

Rocks | Lightning Hacks | Dojo | Nagios | VirtualBox | Privacy

LINUX JOURNAL

Since 1994: The Original Magazine of the Linux Community

JUNE 2009 | ISSUE 182 | www.linuxjournal.com

Build a
High-Performance
Compute Cluster
with **Rocks**

Using **VirtualBox**,
JumpBox and
Virtual Appliances

AND THE WINNERS ARE...



Your Favorite Gadgets, Programs, Tools, Hardware and More!

**How-To:
Dojo's Grid
Widget**

**Tools for
Validating
HTML**

**Monitoring
Dell Servers
with Nagios**



SNMP Monitoring with Nagios

Using Nagios, you can monitor Dell servers with SNMP via Dell's server administration tools. **JASON ELLISON**

Nagios has been around since 2002 and is considered stable software. It is in use by the likes of American Public Media, JP Morgan Chase and Yahoo, just to name a few. It is an enterprise-level network and systems-monitoring platform. Nagios performs checks of services and hosts using external programs called Nagios plugins.

SNMP (Simple Network Management Protocol) is a network protocol designed for monitoring network-attached devices. It uses OIDs (Object Identifiers) for defining the information, known as MIBs (Management Information Base), that can be monitored. The design is extensible, so vendors can define their own items to be monitored.

OpenManage is provided with Dell servers and is an extremely well-documented system (see Resources) that provides extensive server administration capabilities. OpenManage works with both Linux and Windows. The OpenManage "SNMP Reference Guide" (see Resources) is a 732-page document that is "intended for system administrators, network administrators and anyone who wants to write SNMP MIB applications to monitor systems". The "SNMP Reference Guide" documents the SNMP OIDs/MIBs for monitoring Dell's servers.

The system described here was implemented for a local utility company when it upgraded to Dell Power Edge servers. As often is the case, out of the box, Nagios didn't do exactly what the company needed, but being an open-source project, it easily was extended to accomplish the goal. All we needed was a Nagios plugin to monitor the new servers.

Don't Re-invent the Wheel

The first thing I set out to do was find an existing Nagios plugin that offered similar functionality to what we needed. Quite a number of existing plugins are available. In

The system described here was implemented for a local utility company when it upgraded to Dell Power Edge servers.

less than one hour, I found `check_snmp_temperature.pl` by William Leibzon. This is a plugin module that monitors the temperature of various devices remotely via SNMP. Although monitoring temperatures was not our goal, retrieving information via SNMP and reporting it to Nagios was. The module is written in Perl and after reading it over, it looked very well written.

Chapter 4 of the Dell's "SNMP Reference Guide" is the "System State Group". It states:

The Management Information Base (MIB) variables presented in this section enable you to track various attributes that describe the state of the critical components supported by your system. Components monitored under the System State Group include power supplies, AC power cords, AC power switches, and cooling devices, as well as temperature, fan, amperage, and voltage probes.

The associated OIDs provide the overall state of all the critical subsystems that we were interested in. OIDs exist that provide much greater detail, but in this situation, the requirement was to be alerted only if a server had a problem and to indicate the particular subsystem that had the problem.

One of the benefits to choosing these particular OIDs turned out to be that they all respond in the same format.

One subsystem was not addressed in the "System State Group" chapter—the RAID subsystem. There is, however, an OID for monitoring it. This OID is described in Chapter 23, the "Storage Management Group".

As stated earlier, these OIDs are used to define particular MIBs that can be queried via SNMP. On the Dell server, there is an SNMP server running. The SNMP server answers queries that are in the form of a long string of numbers (the OID). This string of numbers is understood by the SNMP server to be a specific question. For instance, if you want to ask the SNMP server "How are your power supplies?", you would send it the OID `.1.3.6.1.4.1.674.10892.1.200.10.1.9.1` (Figure 1). The SNMP server will respond with 3 if the power supplies are okay.

Table 1 shows the OIDs we are interested in.

One of the benefits to choosing these particular OIDs turned out to be that they all respond in the same format. Dell refers to this format as `DellStatus`, and it maps integers to subsystem states:

Variable Name:	DellStatus
Data Type:	Integer
Possible Data Values	Meaning of Data Value:
other(1)	The object's status is not one of the following:

unknown(2)	The object's status is unknown.
ok(3)	The object's status is OK.
nonCritical(4)	The object's status is warning, noncritical.
critical(5)	The object's status is critical (failure).
nonRecoverable(6)	The object's status is nonrecoverable (dead).

Now that we knew what we wanted to monitor, it was time to modify `check_snmp_temperature.pl` to do what was needed. The result, `check_dell_openmanager.0.7-test.pl`, is too long to print here, but it is available on the *Linux Journal* FTP site (see Resources).

