Volume 3, Issue 9, September 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Agent-Less Hardware Monitoring Tool for Cluster Management

Shival Dubey[*]
Mechanical Engg, U.P.E.S.
India

Anant Wadhwa
Mechanical Engg, U.P.E.S.
India

Subodh Kumar
Mechanical Engg, V.V.I.T.
India

Vineet Virmani
Computer Sc. Engg, U.P.E.S.
India

Vaibhav Hans
Computer Sc. Engg, U.P.E.S.
India

Abstract— In a high performance cluster environment, usage of an agent based monitoring will consume CPU cycle and memory footprint. Currently host OS monitoring and hardware monitoring are done separately. Hardware monitoring can be done using the IPMI stack. Host OS monitoring can be done using Nagios, Ganglia or a similar software. In order to develop an agent free monitoring tool for HPC cluster monitoring, we integrated hardware monitoring tool IPMI (Intelligent Platform Management Interface) stack with Nagios. The integrated framework abstracts the hardware monitoring of the server to the Nagios GUI web interface.

Keywords— Hardware monitoring, Agentless, NagiOS, Web interface, IPMI

I. Introduction

High Performance computing (HPC) used compute clusters to solve large computational problems. The user space applications are getting hungrier for more computation power. With the computation power of today's computer advancing in the order of Teraflops and each bit of computation power being squeezed out to suffice the requirements of the user space application. The existing technique to monitor a high performance computing cluster is to use an agent based monitoring tool for hardware monitoring in conjunction with OS monitoring applications like Nagios. This essentially means that one would have to use two monitoring interfaces. This would also incur an expense of CPU cycles and memory footprint to be used for monitoring which could have been otherwise used by user space application.

We have implemented the framework to integrate Nagios and IPMI by following a sequence of steps including changes to the access control permissions to certain files and configurations. Following the below detailed steps, a user will be able to monitor hardware parameters (viz. fan speed, temperature, voltage, current, power consumption etc.) from within Nagios.

II. Components Used

- ROCKS 5.4.
- CENTOS 5.5 64 BIT.
- X86 ARCHITECTURE BASED SERVER.
- NAGIOS.
- FREEIPMI.

A. Nagios

Nagios is a powerful monitoring system that enables organizations to identify and resolve IT infrastructure problems before they affect critical business processes. Designed with scalability and flexibility in mind, Nagios gives you the peace of mind that comes from knowing your organization's business processes won't be affected by unknown outages.

Nagios is a powerful tool that provides you with instant awareness of your organization's mission critical IT infrastructure. Nagios allows you to detect and repair problems and mitigate future issues before they affect end-users and customers.

Nagios is a popular open source computer system and network monitoring software application. It watches hosts and services, alerting users when things go wrong and again when they get better. Nagios monitors your entire IT infrastructure to ensure systems, applications, services, and business processes are functioning properly. In the event of a failure, Nagios can alert technical staff of the problem, allowing them to begin remediation processes before outages affect business processes, end-users, or customers. With Nagios you'll never be left having to explain why a unseen infrastructure outage hurt your organization's bottom line.

B. Rocks

Rocks Cluster Distribution (originally called NPACI Rocks) is a Linux distribution intended for high performance computing clusters. It was started by National Partnership for Advanced Computational Infrastructure and the SDSC in 2000 and was initially funded in part by an NSF grant (2000-2007) but is currently funded by the follow-up NSF grant.

