Technical Guide

Contents

Integrating Compaq Management Agents with Open Source Monitoring Tools

Abstract: This document explains how to integrate *Compaq Management Agents for Linux* with the popular Open Source SNMP monitoring tools Net-SNMP, MRTG and RRDTOOL.

This technical guide allows enterprises and service providers already familiar with these tools to quickly integrate monitoring of their Compaq *ProLiant*™ systems running Linux.

Notice

This publication does not constitute an endorsement of the product or products that were tested. The configuration or configurations tested or described may or may not be the only available solution. This test is not a determination of product quality or correctness, nor does it ensure compliance with any federal, state or local requirements.

Compaq, the Compaq logo, ActiveAnswers, and ProLiant are trademarks of Compaq Information Technologies Group, L.P. in the U.S. and/or other countries.

All other product names mentioned herein may be trademarks of their respective companies.

Compaq shall not be liable for technical or editorial errors or omissions contained herein. The information is provided "as is" without warranty of any kind and is subject to change without notice. The warranties for Compaq products are set forth in the express limited warranty statements accompanying such products. Nothing herein should be construed as constituting an additional warranty.

Introduction

This document explains how to integrate *Compaq Management Agents (CMA) for Linux* into the popular Open Source SNMP monitoring tools Net-SNMP, MRTG and RRDTOOL.

Solution Overview

Compaq provides a number of management agents that allow for *ProLiant*-specific information to be made available through an industry standard SNMP interface. Examples of the *ProLiant*-specific information include temperature levels on the motherboard, fan operation, as well as information about the operating of the system such as CPU load and running processes.

This document will identify:

- Where to obtain the *Compaq Management Agents for Linux* kit and installation guide.

- Where to obtain the *Compaq Management Agents* MIBs and how to integrate them into Net-SNMP.

- Once the agents are installed, how to add get system information using Net-SNMP.

- Examples of MRTG and RRDTOOL showing how to monitor and graph *Compaq Management Agent*-specific information such as temperature, CPU, and memory usage.

Obtaining Compaq Management Agents (CMA) for Linux

The best starting place for information about the *Compaq Management Agents* for Servers is at http://www.compaq.com/manage. For detailed information about the Agents dependencies and where to download them, go to the Compaq support website located at http://www.compaq.com/support/files/server/us/index.html.

From the support website, you will be presented with a list of various downloadable packages. You will need to download at least the following management agents packages and README files for:

- *Compaq Foundation Agents for Servers – Linux (cpqfdtn)*

- *Compaq NIC Agents for Servers – Linux (cmanic)*

- *Compaq Server Agents for Servers - Linux (cmasvr)*

- *Compaq Storage Agents for Servers - Linux (cmastor)*

Also, under Management applications and Utilities, download:

- *Health and Wellness Driver and the Compaq Integrated Management Log (IML) Viewer Application (cpqhealth)*

Also download the *How to Install and Run Compaq Management Agents for Servers for Linux* guide. There is a link on each of the download pages for each of the agents, or the document can be downloaded directly from http://www.compaq.com/products/servers/management/agents-description.html.

Quick Installation Instructions

This section is intended for advanced Linux administrators who are already familiar with the shell command line and using RPM.

For complete documentation, please refer to the *How to Install and Run Compaq Management Agents for Servers for Linux* guide.

Note: The following instructions reference specific versions of the RPM packages. Newer versions may be available. Please check the Compaq Software & Drivers website: http://www.compaq.com/support/files/server/us/index.html

1. Download, from the *ProLiant* server section of the Compaq Software & Drivers site (http://www.compaq.com/support/files/server/us/index.html), the *Compaq Health & Wellness* driver. This driver creates the **/proc** device area for the Compaq system health information.

   ```
   # rpm -i cpqhealth-2.3.0-20.Redhat7_2.i386.rpm
   ```

2. Verify that the driver has properly loaded by looking for a number of new files in the **/proc** filesystem:

   ```
   # ls /proc/cpq*
       /proc/cpqfan  /proc/cpqnvr  /proc/cpqpow
       /proc/cpqiml  /proc/cpqpci  /proc/cpqtmp
   ```

3. Check to see if the standard **ucd-snmp** package is already installed on your system:

   ```
   # rpm –q ucd-snmp
   ```

4. If it is not already installed, install the package from the Red Hat distribution CD. Depending on which RPM packages were already installed, there may be other packages you need to install to satisfy the dependencies that the ucd-snmp package has.

   ```
   # rpm –i /mnt/cdrom/RedHat/RPMS/ucd-snmp-4.2.1-7.i386.rpm
   ```

5. Now, also install the source package for **ucd-snmp** from the Red Hat sources CD, if not previously installed:

   ```
   # rpm -i /mnt/cdrom/RedHat/RPMS/ucd-snmp-4.2.1-7.src.rpm
   ```

6. In order to verify that the *Compaq Management Agents* are properly installed, it is useful to use the SNMP command line tools. They are provided by the Red Hat **ucd-snmp-utils** package. Use the following command and install the package from the Red Hat distribution CD.

   ```
   # rpm -i /mnt/cdrom/RedHat/RPMS/ucd-snmp-utils-4.2.1-7.i386.rpm
   ```

7. Install the Compaq Foundation Class Agent package. This agent receives the information from the various health and wellness drivers and forwards that information to the SNMP daemon. It also provides the Compaq Management Web server listening on port 2301.

