
- s | --silent**
Performs the update non-interactively. The default is False.
- r | --repouupdate**
Updates the repository. The default is False.
- u | --update**
Updates the software repository with the specified archive. The default is False.
- v | --verbose**
Specifies the level of detail. The default level is 0. The maximum level is 1.
- V | --version**
Displays the program's version number and exits.
- { -t | --tarch } *TARCH***
Specifies the target architecture of the deployment. The default is the architecture of the running operating system. This option can be used when the extracted ESS software binary is not matching with the running operating system binary. For example, running operating system architecture is ppc64 (Big Endian) and the used binary architecture is ppc64le (Little Endian).
- h | --help**
Displays usage information about this command and exits.

Exit status

0 Successful completion.

nonzero

A failure has occurred.

Security

You must have root authority to run the **gssinstall** command.

Examples

1. This example shows the current version of **gssinstall**.

PPC64 (Big Endian) node

```
gssinstall_ppc64 -V
```

PPC64LE (Little Endian) node

```
gssinstall_ppc64le -V
```

2. This example shows how to compare the existing ESS software repository with the manifest file. The following command compares the repository with the manifest file and shows the list of Not Installed, Currently Installed, New and Older packages. It does not create any new repository or update any existing repository. The **-m** option can be just used to compare the repository with manifest.

```
gssinstall_ppc64 -m /opt/ibm/gss/install/rhel7/ppc64/manifest
```

3. This example shows how to create a new or update an existing software repository of the ESS package installer on PPC64 (Big Endian) type of nodes.

```
gssinstall_ppc64 -u
```

4. This example shows how to create a new or update an existing software repository of the ESS package installer on PPC64LE (Little Endian) type of nodes.

```
gssinstall_ppc64le -u
```

5. This example shows how to refresh an existing repository.

```
gssinstall_ppc64 -r
```

6. This example shows how to create a different ESS software architecture repository.

```
gssinstall_ppc64le -u --tarch ppc64
```

Consider a scenario in which the running operating system architecture is PPC64 and the used ESS binary architecture is PPC64LE. User wants to host the PPC64LE software repository on the PPC64 architecture host. Here, the file name is `gssinstall_ppc64le` as the ESS binary used in this example is of PPC64LE however the running operating system architecture is PPC64.

See also

See also the following *Deploying the Elastic Storage Server* topics:

- “gssdeploy script” on page 125
- “gssinstallcheck command” on page 34

Location

```
/opt/ibm/gss/install/installer
```

gsssnap script

Collects the snapshot of the ESS deployment

Synopsis

```
gsssnap [ { -N | --nodes } NodeList ] [ -g | --gpfs ] [ -V | --version ]  
[ -i | --installcheck ] [ -t | --tarch ] [ -h | --help ]
```

Availability

Available with the Elastic Storage Server.

Description

The **gsssnap** script creates an informational system snapshot at a single point in time. This system snapshot consists of cluster configuration, disk configuration, network configuration, network status, ESS logs, dumps, and traces. Use the **gsssnap** script as one of the main tools to gather preliminary data when an ESS problem is encountered, such as a hung ESS script. The information that is gathered with the **gsssnap** script can be used in conjunction with other information (for example, ESS internal dumps, traces, and kernel thread dumps) to solve an ESS problem.

By default, the **gsssnap** script collects snapshot information from the management server node.

Parameters

- g | --gpfs**
Takes a GPFS snapshot.
- i | --installcheck**
Capture the **gssinstallcheck** output.
- t | --tarch**
Specifies the target architecture of deployment. ppc64 or ppc64le.
- { -N | --nodes } *NodeList***
Specifies a comma-separated list of nodes from which to collect snapshot information.
- V, --version**
Displays the program's version number and exits.
- h | --help**
Displays usage information about this script and exits.

Exit status

0 Successful completion.

nonzero
A failure has occurred.

Security

You must have root authority to run the **gsssnap** script.

