

Power Management: How Do I monitor APM In An HP Apollo 8000?

APM Monitoring Setup for An HP Apollo 8000 rack in Bright 7.x

Static Routes:

The head node can access the APMs through an intermediate host which has a direct connection to the APMs. The static routes can be set up as follows:

1. on both the nodes, add a static route to the 10.243.56.0/21 subnet via the intermediate host:

```
adm01:~ # ip route add 10.243.56.0/21 via 10.243.71.253 dev eth4
```

```
adm01:~ # ip route
```

```
default via 14.0.0.44 dev eth5
```

```
10.243.56.0/21 via 10.243.71.253 dev eth4
```

```
10.243.64.0/21 dev eth4 proto kernel scope link src 10.243.71.251
```

```
10.243.72.0/21 dev ib0 proto kernel scope link src 10.243.72.4
```

```
14.0.0.0/24 dev eth5 proto kernel scope link src 14.0.0.101
```

```
127.0.0.0/8 dev lo scope link
```

```
adm01:~ #
```

2. Add a static route on the intermediate host

```
[root@sysmgr ~]# ip route add 10.243.56.0/21 dev eno2 proto kernel scope link src 10.243.63.253
```

```
[root@sysmgr ~]# ip route
```

```
10.243.56.0/21 dev eno2 proto kernel scope link src 10.243.63.253
```

```
10.243.64.0/21 dev eno1 proto kernel scope link src 10.243.71.253
```

```
169.254.0.0/16 dev eno1 scope link metric 1002
```

```
169.254.0.0/16 dev eno2 scope link metric 1003
```

Power Management: How Do I monitor APM In An HP Apollo 8000?

3. Enable IP forwarding on the intermediate host:

```
[root@sismgr ~]# echo 1 > /proc/sys/net/ipv4/ip_forward
```

4. Set up the gateway on the APMs to point to the intermediate host so that it can respond back to the head node:

```
-apm> set gateway 10.243.63.253
```

```
IP default gateway set to 10.243.63.253
```

Monitoring Host:

- If the head node is chosen to be the monitoring host for the APMs, then all the health checks and metric collection scripts will run on the head nodes. The downside of this approach is that both head nodes will be collecting the same metrics and running the same health checks.
- The other approach is to add the APMs as racksensor objects. This way, each APM will be monitored separately.

Monitoring protocol:

The APMs can be accessed via SSH and SNMP, or through the system manager using an XML format. The available MIBs for SNMP don't provide valuable information, so grabbing any useful information through SNMP will not be possible. For more information about the supported MIBs, please refer to "Supported MIB objects" in HP Advanced Power Management User Guide: "<http://h10032.www1.hp.com/ctg/Manual/c04338737>"

An expect script was developed for use from within the health checks and metric collection scripts, in order to access the APM through SSH.

Health checks:

The health checks have been implemented as generic scripts in python. They accept the IP address of the APM as a mandatory argument, and run the health check against this IP address.

Available health checks:

Power Management: How Do I monitor APM In An HP Apollo 8000?

1. **Rack Zone:** each rack has two zones, upper and lower, with separate thermal sensors and fans. If a sensor is not present or is in an error state, then the health check will fail, and it will show which zone is faulty along with any associated error message.
2. **Leak Detector:** responsible for detecting any water leakage in the rack, it checks the leak detectors in upper and lower zones of a particular rack. A “no leak” response results in a PASS and is assumed to be the optimal response, otherwise the check fails and an “Attention Needed” message is shown.
3. **Power Supply:** responsible for checking the available power supplies. A “present - OK” response results in a PASS and is assumed to be the optimal response, otherwise the check fails and an “Attention Needed” message is shown.

Setup Health Checks:

1. Download the exepct.sh script which will be used to connect to the APM:

```
# mkdir -p /cm/local/apps/cmd/scripts/healthchecks/apm
```

```
# cd /cm/local/apps/cmd/scripts/healthchecks/apm
```

```
# wget -c http://support.brightcomputing.com/GL2eqAc5LI/expect.sh
```

2. Download the health checks from the following link:

```
# cd /cm/local/apps/cmd/scripts/healthchecks/apm
```

```
# wget -c http://support.brightcomputing.com/GL2eqAc5LI/hp\_apollo\_healthchecks.tar.gz
```

3. Unpack the tar.gz file under /cm/local/apps/cmd/scripts/healthchecks/apm

```
# cd /cm/local/apps/cmd/scripts/healthchecks/apm
```

```
# tar -xzvf hp_apollo_healthchecks.tar.gz
```

```
# chmod 755 *
```

Power Management: How Do I monitor APM In An HP Apollo 8000?

4. Add the APM healthchecks to the monitoring healthchecks:

```
# cmsh
% monitoring healthchecks
% add apmleakdetector_sensor
% set timeout 20
% set validfor racksensor
% set samplingmethod samplingonheadnode
% set command /cm/local/apps/cmd/scripts/healthchecks/leakdetector_racksensor.py # (in case racksensors will be used as this script will use the IP address of the racksensor)
% set command /cm/local/apps/cmd/scripts/healthchecks/leakdetector.py # (in case head node will be used as this script will use the IP address passed as a parameter)
% set parameterpermissions required # (in case head node will be used)
% commit
```

5. Assign the health check for the head node or the racksensor

```
% monitoring setup healthconf racksensor
% add apmleakdetector_sensor
% monitoring setup healthconf headnode
% add apmleakdetector_sensor
% set healthcheckparam <IP address of the APM> # (in case head node will be used)
% commit
```

6. The previous steps shall be repeated for “powersupply_racksensor.py”, and “rackzone_racksensor.py” (for racksensor) or “powersupply.py” and “rackzone.py” (for head node) scripts.