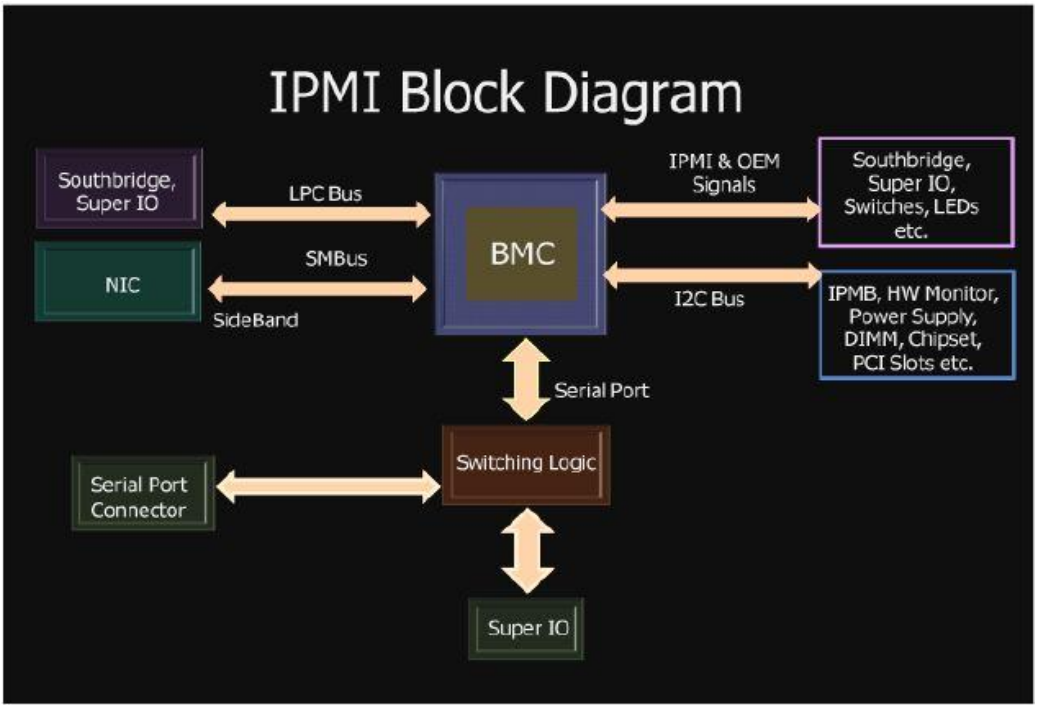
Rocks was initially based on the Red Hat Linux distribution, however modern versions of Rocks are now based on CentOS, with a modified Anaconda installer that simplifies mass installation onto many computers. Rocks include many tools (such as MPI) which are not part of CentOS but are integral components that make a group of computers into a cluster. The advantage of using Rocks to build and maintain your cluster is simple. Building clusters is straightforward, but managing its software can be complex. This complexity becomes most unmanageable during cluster installation and expansion. “Rocks” provides mechanisms to control the complexity of the cluster installation and expansion process.

C. FreeIPMI

The Intelligent Platform Management Interface (IPMI) is a standardized computer system interface used by system administrators to manage a computer system and monitor its operation. FreeIPMI provides in-band and out-of-band IPMI software based on the IPMI specification.

An IPMI sub-system operates independently of the operating system and allows administrators to manage a system remotely in the absence of an operating system or of the system management software. The monitored system may be powered off, but must be connected to a power source and the monitoring medium, typically a local area network connection. IPMI can also function after the operating system has started, and exposes management data and structures to the system management software. IPMI prescribes only the structure and format of the interfaces as a standard, while detailed implementations may vary.

The features of IPMI that most users will be interested in are sensor monitoring, remote power control, Serial-Over-LAN (SOL), and system debugging.



III. Steps To be Followed

1) We have to setup a cluster, for cluster setup first install Cent OS 5.5 using Rocks 5.4 with appropriate roles in master node.

2) Use “insert-ethers” command to insert compute nodes.

3) Download Nagios from http://www.nagios.org/download/ and install it by running “./configure” on master node.

4) Download FreeIPMI from http://www.gnu.org/s/freeipmi/download.html/ and install it by running “./configure” on master node.

5) Edit
“/usr/local/nagios/etc/objects/commands.cfg” as

```
“define command{
command_name ipmi_sensor
command_line $USER1$/ipmi_sensor -H
localhost -T fan,voltage -O '--legacy-output --
interpret-oem-data'
}”
```

6) Edit
“ usr/local/nagios/etc/objects/localhost.cfg" . as

```
“define host{
use linux-server
host_name localhost
alias localhost
address 127.0.0.1
_ipmi_ip 192.168.0.120
}
# 192.168.0.120 is the address of ipmi which can be found by "ipmitool lan print" command.
define service{
use generic-service
host_name localhost
service_description IPMI
check_command ipmi_sensor
}
# Note: check_command in service or localhost file should be same as the comand_name in commands.cfg file.
```