Testing

Because I did not have a spare Dell Power Edge server sitting around to test the modified script, I had to test it another way. Reading the man page for `snmpd.conf`, I found that you could have external programs answer certain OIDs using "pass-through" scripts. The bash script (`dell_open_manager_test.sh`) below serves as my pass-through script for testing. With this script, I can simulate all of the states that the Dell server could be in:

```
#!/bin/bash
#
# bash script to replicate a working Dell OpenManage SNMP agent
# works with Net-SNMP daemon. info@fotek@gmail.com
#

REQUEST_OID="$2"

echo "$REQUEST_OID";
case "$REQUEST_OID" in
    .1.3.6.1.4.1.674.10892.1.200.10.1.4.1)
        echo "integer"; echo "3"; exit 0 ;;
    .1.3.6.1.4.1.674.10892.1.200.10.1.9.1)
        echo "integer"; echo "5"; exit 0 ;;
    .1.3.6.1.4.1.674.10892.1.200.10.1.12.1)
        echo "integer"; echo "3"; exit 0 ;;
    .1.3.6.1.4.1.674.10892.1.200.10.1.21.1)
        echo "integer"; echo "4"; exit 0 ;;
    .1.3.6.1.4.1.674.10892.1.200.10.1.24.1)
        echo "integer"; echo "3"; exit 0 ;;
    .1.3.6.1.4.1.674.10892.1.200.10.1.27.1)
        echo "integer"; echo "3"; exit 0 ;;
    .1.3.6.1.4.1.674.10892.1.200.10.1.38.1)
        echo "integer"; echo "3"; exit 0 ;;
    .1.3.6.1.4.1.674.10892.1.200.10.1.41.1)
        echo "integer"; echo "3"; exit 0 ;;
    .1.3.6.1.4.1.674.10893.1.20.110.13.8)
        echo "integer"; echo "3"; exit 0 ;;
    *)
        echo "string"; echo "50"; exit 0 ;;
esac
exit
```

To use the script, I added the following lines to the end of `/etc/snmp/snmpd.conf`:

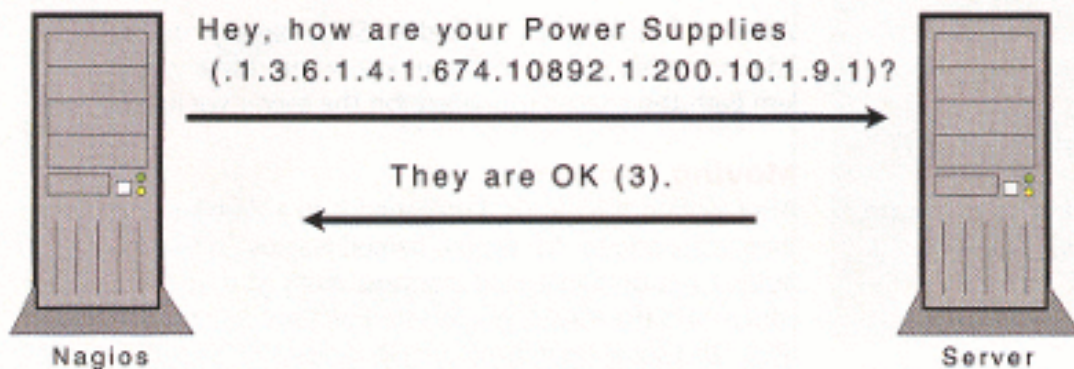


Figure 1. Sample SNMP Query

Table 1. OIDs

Name	Object ID	Description
systemStateChassisStatus	1.3.6.1.4.1.674.10892.1.200.10.1.4	Defines the system status of this chassis.
systemStatePowerSupplyStatusCombined	1.3.6.1.4.1.674.10892.1.200.10.1.9	Defines the status of all power supplies in this chassis.
systemStateVoltageStatusCombined	1.3.6.1.4.1.674.10892.1.200.10.1.12	Defines the status of all voltage probes in this chassis.
systemStateCoolingDeviceStatusCombined	1.3.6.1.4.1.674.10892.1.200.10.1.21	Defines the cooling device status of all cooling devices in this chassis. The result is returned as a combined status value. The value has the same definition type as DellStatus.
systemStateTemperatureStatusCombined	1.3.6.1.4.1.674.10892.1.200.10.1.24	Defines the status of all temperature probes in this chassis. The result is returned as a combined status value. The value has the same definition type as DellStatus.
systemStateMemoryDeviceStatusCombined	1.3.6.1.4.1.674.10892.1.200.10.1.27	Defines the status of all memory devices in this chassis.
systemStateChassisIntrusionStatusCombined	1.3.6.1.4.1.674.10892.1.200.10.1.30	Defines the intrusion status of all intrusion-detection devices in this chassis. The result is returned as a combined status value. The value has the same definition type as DellStatus.
systemStateEventLogStatus	1.3.6.1.4.1.674.10892.1.200.10.1.41	Defines the overall status of this chassis (ESM) event log.
agentGlobalSystemStatus	1.3.6.1.4.1.674.10893.1.20.110.13	Global health information for the subsystem managed by the Storage Management software. This global status should be used by applications other than HP OpenView. HP OpenView should refer to the globalStatus in the root level object group. This is a rollup for the entire agent including any monitored devices. The status is intended to give initiative to an SNMP monitor to get further data when this status is abnormal.

```
### dell open manager test
view systemview included .1.3.6.1.4.1.674
pass .1.3.6.1.4.1.674 /bin/bash \
    /usr/local/bin/dell_open_manager_test.sh
```

