

SNMP-based monitoring for GPFS clusters

Set up and verify status, performance, and configuration monitoring for GPFS 3.2 clusters

Skill Level: Intermediate

[David C. Johns \(dcjohns@us.ibm.com\)](mailto:dcjohns@us.ibm.com)

Staff Software Engineer
IBM

[Frank J. Mangione \(mangione@us.ibm.com\)](mailto:mangione@us.ibm.com)

Advisory Software Engineer
IBM

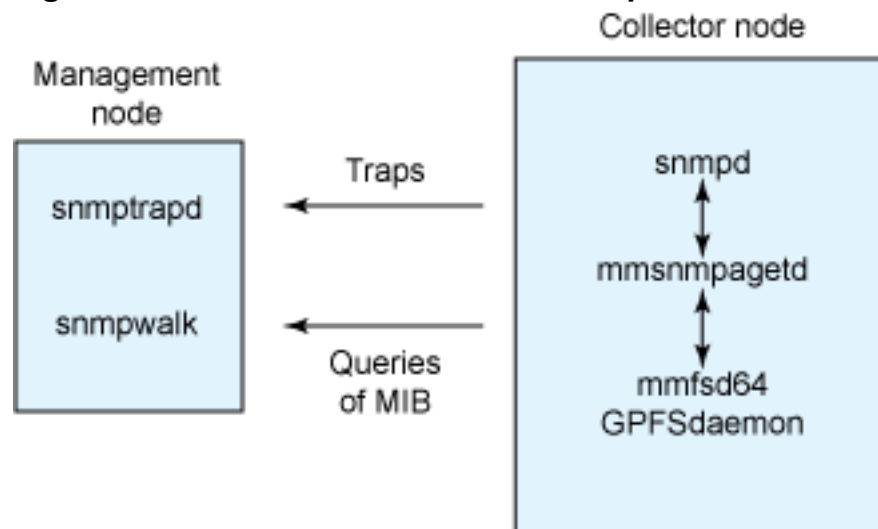
29 Jan 2008

New in version 3.2, IBM General Parallel File System™ (GPFS) on Linux® provides Simple Network Management Protocol (SNMP) services that let administrators collect SNMP data about the health of a GPFS cluster so that problems such as disk failure can be quickly identified. The system lets a collector node gather the trap information, which an administrator can then monitor and analyze remotely on a separate management node. This article provides a method for basic verification of SNMP in a GPFS cluster.

The Simple Network Management Protocol service is available to users of the 3.2 release of the IBM General Parallel File System. It allows users to perform SNMP service collection from a defined Linux OS collector node in the GPFS cluster to a management node. In conjunction with Net-SNMP installed on the collector node and with the mmsnmpagentd service running, the user can capture SNMP trap information for monitoring and analysis on the management node. (Net-SNMP is a suite of applications used to implement SNMP v1, SNMP v2c, and SNMP v3 using both IPv4 and IPv6.) For more information on SNMP in GPFS 3.2, including a complete listing of the types of data that can be collected, refer to the *General Parallel File System Advanced Administration Guide, Version 3 Release 2* (see [Resources](#) for a link).

This article shows you how to verify the SNMP functionality in a GPFS cluster. Figure 1 is an overview of the SNMP verification process.

Figure 1. Overview of SNMP verification process



Prerequisites

To verify the SNMP functionality in a GPFS cluster, you need:

- **Software**
 - GPFS 3.2 or later
 - Net-SNMP (preferably version 5.4.1)
 - Linux
- **Hardware**
 - A node/lpar running Linux in the GPFS cluster as the collector node
 - A management node (also running Linux in this verification procedure)

Setting up and verifying the SNMP agent

The following 12 steps show you how to initially set up and verify the SNMP agent on the collector and manager nodes.

1. Choose the collector node

Pick a GPFS cluster node to be the collector node. This is the node where the

SNMP sub-agent will run, the node that will collect and report GPFS SNMP information to an SNMP management node/application.