   ```
   # rpm –ih cmafdtn-5.30-8.i386.rpm
   ```

8. Next, the standard ucd-snmp package needs to be patched in order to integrate the Compaq Foundation Class Agent. The previous package installed a shell script to automate the process. It will patch, rebuild and asks to install the new modified SNMP daemon (snmpd). Be sure to answer **yes** when prompted.

    ```
    # /opt/compaq/foundation/etc/snmpdbld
    ```

9. Copy the **snmpd.conf** file provided by the ucd-snmp package to **/usr/share/snmp/**

    ```
    # cp /etc/snmp/snmpd.conf /usr/share/snmp/snmpd.conf
    ```

10. Edit **/usr/share/snmp/snmpd.conf** and change the following line from:

    ```
    view systemview included system
    ```

    to

    ```
    view systemview included .1
    ```

    Add the following line to allow read/write access to the SNMP agents from only the local system. Be sure to use a secure password in the appropriate place below.

    ```
    rwcommunity YourWritePassword 127.0.0.1
    rocommunity public 127.0.0.1
    ```

    If it exists in the file, remove or comment out the line that looks like this:

    ```
    pass .1.3.6.1.4.1.4413.4.1 /usr/bin/ucd5820stat
    ```

11. Restart the SNMP daemon to load the new configuration file:

    ```
    # killall –HUP snmpd
    ```

12. Download and install the remaining *Compaq Management Agents*. These provide detailed information about the running host system as well as network and storage devices.

    ```
    # rpm -i cmanic-5.30.0-3.i386.rpm cmastor-5.30.0-10.i386.rpm \
             cmasvr-5.30.0-3.i386.rpm
    ```

13. Start the *Compaq Management Agents*:

    ```
    # /etc/rc.d/init.d/cmafdtn restart
    # /etc/rc.d/init.d/cmastor restart
    # /etc/rc.d/init.d/cmanic restart
    # /etc/rc.d/init.d/cmasvr restart
    ```

14. To verify that the *Compaq Management Agents* are fully working properly, use a Web browser and go to the URL http://localhost:2301/. Login using a username of **administrator** and password **administrator**.

15. An alternative way is to use the SNMP command line tool `snmpwalk` to display SNMP information. From a terminal window, enter the following command which will display many lines of information.

    ```
    # snmpwalk localhost public enterprises.232
    ```

Note: You may have to reconfigure any firewall software running on your Linux system in order to allow SNMP queries and traps to be sent and received. For Red Hat 7.2 systems, use the **lokkit** command and allow access to the following ports for udp: **161** (snmp), **162** (snmptrap) and for tcp on port 2301 (compaq-webm).

Integrating Compaq Management Agents MIBs with Net-SNMP

The MIBs for the *Compaq Management Agents* are available on the Compaq Management CD in the Server Setup and Management package that ships with every *ProLiant* system. Mount the CD and copy the files from the **toolkit/mibs** directory to **/usr/share/snmp/mibs**.

```
# mount /mnt/cdrom
# cp /mnt/cdrom/toolkit/mibs/*.mib /usr/share/snmp/mibs/
# umount /mnt/cdrom
```

By default, the SNMP tools do not display the textual name of OIDs of newly installed MIBs. To have the textual name displayed, add the -**m all** option to SNMP commands. For example, to obtain information from the *Compaq Management Agents* on a host named proliant.exampledomain.com, use the following command.

```
# snmpwalk -m all proliant.exampledomain.com public\
               enterprises.compaq
enterprises.compaq.cpqStdEquipment.cpqSeMibRev.cpqSeMibRevMajor.0 = 1
enterprises.compaq.cpqStdEquipment.cpqSeMibRev.cpqSeMibRevMinor.0 = 12
enterprises.compaq.cpqStdEquipment.cpqSeMibRev.cpqSeMibCondition.0 = ok(2)
enterprises.compaq.cpqStdEquipment.cpqSeComponent.cpqSeInterface.cpqSeOsCommon.c
pqSeOsCommonPollFreq.0 = 30
[…many lines…]
```

It is frequently useful to know the numerical value for a particular OID for which you know the name. In the following example, the textual representation of the temperature sensor OIDs is known and the numerical representation is wanted:

```
# snmpwalk –m all proliant.exampledomain.com public \
       enterprises.compaq.cpqHealth.cpqHeComponent.cpqHeThermal.cpqHeTemperature
Table.cpqHeTemperatureEntry.cpqHeTemperatureCelsius.1
enterprises.compaq.cpqHealth.cpqHeComponent.cpqHeThermal.cpqHeTemperatureTable.c
pqHeTemperatureEntry.cpqHeTemperatureCelsius.1.1 = 34
enterprises.compaq.cpqHealth.cpqHeComponent.cpqHeThermal.cpqHeTemperatureTable.c
pqHeTemperatureEntry.cpqHeTemperatureCelsius.1.2 = 33
enterprises.compaq.cpqHealth.cpqHeComponent.cpqHeThermal.cpqHeTemperatureTable.c
pqHeTemperatureEntry.cpqHeTemperatureCelsius.1.3 = 34
enterprises.compaq.cpqHealth.cpqHeComponent.cpqHeThermal.cpqHeTemperatureTable.c
pqHeTemperatureEntry.cpqHeTemperatureCelsius.1.4 = 34
```

```
# snmpwalk –O n –m all proliant.exampledomain.com public \
       enterprises.compaq.cpqHealth.cpqHeComponent.cpqHeThermal.cpqHeTemperature
Table.cpqHeTemperatureEntry.cpqHeTemperatureCelsius.1
.1.3.6.1.4.1.232.6.2.6.8.1.4.1.1 = 34
.1.3.6.1.4.1.232.6.2.6.8.1.4.1.2 = 33
.1.3.6.1.4.1.232.6.2.6.8.1.4.1.3 = 34
```

Using MRTG to Monitor and Graph CMA-specific Information

Multi Router Traffic Grapher (MRTG) is a very popular Open Source package that uses SNMP to monitor network devices and generate HTML and GIF images that let you visually monitor those devices. While it was originally written to monitor routers, it is frequently used to monitor other SNMP enabled devices. Here, the reader is shown how to monitor the four temperature sensors that are on the motherboard of Compaq *ProLiant* servers.