Example

This example shows how to collect the snapshot information from I/O server node `gssi01`:

```
gsssnap -N gssi01
```

The system displays output similar to this:


```
# gsssnap -N gssio1,gssio2

gsssnap [INFO]: Collecting xcat snap
gsssnap [INFO]: Collecting sosreports for node(s): ems1,gssio1,gssio2
gsssnap [INFO]: Collecting GSS snap
#####
gsssnap tar file: /tmp/gsssnap.20151013T213843Z.tgz
MD5 file: /tmp/gsssnap.20151013T213843Z.tgz.md5
Please provide tar file to IBM service
#####
```

Location

/opt/ibm/gss/xcat/bin/gsssnap

| **gssupgrade.sh script**

| ESS upgrade supplemental tasks

| **Synopsis**

| `gssupgrade.sh { -a | -b | -s { NodeName | NodeClass } | -c { NodeName | NodeClass } | -h }`

| **Availability**

| Available with the Elastic Storage Server.

| **Description**

| The `gssupgrade.sh` script performs the supplemental tasks for ESS 5.3.2 upgrade.

| **Parameters**

| **-a** Perform all supplemental tasks

| (IO Server node config setting of `gss_ppc64` node class,
| EMS node config setting of `ems` node class,
| perform IBM Spectrum Scale RAID Callback settings,
| copy prepare new disk script)

| **-b** Specifies IBM Spectrum Scale RAID callback settings.

| **-s** { *NodeName* | *NodeClass* }

| Specifies the I/O server node configuration settings, optional node name or node class.

| **-c** { *NodeName* | *NodeClass* }

| Specifies the management server node configuration settings, optional node name or node class.

| **-h** Displays usage information about this script and exits.

| **Exit status**

| **0** Successful completion.

| **nonzero**

| A failure has occurred.

| **Security**

| You must have root authority to run this script.

| **Examples**

| 1. This example shows steps to upgrade IBM Spectrum Scale RAID configuration parameters on the
| EMS node during upgrade.

| `$ /opt/ibm/gss/tools/samples/gssupgrade.sh -b ems1-hs,gss_ppc64`

| `$ /opt/ibm/gss/tools/samples/gssupgrade.sh -c`

| 2. This example shows how to update the node configuration for I/O nodes upgrade.

| `$ /opt/ibm/gss/tools/samples/gssupgrade.sh -s CurrentIoServer-hs`

| This command is run from the EMS node and `CurrentIoServer-hs` must be replaced with the I/O
| server node name.

| **See also**

| See also the following *Elastic Storage Server: Quick Deployment Guide* topic:

- | • *Upgrade the ESS system*

| **Location**

| `/opt/ibm/gss/tools/samples`

Chapter 3. ESS environment variables

This topic includes descriptions of the ESS environment variables.

Table 3. ESS environment variables

Environment variable	Set:	Possible values	Default value
GSENV	To indicate the environment in which you are running - a manufacturing environment or an installation and deployment environment, for example.	INSTALL MFG	

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing IBM Corporation North Castle Drive Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing Legal and Intellectual Property Law IBM Japan Ltd. 19-21,

Nihonbashi-Hakozakicho, Chuo-ku Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Dept. 30ZA/Building 707
Mail Station P300

2455 South Road,
Poughkeepsie, NY 12601-5400
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment or a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Intel is a trademark of Intel Corporation or its subsidiaries in the United States and other countries.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Glossary

This glossary provides terms and definitions for the ESS solution.

The following cross-references are used in this glossary:

- *See* refers you from a non-preferred term to the preferred term or from an abbreviation to the spelled-out form.
- *See also* refers you to a related or contrasting term.

For other terms and definitions, see the IBM Terminology website ([opens in new window](http://www.ibm.com/software/globalization/terminology)):

<http://www.ibm.com/software/globalization/terminology>

B

building block

A pair of servers with shared disk enclosures attached.

BOOTP

See Bootstrap Protocol (BOOTP).

Bootstrap Protocol (BOOTP)

A computer networking protocol that is used in IP networks to automatically assign an IP address to network devices from a configuration server.

C

CEC *See central processor complex (CPC).*

central electronic complex (CEC)

See central processor complex (CPC).

central processor complex (CPC)

A physical collection of hardware that consists of channels, timers, main storage, and one or more central processors.

cluster

A loosely-coupled collection of independent systems, or *nodes*, organized into a network for the purpose of sharing resources and communicating with each other. *See also GPFS cluster.*

cluster manager

The node that monitors node status using disk leases, detects failures, drives recovery, and selects file system

managers. The cluster manager is the node with the lowest node number among the quorum nodes that are operating at a particular time.

compute node

A node with a mounted GPFS file system that is used specifically to run a customer job. ESS disks are not directly visible from and are not managed by this type of node.

CPC *See central processor complex (CPC).*

D

DA *See declustered array (DA).*

datagram

A basic transfer unit associated with a packet-switched network.

DCM *See drawer control module (DCM).*

declustered array (DA)

A disjoint subset of the pdisks in a recovery group.

dependent fileset

A fileset that shares the inode space of an existing independent fileset.