2. Choose the management node

Pick a node to be the SNMP management node. This is the node where a sysadmin will run an SNMP management application such as NetView® or OpenNMS. (An off-cluster choice is more realistic, but you could even choose the same node as the collector node.)

3. Install Net-SNMP on nodes

Both the collector node and management node should have Net-SNMP installed:

```
rpm -qa | grep net-snmp.
```

4. Edit the SNMP daemon operating parameters

On the collector node, edit the file `/etc/snmp/snmpd.conf`. This file defines the operating parameters of the master SNMP daemon. Include the following lines:

```
master agentx
trap2sink [HOSTNAME or IP ADDRESS of MANAGEMENT NODE]
AgentXSocket tcp:localhost:705
AgentXTimeout 20
AgentXRetries 10
```

5. Edit general SNMP information

On the collector node and the management node, edit the `/etc/snmp/snmp.conf` file. This file determines general SNMP information for applications on the node. Include the following line: `mibs +GPFS-MIB`.

6. Copy the GPFS MIB definition

On the collector node and the management node, copy the GPFS MIB definition to the directory `/usr/share/snmp/mibs`:

```
cp /usr/lpp/mmfs/data/GPFS-MIB.txt /usr/share/snmp/mibs
rcp /usr/lpp/mmfs/data/GPFS-MIB.txt managementnode:/usr/share/snmp/mibs
```

If the MIB definition file changes in the GPFS build images, repeat this step.

7. Enable the SNMP daemon to grab the new configuration

On the collector node, stop and start the SNMP daemon (also known as the SNMP

master agent or snmpd) so that it picks up the configuration changes:

```
SUSE: /etc/rc.d/snmpd stop
SUSE: /etc/rc.d/snmpd start

Redhat: ps -ef | grep snmpd
Redhat: kill [ /usr/sbin/snmpd PID]
Redhat: /usr/sbin/snmpd
```

Verify that the SNMP daemon is running: `ps -ef | grep snmpd`. Check `dmesg` and `/var/log/snmpd.log` for any bad news.

8. Start receiving SNMP traps

On the management node, open a window and run the command to begin to receive SNMP traps: `/usr/sbin/snmptrapd -Lo -t -f`.

9. Enable the GPFS SNMP sub-agent

On any node in the GPFS cluster, turn on the GPFS SNMP sub-agent: `mmchnode --snmp-agent -N [COLLECTOR-NODE]`.

10. Verify that the sub-agent is running

On the collector node, verify that the GPFS SNMP sub-agent is running: `ps -ef | grep mmsnmpagentd`.

If you don't see it running, make sure GPFS is running, make sure snmpd is running, and check `/var/adm/ras/mmfs.log.latest` for any diagnostic messages.

11. See if you caught anything in your trap

On the management node, see if the `GPFS-MIB::gpfsNewConnectionTrap` trap was caught (it takes about 20 seconds for the sub-agent to collect its initial information).

12. Query the collector node for GPFS SNMP info

On the management node, query the collector node for GPFS SNMP info: `snmpwalk -t 10 -r 10 -c public [COLLECTOR-NODE] ibmGPFS`.

Verify that the information is correct; typical results are shown below.

Results

The following samples demonstrate what you should see. Listing 1 shows output for a typical trap capture with `snmptrapd`; Listing 2 shows typical GPFS cluster

information gathered from `snmpwalk`.

Listing 1. A typical trap capture with `snmptrapd`

```
NET-SNMP version 5.4
2007-10-26 13:29:40 <UNKNOWN> [UDP: [9.114.119.112]:56357]:
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (46843) 0:07:48.43
  SNMPv2-MIB::snmpTrapOID.0 = OID:
GPFS-MIB::gpfsStgPoolUtilizationTrap
  GPFS-MIB::gpfsStgPoolFSName = STRING: "gpfs5"
  GPFS-MIB::gpfsStgPoolName = STRING: "system"
  GPFS-MIB::gpfsStgPoolUtil = Gauge32: 91
2007-10-26 13:31:16
```

Listing 2 shows typical GPFS cluster information gathered from `snmpwalk`.