If the MRTG package is not already installed on your system, visit http://www.mrtg.org and download and install the latest kit.

Once you have MRTG installed, you need to create a configuration file that specifies the OID and system to monitor. Note that the OIDs in the example are the same as discovered in the section titled *Integrating Compaq Management Agents MIBs with Net-SNMP*. Below is a complete sample configuration file, **cpqinfo.cfg,** to monitor the temperature sensors. The lines that need to be modified to suit your configuration are in italics.

```
WorkDir: /home/someuser/public_html/mrtg

Target[localhost.temp]:
1.3.6.1.4.1.232.6.2.6.8.1.4.1.1&1.3.6.1.4.1.232.6.2.6.8.1.4.1.2:public@proliant.
exampledomain.com
MaxBytes[localhost.temp]: 200
Options[localhost.temp]: gauge
Title[localhost.temp]: proliant.exampledomain.com: 2 Temperature Sensors
YLegend[localhost.temp]: Temperature (C)
LegendI[localhost.temp]:  CPUSensorOne:
LegendO[localhost.temp]:  IOBoardSensor:
PageTop[localhost.temp]: <H1>Temperature Sensors for
proliant.exampledomain.com</H1>
   <TABLE>
      <TR><TD>System:</TD>       <TD>proliant.exampledomain.com lab, rack
3</TD></TR>
      <TR><TD>Maintainer:</TD> <TD>someuser@exampledomain.com</TD></TR>
      <TR><TD>Ip:</TD>           <TD>xx.xx.xx.xx</TD></TR>
  </TABLE>

Target[localhost.moretemp]:
1.3.6.1.4.1.232.6.2.6.8.1.4.1.3&1.3.6.1.4.1.232.6.2.6.8.1.4.1.3:public@prolaint.
exampledomain.com
MaxBytes[localhost.moretemp]: 200
Options[localhost.moretemp]: gauge
Title[localhost.moretemp]: proliant.exampledomain.com: 2 More Temperature
Sensors
YLegend[localhost.moretemp]: Moretemperature (C)
LegendI[localhost.moretemp]:  CPUSensorTwo:
LegendO[localhost.moretemp]:  CPUSensorThree:
PageTop[localhost.moretemp]: <H1>More Temperature Sensors for
proliant.exampledomain.com</H1>
   <TABLE>
      <TR><TD>System:</TD>       <TD>proliant.exampledomain.com in lab, rack
3</TD></TR>
      <TR><TD>Maintainer:</TD> <TD>someuser@exampledomain.com</TD></TR>
      <TR><TD>Ip:</TD>           <TD>xx.xx.xx.xx</TD></TR>
  </TABLE>
```

The MRTG package also includes a tool, **indexmaker**, which will generate an HTML file suitable for a top-level page if you are monitoring a number of systems.

In the above example,

```
# indexmaker --columns=1 --show=day cpqinfo.cfg
```

Getting updates every five minutes

The next step is to arrange for MRTG to be run every five minutes to gather the temperature information and generate new graphs. This is done by adding an entry into the **crontab**. The command **crontab -e** will start an editor and allow you to edit the existing entries. Add an entry that looks like this:

```
*/5 * * * *  mrtg /home/someuser/scripts/cpqinfo.cfg
```