DFM *See direct FSP management (DFM).*

DHCP *See Dynamic Host Configuration Protocol (DHCP).*

direct FSP management (DFM)

The ability of the xCAT software to communicate directly with the Power Systems server's service processor without the use of the HMC for management.

drawer control module (DCM)

Essentially, a SAS expander on a storage enclosure drawer.

Dynamic Host Configuration Protocol (DHCP)

A standardized network protocol that is used on IP networks to dynamically distribute such network configuration parameters as IP addresses for interfaces and services.

E

Elastic Storage Server (ESS)

A high-performance, GPFS NSD solution

made up of one or more building blocks that runs on IBM Power Systems servers. The ESS software runs on ESS nodes - management server nodes and I/O server nodes.

ESS Management Server (EMS)

An xCAT server is required to discover the I/O server nodes (working with the HMC), provision the operating system (OS) on the I/O server nodes, and deploy the ESS software on the management node and I/O server nodes. One management server is required for each ESS system composed of one or more building blocks.

encryption key

A mathematical value that allows components to verify that they are in communication with the expected server. Encryption keys are based on a public or private key pair that is created during the installation process. See also *file encryption key (FEK)*, *master encryption key (MEK)*.

ESS See *Elastic Storage Server (ESS)*.

environmental service module (ESM)

Essentially, a SAS expander that attaches to the storage enclosure drives. In the case of multiple drawers in a storage enclosure, the ESM attaches to drawer control modules.

ESM See *environmental service module (ESM)*.

Extreme Cluster/Cloud Administration Toolkit (xCAT)

Scalable, open-source cluster management software. The management infrastructure of ESS is deployed by xCAT.

F

failback

Cluster recovery from failover following repair. See also *failover*.

failover

(1) The assumption of file system duties by another node when a node fails. (2) The process of transferring all control of the ESS to a single cluster in the ESS when the other clusters in the ESS fails. See also *cluster*. (3) The routing of all transactions to a second controller when the first controller fails. See also *cluster*.

failure group

A collection of disks that share common access paths or adapter connection, and could all become unavailable through a single hardware failure.

FEK See *file encryption key (FEK)*.

file encryption key (FEK)

A key used to encrypt sectors of an individual file. See also *encryption key*.

file system

The methods and data structures used to control how data is stored and retrieved.

file system descriptor

A data structure containing key information about a file system. This information includes the disks assigned to the file system (*stripe group*), the current state of the file system, and pointers to key files such as quota files and log files.

file system descriptor quorum

The number of disks needed in order to write the file system descriptor correctly.

file system manager

The provider of services for all the nodes using a single file system. A file system manager processes changes to the state or description of the file system, controls the regions of disks that are allocated to each node, and controls token management and quota management.

fileset A hierarchical grouping of files managed as a unit for balancing workload across a cluster. See also *dependent fileset*, *independent fileset*.

fileset snapshot

A snapshot of an independent fileset plus all dependent filesets.

flexible service processor (FSP)

Firmware that provides diagnosis, initialization, configuration, runtime error detection, and correction. Connects to the HMC.

FQDN

See *fully-qualified domain name (FQDN)*.

FSP See *flexible service processor (FSP)*.

fully-qualified domain name (FQDN)

The complete domain name for a specific computer, or host, on the Internet. The FQDN consists of two parts: the hostname and the domain name.

G

GPFS cluster

A cluster of nodes defined as being available for use by GPFS file systems.

GPFS portability layer

The interface module that each installation must build for its specific hardware platform and Linux distribution.

GPFS Storage Server (GSS)

A high-performance, GPFS NSD solution made up of one or more building blocks that runs on System x servers.

GSS See *GPFS Storage Server (GSS)*.

H

Hardware Management Console (HMC)

Standard interface for configuring and operating partitioned (LPAR) and SMP systems.

HMC See *Hardware Management Console (HMC)*.

I

IBM Security Key Lifecycle Manager (ISKLM)

For GPFS encryption, the ISKLM is used as an RKM server to store MEKs.

independent fileset

A fileset that has its own inode space.

indirect block

A block that contains pointers to other blocks.

inode The internal structure that describes the individual files in the file system. There is one inode for each file.

inode space

A collection of inode number ranges reserved for an independent fileset, which enables more efficient per-fileset functions.

Internet Protocol (IP)

The primary communication protocol for relaying datagrams across network boundaries. Its routing function enables internetworking and essentially establishes the Internet.

