Monitoring and Alert Notification

Architectural Overview

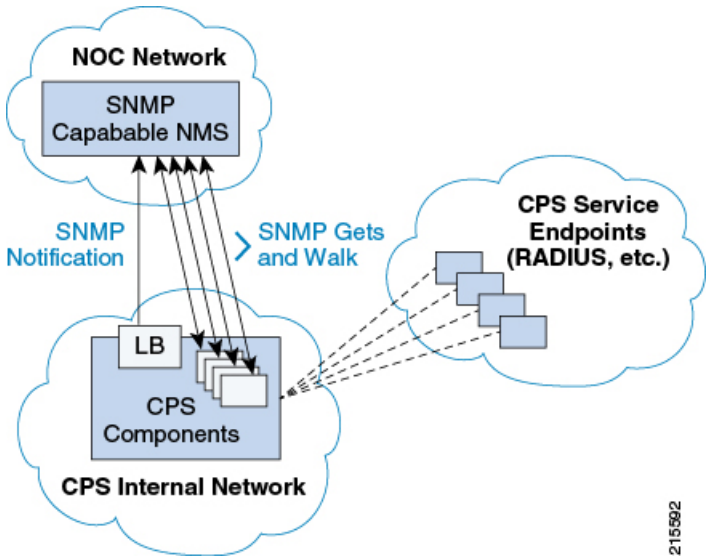
A Cisco Policy Suite (CPS) deployment comprises multiple virtual machines (VMs) deployed for scaling and High Availability (HA) purposes. All VMs present in the system should have an IP address which is a routable IP to the Network Management System (NMS). The NMS can monitor each VM using this routable IP address.

Note The IP addresses do not need to be routable if the NMS has an interface on the same internal network as the CPS VMs.

During runtime any number of VMs can be added to the system and the NMS can monitor them using their routable IP address which makes the system more scalable. The notification alerting from the entire system derives from a single point.

When CPS is deployed in a High Availability (HA) alerting endpoints are deployed as HA as well as shown in the following illustration.

Figure 1: HA Deployment



Technical Architecture

Cisco Policy Suite is deployed as a distributed virtual appliance. The standard architecture uses hypervisor virtualization. Multiple physical hardware host components run Hypervisors and each host runs several virtual machines. Within each virtual machine one-to-many internal CPS components can run. CPS monitoring and alert notification infrastructure simplifies the virtual physical and redundant aspects of the architecture.

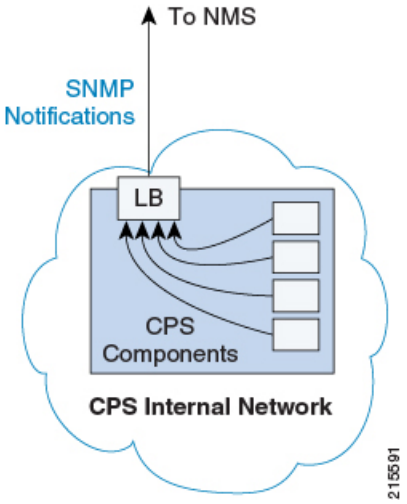
Protocols and Query Endpoints

The CPS monitoring and alert notification infrastructure provides a simple standards-based interface for network administrators and NMS (Network Management System). SNMP is the underlying protocol for all monitoring and alert notifications. Standard SNMP gets and notifications (traps) are used throughout the infrastructure.

At any point of time only one version of SNMP (either SNMPv2 or SNMPv3) will work. By default SNMPv3 is disabled. For information on configuring SNMPv3 refer to the *CPS Installation Guide for VMware* or to the *CPS Installation Guide for OpenStack* for this release.

The following illustration shows the aggregation and mapping on the SNMP endpoint (Policy Director (LB)).

Figure 2: SNMP Endpoint



SNMP Object Identifier and Management Information Base

Cisco has a registered private enterprise Object Identifier (OID) of 26878. This OID is the base from which all aggregated CPS metrics are exposed at the SNMP endpoint. The Cisco OID is fully specified and made human-readable through a set of Cisco Management Information Base (MIB-II) files.