Example of MRTG generated Web pages

A top level page, illustrated in Figure 1, is generated one time by indexmaker while the specific system pages are updated by MRTG. Note that there are two graphs, each one displaying a graph about two of the temperature sensors.
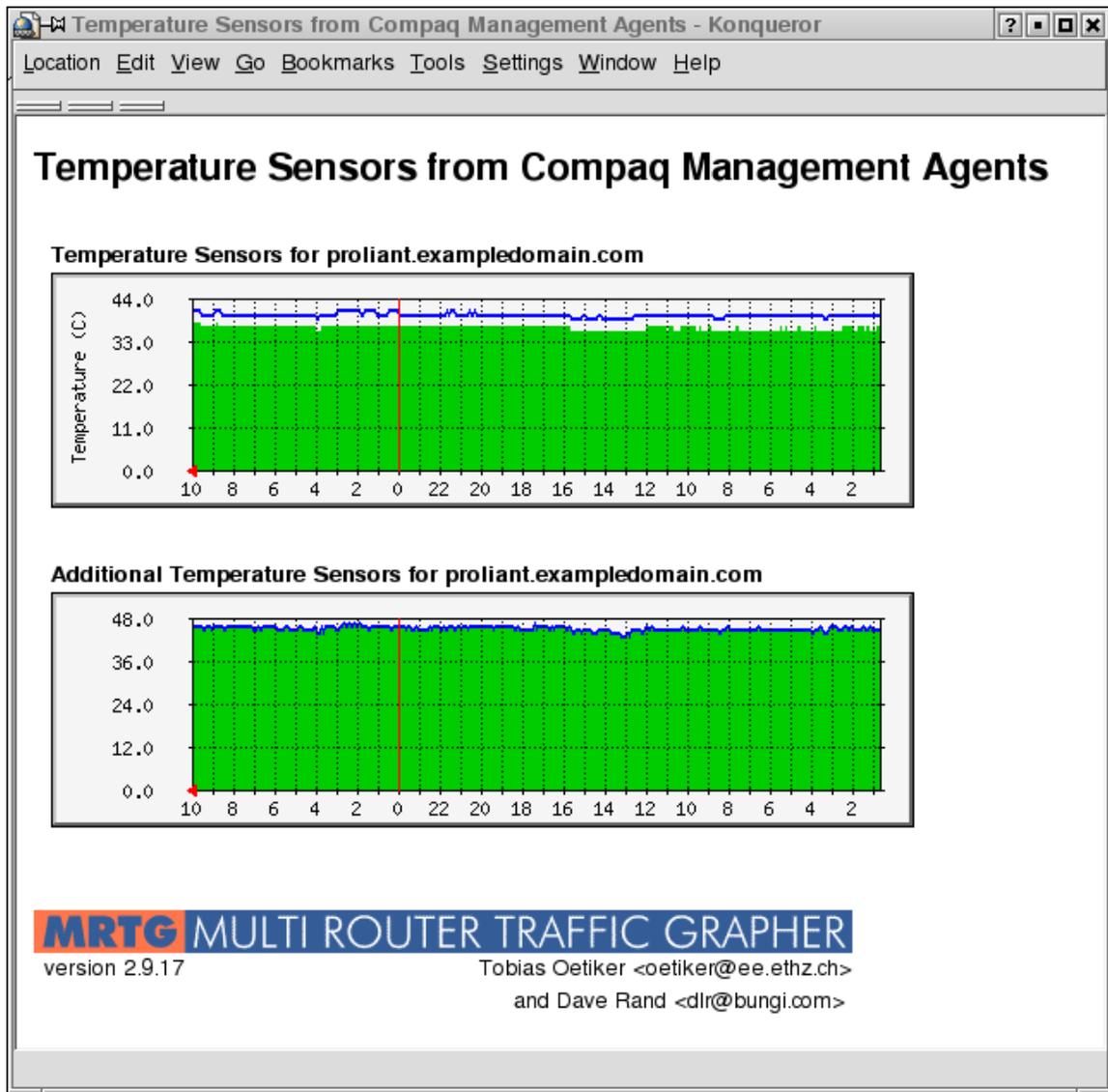
Figure 1. MRTG Temperature Top Page

If you click on the graphs, it takes you to a page that shows graphs with the information broken down by hourly, daily, weekly and yearly graphs, as shown in Figure 2.
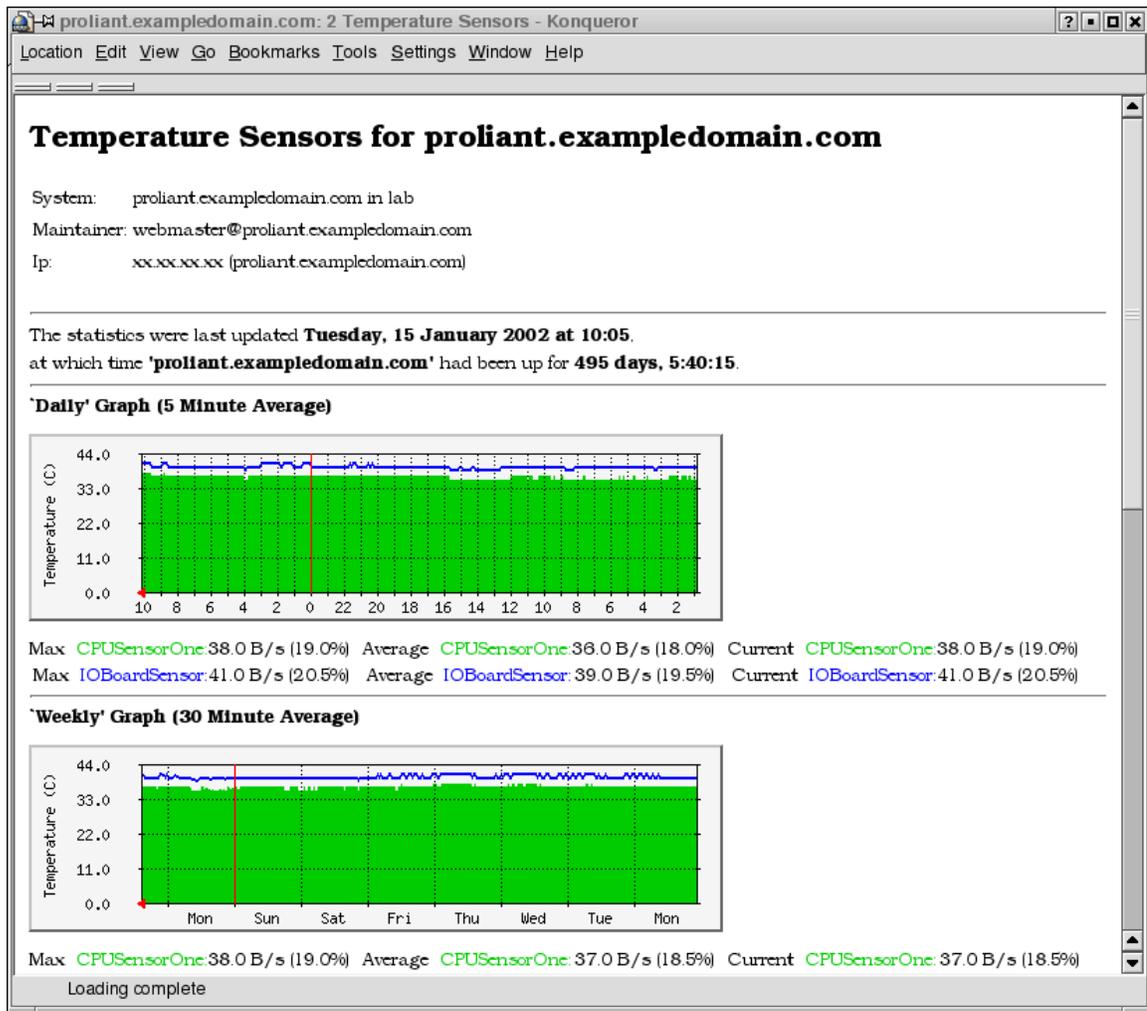
Temperature Sensors for proliant.exampledomain.com