7) Edit "/usr/local/nagios/etc/resource.cfg" as

```
$USER1$=/usr/local/nagios/libexec/
# this is the path of all plugins, you have to copy the "ipmi_sensors" script to this path
$USER2$=/usr/local/nagios/libexec/eventhandlers/
# path to eventhandlers which can be located by "locate eventhandler" command and choose the appropriate one.
$USER3$=root
$USER4$=upes01
# in the respective file changes the username with appropriate username and password with appropriate password and you have to uncomment the quotes
```

8) For remote connection use the address of the remote device and use it as –H <address> with script. In commands .cfg, use $HOSTADDRESS$ as hostname (-H $HOSTADDRESS$) and provide username, password and privilege level in that. In localhost.cfg, define a new host in which the "address" and "_ipmi_ip" is replaced by the address of the remote device (BMC) and in service replace the hostname with the newly created host with the above told configuration

IV. CONCLUSIONS

The version of this template is V2. Most of the formatting instructions in this document have been compiled by Causal Productions from the IEEE LaTeX style files. Causal Productions offers both A4 templates and US Letter templates for LaTeX and Microsoft Word. The LaTeX templates depend on the official IEEEtran.cls and IEEEtran.bst files, whereas the Microsoft Word templates are self-contained. Causal Productions has used its best efforts to ensure that the templates have the same appearance.

V. "IPMI_SENSORS" SCRIPT

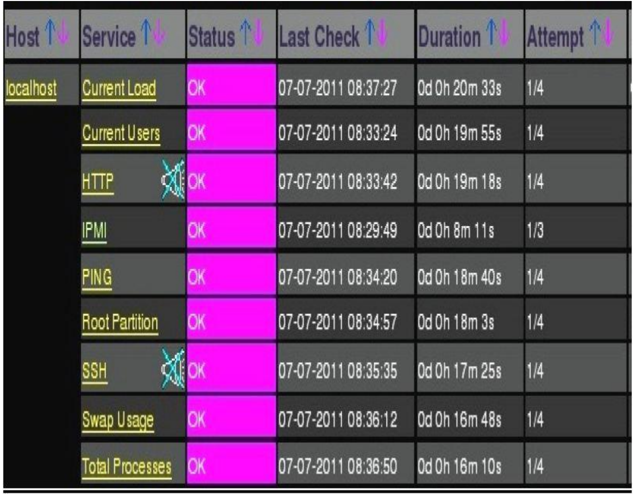
```
## For getting the output in IPMICOMMAND variable.
IPMICOMMAND="/usr/local/sbin/ipmimonitoring"
# # Remove extra spaces and give the output in exact format.
gsub(/ +$/,"",$1)
gsub(/^ +/,"",$2)
gsub(/ +$/,"",$2)
gsub(/^ +/,"",$4)
gsub(/ +$/,"",$4)
gsub(/^ +/,"",$5)
gsub(/ +$/,"",$5)
gsub(/^ +/,"",$6)
gsub(/ +$/,"",$6)
sensor_id[NR]=$1
sensor_name[NR]=$2
# sensor_type[NR]=$3 (currently not used)
sensor_status[NR]=$4
sensor_units[NR]=$5
sensor_reading[NR]=$6
```

VI. NAGIOS AND FREEIPMI INTEGRATION

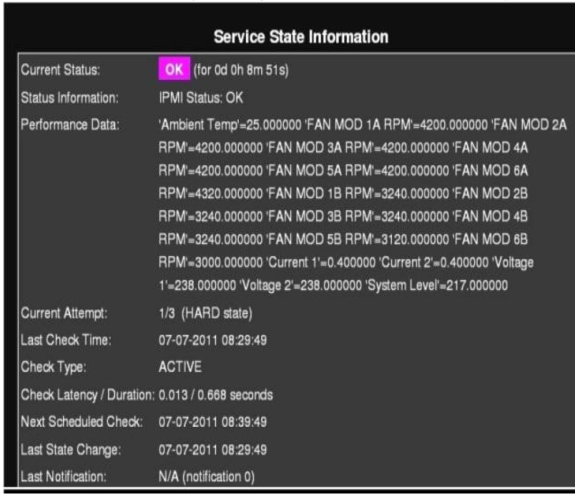
Nagios is a powerful monitoring system which is used to monitor a cluster and resolving their problems.