Power Management: How Do I monitor APM In An HP Apollo 8000?

Metric Collection:

The metrics are collected through metric collection scripts implemented in python.

If the head node is used for monitoring, it will not be possible to make the script generic to run against a given IP address. This is because CMDaemon will run the metric collection prototype with the "--initialize" argument only and will not send any other parameter.

Instead, a rack.conf file will be needed, which contains a rackname-to-IP address mapping. The metric collection script will use the rack configuration file to add the detected metrics during the initialization phase and during the collection phase. In case the APMs are added as rack sensors, the CMD_IP environment variable will be used.

Available metric collection:

1. **Fan:** responsible for monitoring Fan speed in RPMs for the available fans in all configured racks and in all zones within each rack.
2. **Temperature:** responsible for monitoring different temperatures in Celsius for the available thermal sensors in all configured racks and in all zones within each rack. The Flow Rate is also collected by this metric collection script as it is associated with the water temperature, and it also avoids creating a new metric script which would require initiating another ssh connection to the APM.

Setup Metric Collection:

1. Download the expect.sh script:

```
# mkdir -p /cm/local/apps/cmd/scripts/metrics/apm
```

```
# cd /cm/local/apps/cmd/scripts/metrics/apm
```

```
# wget -c http://support.brightcomputing.com/GL2eqAc5LI/expect.sh
```

2. Download the metrics from the following link:

```
# wget -c http://support.brightcomputing.com/GL2eqAc5LI/hp\_apollo\_metrics.tar.gz
```

Power Management: How Do I monitor APM In An HP Apollo 8000?

3. Unpack the tar.gz file under /cm/local/apps/cmd/scripts/metrics/apm

```
# mkdir -p /cm/local/apps/cmd/scripts/metrics/apm
```

```
# cd /cm/local/apps/cmd/scripts/metrics/apm
```

```
# tar -xzf /path/to/hp_apollo_metrics.tar.gz
```

```
# chmod 755 *
```

4. Add the APM metrics to the monitoring metrics:

```
# csh
```

```
% monitoring metrics
```

```
% add apmfan_sensor
```

```
% set timeout 20
```

```
% set validfor racksensor # (in case rack sensor will be used)
```

```
% set validfor headnode # (in case head node will be used)
```

```
% set samplingmethod samplingonheadnode
```

```
% set command /cm/local/apps/cmd/scripts/metrics/apm/fan_racksensor.py # (in case  
racksensor will be used as this script will use the IP address of the racksensor)
```

```
% set command /cm/local/apps/cmd/scripts/metrics/apm/fan.py # (in case head node will be  
used as this script will get the IP address from rack.conf file)
```

```
% commit
```

5. Assign the health check for the head node or the racksensor

```
% monitoring setup metricconf racksensor
```

```
% add apmfan_sensor
```

```
% monitoring setup metricconf headnode
```

Power Management: How Do I monitor APM In An HP Apollo 8000?

```
% add apmfan_sensor
```

```
% commit
```

6. The previous steps should be repeated for “temp_racksensor.py” (for racksensor) or “temp.py” (for head node) scripts.

APM racksensors:

The preferred and recommended way to monitor the APMs is to add them as rack sensors as this will prevent running into the situation where both head nodes will be collecting the same metrics and running the same health checks. Also, using rack sensors is relatively faster as each rack sensor will access its own APM which may take up to 6 seconds, while in case of using the head nodes as the monitoring host, each head node has to access all the APMs to run a single health check, which may take up to 30 seconds in a non-busy environment.

Cooling Distribution Unit (iCDU)

- These are special APMs, namely APM2 and APM5 with IP addresses 10.243.63.5 and 10.243.63.8 respectively. The iCDUs are special in the sense that they don't have a lower rack zone and there is no water flowing through them. If the racksensor is an iCDU APM, then the userdefined1 value for the racksensor object should be set to icdu so that the health check and metric collection scripts will ignore the lower rack zone, the flow rate and the intermediate water temperature.
- At the moment, the head node cannot differentiate between regular APMs and iCDU APMs. So if the head node is used as the monitoring host, then there will be some false alarms that can be ignored for the racks that are known to be iCDU. A way to implement this is to introduce a modification to the rack configuration file where a flag can be used to distinguish between regular APMs and iCDU APMs.

Adding racksensors through CMSH:

```
# cmsH
```

```
% device add racksensor apm1
```

```
% set ip 10.243.63.4
```

```
% set network internalnet
```

```
% set userdefined1 icdu # (in case this is an iCUD APM)
```

Power Management: How Do I monitor APM In An HP Apollo 8000?

% commit

Notes:

1. Configuring the APMs as rack sensors will require a custom ping script as the default telnet port for the APMs is not '2'. So CMDaemon will not be able to detect the APMs as UP without a custom ping script.
2. The health checks will go into the UNKNOWN state if the APM is not reachable or in accessible. The output of the ssh command will be displayed as the error message.
3. The metrics collection scripts will not fail if the APM is not reachable or accessible. Instead, all the collected metrics will have no data.

Unique solution ID: #1301

Author: mohamed adel

Last update: 2016-03-14 13:43