System: proliant.exampledomain.com in lab

Maintainer: webmaster@proliant.exampledomain.com

Ip: xx.xx.xx.xx (proliant.exampledomain.com)

The statistics were last updated **Tuesday, 15 January 2002 at 10:05**,
at which time **'proliant.exampledomain.com'** had been up for **495 days, 5:40:15**.

`Daily' Graph (5 Minute Average)

Max CPUSensorOne:38.0 B/s (19.0%) Average CPUSensorOne:36.0 B/s (18.0%) Current CPUSensorOne:38.0 B/s (19.0%)
Max IOBoardSensor:41.0 B/s (20.5%) Average IOBoardSensor:39.0 B/s (19.5%) Current IOBoardSensor:41.0 B/s (20.5%)

`Weekly' Graph (30 Minute Average)

Max CPUSensorOne:38.0 B/s (19.0%) Average CPUSensorOne:37.0 B/s (18.5%) Current CPUSensorOne:37.0 B/s (18.5%)

Figure 2. MRTG Detail Graph page

Using RRDTOOL to Monitor and Graph CMA-specific Information

RRDTOOL is another popular open source package that stores and displays time-series data in a compact way that does not grow over time. Here, the reader is shown how to monitor CPU and memory usage as well as the temperature sensors that are on the motherboard of Compaq *ProLiant* servers.

If the RRDTOOL package is not already installed on your system, visit http://www.rrdtool.org/ and download and install the latest kit.

Monitoring Temperature with RRDTOOL

The first step is to create an RRDTOOL database in which to store the information. We want to keep enough information so we can generate graphs with a granularity of five minutes and to show up to a year's worth of data.

We take temperature samples every five minutes. This information is based on the example from the RRDTOOL tutorial.

Note: This set of example scripts assumes that the system being monitored has two CPUs and four temperature sensors on the motherboard. The scripts will need to be modified if your system does not match this configuration.

```
#!/bin/bash
# 600 samples of 5 (1 * 5) minutes        (2 days & 2 hours)
# 700 samples of 30 (6 * 5) minutes       (2 days & 2 hours, plus 12.5 days)
# 775 samples of 2 (24 * 5) hours         ( " plus 50 days)
# 797 samples of 1 (288 * 5) day          ( " plus 732 days, rounded up to 797)
/usr/local/rrdtool-1.0.33/bin/rrdtool create cpqinfo.rrd            \
        DS:sensor:GAUGE:600:U:U          \
      DS:sensor2:GAUGE:600:U:U           \
      DS:sensor3:GAUGE:600:U:U           \
      DS:sensor4:GAUGE:600:U:U           \
      DS:memorysize:GAUGE:600:U:U        \
      DS:memoryfree:GAUGE:600:U:U        \
      DS:cpu0:GAUGE:600:U:U              \
      DS:cpu1:GAUGE:600:U:U              \
      RRA:AVERAGE:0.5:1:600              \
      RRA:AVERAGE:0.5:6:700              \
      RRA:AVERAGE:0.5:24:775             \
      RRA:AVERAGE:0.5:288:797            \
      RRA:MAX:0.5:1:600                  \
      RRA:MAX:0.5:6:700                  \
      RRA:MAX:0.5:24:775                 \
      RRA:MAX:0.5:288:797
```

This command will create a file named **cpqinfo.rrd**, which will hold all the samples for two CPUs, memory use as well as four temperature sensors.

Collecting Data Samples

There are a number of ways to gather the information via SNMP. The simplest approach is to write a small program in Perl that uses the **Net::SNMP** module to get the data from the *Compaq Management Agents for Linux*. Depending on your Perl configuration, you may need to download **Net::SNMP, Crypt::DES, Digest::MD5, Digest::SHA1**, and **Digest::HMAC** from http://www.cpan.org/modules/by-module/ and install them.