I/O server node

An ESS node that is attached to the ESS storage enclosures. It is the NSD server for the GPFS cluster.

IP See *Internet Protocol (IP)*.

IP over InfiniBand (IPoIB)

Provides an IP network emulation layer on top of InfiniBand RDMA networks, which allows existing applications to run over InfiniBand networks unmodified.

IPoIB See *IP over InfiniBand (IPoIB)*.

ISKLM

See *IBM Security Key Lifecycle Manager (ISKLM)*.

J

JBOD array

The total collection of disks and enclosures over which a recovery group pair is defined.

K

kernel The part of an operating system that contains programs for such tasks as input/output, management and control of hardware, and the scheduling of user tasks.

L

LACP See *Link Aggregation Control Protocol (LACP)*.

Link Aggregation Control Protocol (LACP)

Provides a way to control the bundling of several physical ports together to form a single logical channel.

logical partition (LPAR)

A subset of a server's hardware resources virtualized as a separate computer, each with its own operating system. See also *node*.

LPAR See *logical partition (LPAR)*.

M

management network

A network that is primarily responsible for booting and installing the designated server and compute nodes from the management server.

management server (MS)

An ESS node that hosts the ESS GUI and xCAT and is not connected to storage. It must be part of a GPFS cluster. From a system management perspective, it is the

central coordinator of the cluster. It also serves as a client node in an ESS building block.

master encryption key (MEK)

A key that is used to encrypt other keys. See also *encryption key*.

maximum transmission unit (MTU)

The largest packet or frame, specified in octets (eight-bit bytes), that can be sent in a packet- or frame-based network, such as the Internet. The TCP uses the MTU to determine the maximum size of each packet in any transmission.

MEK See *master encryption key (MEK)*.

metadata

A data structure that contains access information about file data. Such structures include inodes, indirect blocks, and directories. These data structures are not accessible to user applications.

MS See *management server (MS)*.

MTU See *maximum transmission unit (MTU)*.

N

Network File System (NFS)

A protocol (developed by Sun Microsystems, Incorporated) that allows any host in a network to gain access to another host or netgroup and their file directories.

Network Shared Disk (NSD)

A component for cluster-wide disk naming and access.

NSD volume ID

A unique 16-digit hexadecimal number that is used to identify and access all NSDs.

node An individual operating-system image within a cluster. Depending on the way in which the computer system is partitioned, it can contain one or more nodes. In a Power Systems environment, synonymous with *logical partition*.

node descriptor

A definition that indicates how IBM Spectrum Scale uses a node. Possible functions include: manager node, client node, quorum node, and non-quorum node.

node number

A number that is generated and maintained by IBM Spectrum Scale as the cluster is created, and as nodes are added to or deleted from the cluster.

node quorum

The minimum number of nodes that must be running in order for the daemon to start.

node quorum with tiebreaker disks

A form of quorum that allows IBM Spectrum Scale to run with as little as one quorum node available, as long as there is access to a majority of the quorum disks.

non-quorum node

A node in a cluster that is not counted for the purposes of quorum determination.

O

OFED See *OpenFabrics Enterprise Distribution (OFED)*.

OpenFabrics Enterprise Distribution (OFED)

An open-source software stack includes software drivers, core kernel code, middleware, and user-level interfaces.

P

pdisk A physical disk.

PortFast

A Cisco network function that can be configured to resolve any problems that could be caused by the amount of time STP takes to transition ports to the Forwarding state.

R

RAID See *redundant array of independent disks (RAID)*.

RDMA

See *remote direct memory access (RDMA)*.

redundant array of independent disks (RAID)

A collection of two or more disk physical drives that present to the host an image of one or more logical disk drives. In the event of a single physical device failure, the data can be read or regenerated from the other disk drives in the array due to data redundancy.

recovery

The process of restoring access to file

system data when a failure has occurred. Recovery can involve reconstructing data or providing alternative routing through a different server.

recovery group (RG)

A collection of disks that is set up by IBM Spectrum Scale RAID, in which each disk is connected physically to two servers: a primary server and a backup server.

remote direct memory access (RDMA)

A direct memory access from the memory of one computer into that of another without involving either one's operating system. This permits high-throughput, low-latency networking, which is especially useful in massively-parallel computer clusters.

RGD See *recovery group data (RGD)*.

remote key management server (RKM server)

A server that is used to store master encryption keys.

RG See *recovery group (RG)*.

recovery group data (RGD)

Data that is associated with a recovery group.

RKM server

See *remote key management server (RKM server)*.