FreeIPMI is used to monitor hardware parameters; we can monitor those parameters remotely by providing BMC (Baseboard Management Controller) address. After integration we can get the hardware parameters status and value within Nagios using BMC address. By using BMC address, values of hardware components like fan speed, temperature, voltage, current, power consumption etc. are fetched and displayed on Nagios with proper status.

We can access hardware parameters of compute nodes in master node by using BMC address of compute node.



Host	Service	Status	Last Check	Duration	Attempt
localhost	Current Load	OK	07-07-2011 08:37:27	0d 0h 20m 33s	1/4
	Current Users	OK	07-07-2011 08:33:24	0d 0h 19m 55s	1/4
	HTTP	OK	07-07-2011 08:33:42	0d 0h 19m 18s	1/4
	IPMI	OK	07-07-2011 08:29:49	0d 0h 8m 11s	1/3
	PING	OK	07-07-2011 08:34:20	0d 0h 18m 40s	1/4
	Root Partition	OK	07-07-2011 08:34:57	0d 0h 18m 3s	1/4
	SSH	OK	07-07-2011 08:35:35	0d 0h 17m 25s	1/4
	Swap Usage	OK	07-07-2011 08:36:12	0d 0h 16m 48s	1/4
	Total Processes	OK	07-07-2011 08:36:50	0d 0h 16m 10s	1/4



Service State Information

```
Current Status:              OK (for 0d 0h 8m 51s)
Status Information:          IPMI Status: OK
Performance Data:            'Ambient Temp'=25.000000 'FAN MOD 1A RPM'=4200.000000 'FAN MOD 2A
                             RPM'=4200.000000 'FAN MOD 3A RPM'=4200.000000 'FAN MOD 4A
                             RPM'=4200.000000 'FAN MOD 5A RPM'=4200.000000 'FAN MOD 6A
                             RPM'=4320.000000 'FAN MOD 1B RPM'=3240.000000 'FAN MOD 2B
                             RPM'=3240.000000 'FAN MOD 3B RPM'=3240.000000 'FAN MOD 4B
                             RPM'=3240.000000 'FAN MOD 5B RPM'=3120.000000 'FAN MOD 6B
                             RPM'=3000.000000 'Current 1'=0.400000 'Current 2'=0.400000 'Voltage
                             1'=238.000000 'Voltage 2'=238.000000 'System Level'=217.000000
Current Attempt:             1/3  (HARD state)
Last Check Time:             07-07-2011 08:29:49
Check Type:                  ACTIVE
Check Latency / Duration:    0.013 / 0.668 seconds
Next Scheduled Check:        07-07-2011 08:39:49
Last State Change:           07-07-2011 08:29:49
Last Notification:           N/A (notification 0)
```

A) Nagios with IPMI Monitoring Tool

B) IPMI Monitoring Tool's Output

By using this IPMI tool we can get the fan speed, temperature, voltage, current, power consumption etc. within Nagios.

VII. CONCLUSIONS

We successfully integrated agent free hardware monitoring tool with Nagios and we are able to get the dynamic output of fan speed, temperature, voltage, current, power consumption etc. within Nagios. We verified the output of IPMI and Nagios with FreeIPMI. A similar framework can be used for other host OS monitoring applications

ACKNOWLEDGMENT

We would like to thank Mr. Ramesh Rajagopalan, Director Systems Engineering, Dell India R&D Centre, Bangalore for giving us the opportunity to carry out this work at Dell's facility. The help extended to us by Mr. Aman Wadhwa, Security Analyst, A.T. &T., Bangalore was highly appreciated. Also we would like to extend our thanks to the faculty and staff at the Information Technology Department and Department of Electronics and Electrical Engineering at University of Petroleum and Energy Studies, Dehradun for their tremendous support and friendly spirit that kept us going for this research.

REFERENCES

1) http://www.thomaskrenn.com/en/oss/ipmi-plugin/
2) http://www.nagios.org/download/
3) www.gnu.org/s/freeipmi/download.html/
4) Chris Burgess (2005), "The Nagios Book".
5) www.makelinux.mo.il/man/7/f/freeipmi/
6) Compiler Transformations for High-Performance Computing, D.F. Bacon, S.L. Graham, O.J. Sharp, ACM Computing Surveys (CSUR), 1994