Below is **cpqinfo-get.pl**, a sample Perl script that uses SNMP to retrieve *Compaq Management Agent* information from the local system. Depending on your configuration, the values for **$HOSTNAME**, **$COMMUNITY**, **$DATABASE** and **$RRDTOOL** may need to be changed.

```
#!/usr/bin/perl

# Get memory and CPU use information using SNMP and store it in an
# RRDTOOL database.
# The name of the OIDs we are interested in are:
#
# Temperature Sensors:

#   enterprises.compaq.cpqHealth.cpqHeComponent\
#       .cpqHeThermal.cpqHeTemperatureTable.cpqHeTemperatureEntry\
#           .cpqHeTemperatureCelsius.1.1
#   enterprises.compaq.cpqHealth.cpqHeComponent\
#       .cpqHeThermal.cpqHeTemperatureTable.cpqHeTemperatureEntry\
#           .cpqHeTemperatureCelsius.1.2
#   enterprises.compaq.cpqHealth.cpqHeComponent\
#       .cpqHeThermal.cpqHeTemperatureTable.cpqHeTemperatureEntry\
#           .cpqHeTemperatureCelsius.1.3
#   enterprises.compaq.cpqHealth.cpqHeComponent\
#       .cpqHeThermal.cpqHeTemperatureTable.cpqHeTemperatureEntry\
#           .cpqHeTemperatureCelsius.1.4
#
# Memory Information:
#
#   enterprises.compaq.cpqHostOs.cpqHoComponent\
#       .cpqHoMemory.cpqHoPhysicalMemorySize.0
#   enterprises.compaq.cpqHostOs.cpqHoComponent\
#       .cpqHoMemory.cpqHoPhysicalMemoryFree.0
#
# CPU use in the last minute on a 2 CPU system
#
#   enterprises.compaq.cpqHostOs.cpqHoComponent\
#       .cpqHoUtil.cpqHoCpuUtilTable.cpqHoCpuUtilEntry.cpqHoCpuUtilMin.0
#   enterprises.compaq.cpqHostOs.cpqHoComponent\
#       .cpqHoUtil.cpqHoCpuUtilTable.cpqHoCpuUtilEntry.cpqHoCpuUtilMin.1
#
# Those correspond to the numeric representation:
#
# Temperature Sensors:
#
#       .1.3.6.1.4.1.232.6.2.6.8.1.4.1.1
#       .1.3.6.1.4.1.232.6.2.6.8.1.4.1.2
#       .1.3.6.1.4.1.232.6.2.6.8.1.4.1.3
#       .1.3.6.1.4.1.232.6.2.6.8.1.4.1.4
#
# Memory Information:
#
#       .1.3.6.1.4.1.232.11.2.13.1.0
#       .1.3.6.1.4.1.232.11.2.13.2.0
```

```
#
# CPU use in the last minute on a 2 CPU system
#
#       .1.3.6.1.4.1.232.11.2.3.1.1.2.0
#       .1.3.6.1.4.1.232.11.2.3.1.1.2.1
#


use Net::SNMP;


$HOSTNAME =  "localhost";
$COMMUNITY = "public";
$DATABASE =  "/home/someuser/public_html/cpqinfo.rrd";
$RRDTOOL =   "/usr/local/rrdtool-1.0.33/bin/rrdtool";


my ($session, $error) = Net::SNMP->session( -hostname  => $HOSTNAME,
                                            -community => $COMMUNITY,
                                            -port      => 161);
if (!defined($session)) {
    printf("ERROR: %s.\n", $error);
    exit 1;
}


my $sensor1oid =      '1.3.6.1.4.1.232.6.2.6.8.1.4.1.1';
my $sensor2oid =      '1.3.6.1.4.1.232.6.2.6.8.1.4.1.2';
my $sensor3oid =      '1.3.6.1.4.1.232.6.2.6.8.1.4.1.3';
my $sensor4oid =      '1.3.6.1.4.1.232.6.2.6.8.1.4.1.4';
my $memorysizeoid =   '1.3.6.1.4.1.232.11.2.13.1.0';
my $memoryfreeoid =   '1.3.6.1.4.1.232.11.2.13.2.0';
my $cpu0oid =           '1.3.6.1.4.1.232.11.2.3.1.1.2.0';
my $cpu1oid =         '1.3.6.1.4.1.232.11.2.3.1.1.2.1';


my $result = $session->get_request(
                        -varbindlist => [$sensor1oid,
                                         $sensor2oid,
                                         $sensor3oid,
                                         $sensor4oid ,
                                         $memorysizeoid,
                                         $memoryfreeoid,
                                         $cpu0oid,
                                         $cpu1oid]
                        );
if (!defined($result)) {
    printf("ERROR: %s.\n", $session->error);
    $session->close;
    exit 1;
}


$session->close;


# Now fork a process to run rrdtool to insert the information into the
# database.
#
# NOTE: The command is very long and should be on one line.  Printed
#       documentation may wrap the line.
#


`$RRDTOOL update $DATABASE N:$result->{$sensor1oid}:$result-
>{$sensor2oid}:$result->{$sensor3oid}:$result->{$sensor4oid}:$result-
>{$memorysizeoid}:$result->{$memoryfreeoid}:$result->{$cpu0oid}:$result-
>{$cpu1oid}\n`;


exit 0;
```

Getting Updates Every Five Minutes

Use the command **crontab -e** to create a job that runs **cpqinfo-get.pl** every five minutes. Add an entry that looks like this:

```
*/5 * * * *  /home/someuser/scripts/cpqinfo-get.pl
```