To make the changes in the configuration file take effect, restart the `snmpd` daemon. On Slackware, this is done via the following:

```
# /etc/rc.d/rc.snmpd restart
Shutting down snmpd: . DONE
Starting snmpd: /usr/sbin/snmpd -A -p \
    /var/run/snmpd -a -c /etc/snmp/snmpd.conf
```

To query the SNMP server, we use Net-SNMP's command-line `snmpget` utility:

```
# snmpget -v 1 -c public 127.0.0.1 \
    .1.3.6.1.4.1.674.10892.1.200.10.1.9.1
SNMPv2-SMI::enterprises.674.10892.1.200.10.1.9.1 = INTEGER: 3
```

The response is an integer value of 3. The value 3 in the DellStatus (see above) maps to "ok(3) The object's status is OK". This tells us that the pass-through script is working. Now, we test the `/check_dell_openmanager.pl` Perl script:

```
# ./check_dell_openmanager.pl -H 127.0.0.1 -C public -T pe2950
OK
```

To test other values, simply modify the `dell_open_manager_test.sh` shell script. For example, to simulate an error in the Cooling Device OID (.1.3.6.1.4.1.674.10892.1.200.10.1.21), modify that OID's line in the script to return a code of 4 for nonCritical:

```
.1.3.6.1.4.1.674.10892.1.200.10.1.21.1)
    echo "integer"; echo "4"; exit 0 ;;
```

Now, running the Perl script produces a warning:

```
# ./check_dell_openmanager.pl -H 127.0.0.1 -C public -T pe2950
WARNING:Cooling Device Status=Non-Critical
```

To simulate a critical error, let's modify the Power Supply OID to reply with a 5:

```
.1.3.6.1.4.1.674.10892.1.200.10.1.9.1)
    echo "integer"; echo "5"; exit 0 ;;
```

```
# ./check_dell_openmanager.pl -H 127.0.0.1 -C public -T pe2950
CRITICAL:Cooling Device Status=Non-Critical. \
    Power Supply Status=Critical
```

To test the script on the live production systems, we added the `check_dell_openmanager.pl` command to a working Nagios server. We opened the case cover on a live system to generate a Chassis Intrusion Status error to test the plugin.

Within a few seconds, we had an SMS message on the IT administrator's phone letting us know that there was a problem with the chassis subsystem on the server we just opened.

Moving Forward

After writing this plugin, I uploaded it to a Web site that hosts third-party addons for Nagios named Nagios Exchange. In short order, I was getting e-mail messages from all over the world concerning the Nagios plugin I had written. Some were suggestions, and some were from people in need of help. It was not an overwhelming number of messages. At most, two a week and sometimes none. It was just enough to let me know that people other than me actually were using this thing.

I would like to make a few improvements to the module. For one, I think there may be a way to reduce the SNMP queries to only one query to obtain the overall global status of the machine. Then, only if the state is not "ok(3)", move to query the other OIDs so that a more specific error can be reported.

It also would be nice to be able to evaluate the existence of the various subsystems, that way, for example, if a machine has a RAID array, it is monitored, and if not, the script skips it.

One of the most common e-mail messages I get is about missing the Net::SNMP Perl module. I would like to test for these common-case scenarios. If the test fails, I would like to print the problem with a solution. In the case of "Net::SNMP", it should print:

```
You are missing the Net::SNMP perl module.
Please install it using:
perl -MCPAN -e shell
cpan> install "Net::SNMP"
```

This would improve end-user experience significantly, especially for users new to Linux. ■

Jason Ellisen (www.jasonellison.net/) is an independent IT consultant in the Gulf Coast area. He can be reached at infotek@gmail.com.

Resources

`check_dell_openmanager.0.7-test.pl`: ftp.linuxjournal.com/pub/lj/listings/issue182/10204.tgz

Nagios: www.nagios.org

`check_dell_openmanager`: www.nagiosexchange.org/cgi-bin/page.cgi?g=1437.html

Dell OpenManage Server Administrator Documentation: support.dell.com/support/edocs/software/svradmin

Dell OpenManage Server Administrator Version 5.4 SNMP Reference Guide: support.dell.com/support/edocs/software/svradmin/5.4/en/snmp/pdf/om_54_snmp_ref_gd.pdf