S

SAS See *Serial Attached SCSI (SAS)*.

secure shell (SSH)

A cryptographic (encrypted) network protocol for initiating text-based shell sessions securely on remote computers.

Serial Attached SCSI (SAS)

A point-to-point serial protocol that moves data to and from such computer storage devices as hard drives and tape drives.

service network

A private network that is dedicated to managing POWER8 servers. Provides Ethernet-based connectivity among the FSP, CPC, HMC, and management server.

SMP See *symmetric multiprocessing (SMP)*.

Spanning Tree Protocol (STP)

A network protocol that ensures a loop-free topology for any bridged

Ethernet local-area network. The basic function of STP is to prevent bridge loops and the broadcast radiation that results from them.

SSH See *secure shell (SSH)*.

STP See *Spanning Tree Protocol (STP)*.

symmetric multiprocessing (SMP)

A computer architecture that provides fast performance by making multiple processors available to complete individual processes simultaneously.

T

TCP See *Transmission Control Protocol (TCP)*.

Transmission Control Protocol (TCP)

A core protocol of the Internet Protocol Suite that provides reliable, ordered, and error-checked delivery of a stream of octets between applications running on hosts communicating over an IP network.

V

VCD See *vdisk configuration data (VCD)*.

vdisk A virtual disk.

vdisk configuration data (VCD)

Configuration data that is associated with a virtual disk.

X

xCAT See *Extreme Cluster/Cloud Administration Toolkit*.

Index

A

audience vii

C

command

- gsscallhomeconf 5
- gsscallhomeevent 7
- gsschenv 13
- gsscrchxml 15
- gssnettest 37
- gssutils 53, 56, 58, 62, 64, 66, 68, 71, 74, 76, 79, 81, 83, 85, 87, 89, 92, 94, 96, 98, 100, 103, 105, 107, 108, 109, 110, 112, 114, 116, 117, 119, 121

commands 1

- gssaddnode 3
- gsscheckdisks 9
- gssfindmissingdisks 17
- gssgencluster 19
- gssgenclusterrgs 22
- gssgenvdisk 30
- gssinstallcheck 34
- gssnodedetails 39
- gssprecheck 42
- gssruntask 45
- gssstoragequickcheck 47
- gssstress 51

comments ix

D

data

- gathering to solve problems 49, 132

documentation

- on web vii

E

environment variables

- GSENV 137

ESS

command

- gssgennetworks 25

commands 1

- gssaddnode 3
- gsscheckdisks 9
- gssfindmissingdisks 17
- gssgencluster 19
- gssgenclusterrgs 22
- gssgenvdisk 30
- gssinstall 130
- gssinstallcheck 34
- gssnodedetails 39
- gssprecheck 42
- gssruntask 45
- gssstoragequickcheck 47
- gssstress 51

scripts 123

- gssdelvdisk 124

ESS (*continued*)

scripts (*continued*)

- gssdeploy 125
- gsssnap 49, 132
- gssupgrade.sh 134
- mtuset 123

G

gssaddnode 3

gsscallhomeconf 5

gsscallhomeevent 7

gsscheckdisks 9

gsschenv 13

gsscrchxml 15

gssdelvdisk 124

gssdeploy script 125

GSENV environment variable 137

gssfindmissingdisks 17

gssgencluster 19

gssgenclusterrgs 22

gssgennetworks command 25

gssgenvdisk 30

gssinstall script 130

gssinstallcheck command 34

gssnettest 37

gssnodedetails 39

gssprecheck 42

gssruntask 45

gsssnap 49, 132

gssstoragequickcheck 47

gssstress 51

gssupgrade.sh script 134

gssutils 53, 56, 58, 62, 64, 66, 68, 71, 74, 76, 79, 81, 83, 85, 87, 89, 92, 94, 96, 98, 100, 103, 105, 107, 108, 109, 110, 112, 114, 116, 117, 119, 121

I

information overview vii

L

license inquiries 139

M

MTU

- changing 123

- mtuset script 123

N

notices 139

O

overview
of information vii

P

patent information 139
preface vii
problems
gathering data to solve 49, 132

R

resources
on web vii

S

scripts 123
gssdelvdisk 124
gssdeploy 125
gssgennetworks 25
gssinstall 130
gsssnap 49, 132
gssupgrade.sh 134
snapshots 49, 132
submitting ix
system snapshots 49, 132

T

trademarks 140

W

web
documentation vii
resources vii



Printed in USA

SC27-9566-00