Generating Thermal Reports

Below are examples of commands that generate various graphs of the collected data. Below each section is the resulting PNG image. The first is illustrated as Figure 3.

```
#!/bin/sh
rrdtool graph average-1hour.png \
        --imgformat PNG\
        --start -1hour\
        --title "Temperature (Average)"\
        --vertical-label "Celsius"\
        DEF:s1=cpqinfo.rrd:sensor:AVERAGE \
        DEF:s2=cpqinfo.rrd:sensor2:AVERAGE \
        DEF:s3=cpqinfo.rrd:sensor3:AVERAGE \
        DEF:s4=cpqinfo.rrd:sensor4:AVERAGE \
        LINE1:s1#0000ff:"CPU Sensor 1"\
        LINE2:s2#ff0000:"IO Board Sensor"\
        LINE3:s3#00ff00:"CPU Sensor 2"\
        LINE1:s4#fffff:"CPU Sensor 3"
```
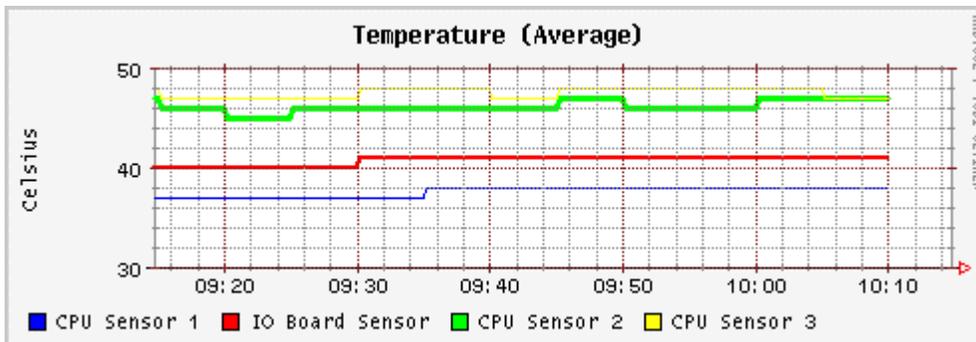

Figure 3. RRDTOOL Graph: Average Temperature (Celsius)

Using standard RRDTOOL techniques it is possible to convert the temperature values from Celsius for Fahrenheit, with results as illustrated in Figure 4.

```
#!/bin/sh
rrdtool graph average-1hour-f.png \
        --imgformat PNG\
        --start -1hour\
        --title "Temperature (Average)"\
        --vertical-label "Fahrenheit"\
        DEF:s1=cpqinfo.rrd:sensor:AVERAGE \
        DEF:s2=cpqinfo.rrd:sensor2:AVERAGE \
        DEF:s3=cpqinfo.rrd:sensor3:AVERAGE \
        DEF:s4=cpqinfo.rrd:sensor4:AVERAGE \
```

```
CDEF:fs1=9,5,/,s1,*,32,+\
CDEF:fs2=9,5,/,s2,*,32,+\
CDEF:fs3=9,5,/,s3,*,32,+\
CDEF:fs4=9,5,/,s4,*,32,+\
LINE1:fs1#0000ff:"CPU Sensor 1"\
LINE2:fs2#ff0000:"IO Board Sensor"\
LINE3:fs3#00ff00:"CPU Sensor 2"\
LINE1:fs4#ffffff:"CPU Sensor 3"
```
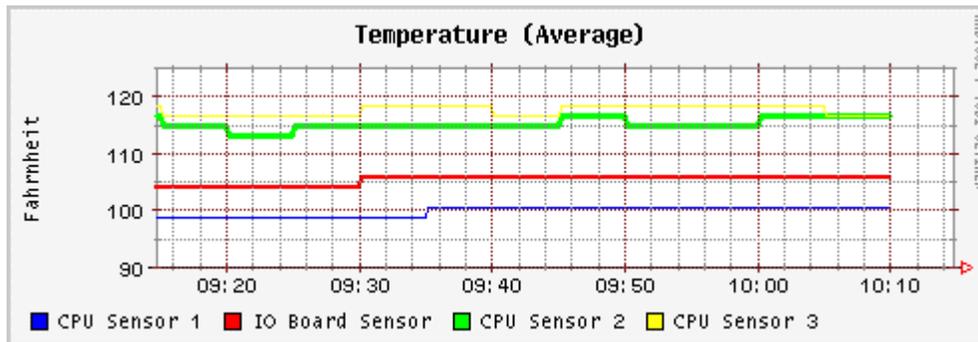

Figure 4. RRDTOOL graph: Average Temperature (Fahrenheit)

To alter the granularity, as illustrated in Figure 5, use the following script.

```
#!/bin/bash
rrdtool graph average-1day.png \
        --imgformat PNG\
        --start -1day\
        --title "Temperature (Average)"\
        --vertical-label "Celsius"\
        DEF:s1=cpqinfo.rrd:sensor:AVERAGE \
        DEF:s2=cpqinfo.rrd:sensor2:AVERAGE \
        DEF:s3=cpqinfo.rrd:sensor3:AVERAGE \
        DEF:s4=cpqinfo.rrd:sensor4:AVERAGE \
        LINE1:s1#0000ff:"CPU Sensor 1"\
        LINE2:s2#ff0000:"IO Board Sensor"\
        LINE3:s3#00ff00:"CPU Sensor 2"\
        LINE1:s4#ffffff:"CPU Sensor 3"
```
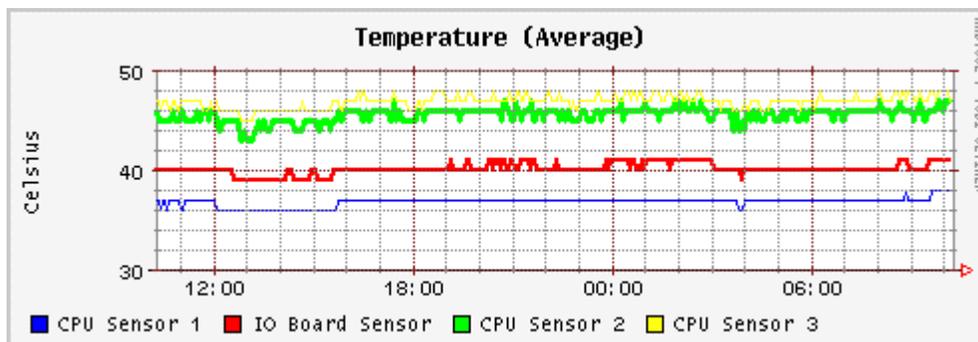