Listing 2. Typical GPFS cluster information gathered from `snmpwalk`

```
GPFS-MIB::gpfsDiskData."gpfs4"."SP4gpfs1"."GPFSNSD20" = STRING: "y"
GPFS-MIB::gpfsDiskData."gpfs4"."SP4gpfs1"."GPFSNSD21" = STRING: "y"
GPFS-MIB::gpfsDiskData."gpfs4"."SP4gpfs1"."GPFSNSD22" = STRING: "y"
GPFS-MIB::gpfsDiskData."gpfs4"."SP4gpfs1"."GPFSNSD23" = STRING: "y"
GPFS-MIB::gpfsDiskData."gpfs5"."system"."GPFSNSD24" = STRING: "y"
GPFS-MIB::gpfsDiskData."gpfs5"."SP5gpfs1"."GPFSNSD30" = STRING: "y"
GPFS-MIB::gpfsDiskData."gpfsuser"."system"."GPFSNSD28" = STRING: "y"
GPFS-MIB::gpfsDiskData."gpfsuser"."SP5gpfs1"."GPFSNSD26" = STRING: "y"
GPFS-MIB::gpfsDiskData."gpfsuser"."SP5gpfs1"."GPFSNSD27" = STRING: "y"
```

Resources

Learn

- In "[Setting up a multicluster environment using General Parallel File System](#)" (developerWorks, May 2007), learn about the role GPFS plays in clustering.
- The "[Installing a large Linux cluster](#)" series (developerWorks, December 2006 to June 2007) covers Linux cluster topics like GPFS and SNMP.
- The "[High-performance Linux clustering](#)" series (developerWorks, September 2005) looks at high performance computing with Linux systems. Also, the "[Installing a large Linux cluster](#)" series can answer many of your multi-node Linux questions.
- Learn more about [GPFS 3.2](#) in the IBM Cluster information center. In particular, Chapter 7 of the [General Parallel File System Advanced Administration Guide, Version 3 Release 2](#) provides a thorough overview of the SNMP capabilities.
- The [Net-SNMP](#) page on SourceForge includes a [tutorial on using Net-SNMP traps](#) and [documentation on snmpwalk](#) (which lets you retrieve a subtree of management values using SNMP GETNEXT requests).
- In the [developerWorks Linux zone](#), find more resources for Linux developers, and scan our [most popular articles and tutorials](#).
- See all [Linux tips](#) and [Linux tutorials](#) on developerWorks.
- Stay current with [developerWorks technical events and Webcasts](#).

Get products and technologies

- Get [GPFS 3.2](#).
- Get [Net-SNMP](#).
- [Order the SEK for Linux](#), a two-DVD set containing the latest IBM trial software for Linux from DB2®, Lotus®, Rational®, Tivoli®, and WebSphere®.
- With [IBM trial software](#), available for download directly from developerWorks, build your next development project on Linux.

Discuss

- Get involved in the [developerWorks community](#) through blogs, forums, podcasts, and community topics in our [new developerWorks spaces](#).

About the authors

David C. Johns

David C. Johns works as a Staff Software Engineer for IBM in Cluster System test. Prior to this role, he spent three years in Platform Evaluation Test working with SANs and back-end storage devices. His areas of technical expertise include design and implementation of SANs, configuration, and implementation of all forms of back-end storage and testing of GPFS on both the AIX and pLinux operating systems. He has written many field support documents and technical publications in his 20+ years at IBM.

Frank J. Mangione

Frank Mangione is an Advisory Software Engineer working for IBM in GPFS development and test. He has vast experience in the administration of UNIX and its filesystems, including the underlying storage subsystems.