Figure 5. Average Temperature – different granularity

Generating Memory and CPU Reports

As with graphing the temperature in the previous section, here memory, as shown in Figure 6, and CPU use can be graphed.

```
#!/bin/bash
rrdtool graph average-memory-1day.png \
      --imgformat PNG\
      --start -1day\
      --title "Memory Use"\
      --vertical-label "megabyte"\
      DEF:memorysize=cpqinfo.rrd:memorysize:AVERAGE \
      DEF:memoryfree=cpqinfo.rrd:memoryfree:AVERAGE \
      LINE1:memorysize#0000ff:"Max Memory"\
      LINE2:memoryfree#ff0000:"FreeMemory"
```
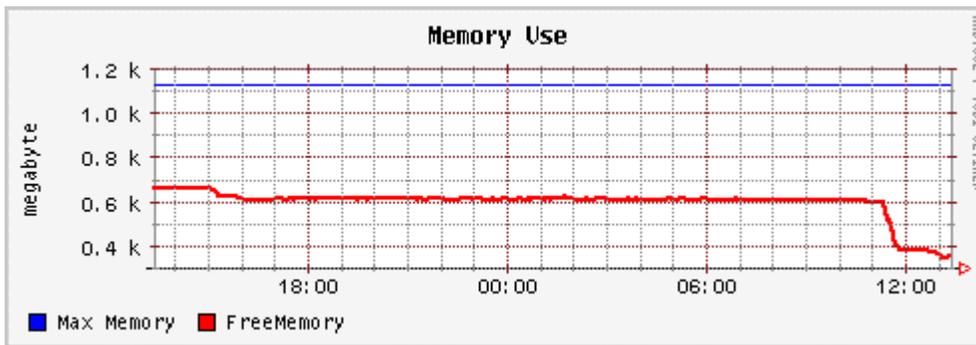

Figure 6. Memory Use

In the following example the usage of a two-CPU system is graphed as shown in Figure 7.

```
#!/bin/bash
rrdtool graph average-cpu-1day.png \
      --imgformat PNG\
      --start -1day\
      --title "CPU Use (Average)"\
      --vertical-label "Percent"\
      DEF:cpu0=cpqinfo.rrd:cpu0:AVERAGE \
      DEF:cpu1=cpqinfo.rrd:cpu1:AVERAGE \
      LINE1:cpu0#0000ff:"CPU 0"\
      LINE2:cpu1#ff0000:"CPU 1"
```
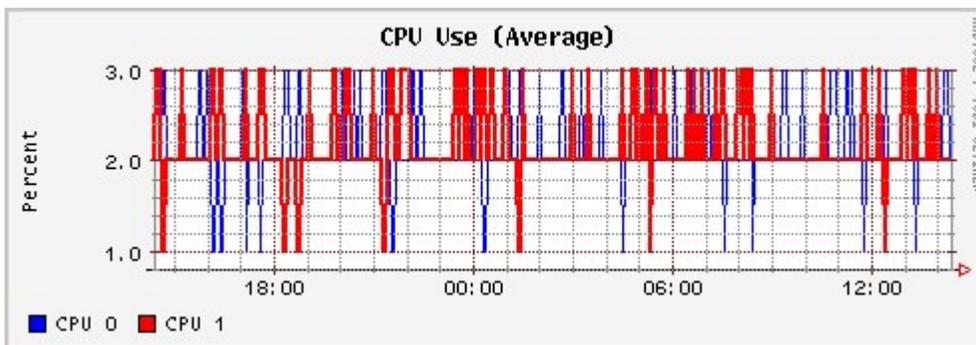

Figure 7. CPU Use

Further Reading

For information on the software packages discussed in this guide, see the following websites:

- Compaq Management website – http://www.compaq.com/manage

- Compaq Software and Driver website --
  http://www.compaq.com/support/files/server/us/index.html

- Net-SNMP (formerly UCD-SNMP) – http://net-snmp.sourceforge.net

- Multi Router Traffic Grapher (MRTG) – http://www.mrtg.org

- RRDTOOL – http://www.rrdtool.org

- Comprehensive Perl Archive Network – http://www.cpan.org

  o Net::SNMP kit – http://www.cpan.org/modules/by-module/Net/

  o Crypt::DES kit – http://www.cpan.org/modules/by-module/Crypt/

  o Digest::MD5 kit – http://www.cpan.org/modules/by-module/Digest/

  o Digest::SHA1 kit – http://www.cpan.org/modules/by-module/Digest/

  o Digest::HMAC kit – http://www.cpan.org/modules/by-module/Digest/