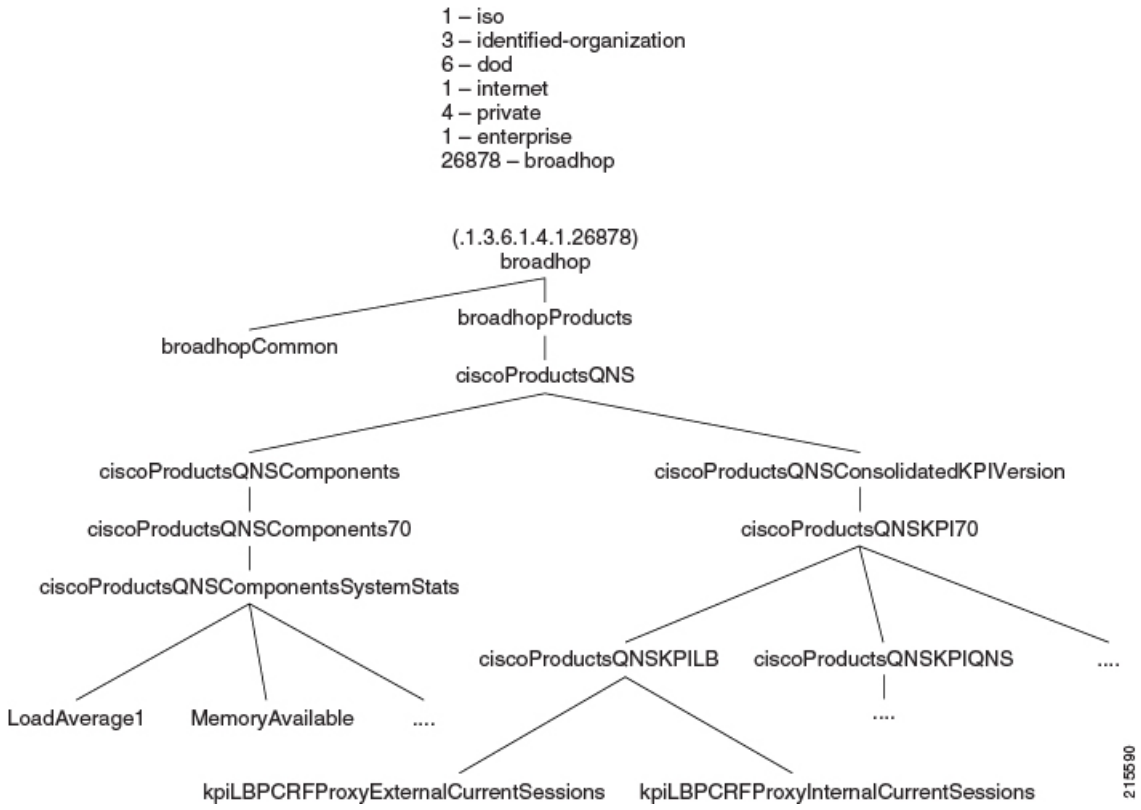
The current MIBs are defined as follows:

Table 1: MIBs

MIB Filename	Purpose
BROADHOP-MIB.mib	Defines the main structure including structures and codes.
CISCO-QNS-MIB.mib	Defines the retrievable statistics and KPI.
BROADHOP-NOTIFICATION-MIB.mib	Defines Notifications/Traps available.

A graphical overview of the CPS OID and MIB structure is shown in the next figure.

Figure 3: SNMP Notifications



Note that in the above illustration the entire tree is not shown.

SNMP Data and Notifications

The Monitoring and Alert Notification infrastructure provides standard SNMP get and getnext access to the CPS system. This provides access to targeted metrics to trend and view Key Performance Indicators (KPIs). Metrics available through this part of the infrastructure are as general as component load and as specific as transactions processed per second.

SNMP Notifications in the form of traps (one-way) are also provided by the infrastructure. CPS notifications do not require acknowledgments. These provide both proactive alerts that predetermined thresholds have been passed (for example a disk is nearing capacity or CPU load is too high) and reactive alerting when system components fail or are in a degraded state (for example a process died or network connectivity outage has occurred).

Notifications and traps are categorized by a methodology similar to UNIX System Logging (syslog) with both Severity and Facility markers. All event notifications (traps) contain these items

- Facility
- Severity
- Source (device name)
- Device time

These objects enable Network Operations Center (NOC) staff to identify where the issue lies the Facility (system layer) and the Severity (importance) of the reported issue.

Note For more information on CPS statistics, refer to CPS Statistics chapter in *CPS Operations Guide* for this release. For more information on CPS logging, refer to Logging chapter in *CPS Troubleshooting Guide* for this release.

Facility

The generic syslog facility has the following definitions.

Note Facility defines a system layer starting with physical hardware and progressing to a process running in a particular application.

Table 2: Syslog Facility

Number	Facility	Description
0	Hardware	Physical Hardware – Servers SAN NIC Switch and so on.
1	Networking	Connectivity in the OSI (TCP/IP) model.
2	Virtualization	VMware ESXi (or other) Virtualization
3	Operating System	Linux Microsoft Windows and so on.
4	Application	Apache httpd load balancer CPS Cisco sessionmgr and so on.
5	Process	Particular httpd process CPS qns01_A and so on.

There may be overlaps in the Facility value as well as gaps if a particular SNMP agent does not have full view into an issue. The Facility reported is always shown as viewed from the reporting SNMP agent.

Severity

In addition to Facility each notification has a Severity measure. The defined severities are directly from UNIX syslog and defined as follows:

Table 3: Severity Levels

Number	Severity	Description
0	Emergency	System is unusable.
1	Alert	Action must be taken immediately.
2	Critical	Critical conditions.
3	Error	Error conditions.

Number	Severity	Description
4	Warning	Warning conditions.
5	Notice	Normal but significant condition.
6	Info	Informational message.
7	Debug	Lower level debug messages.
8	None	Indicates no severity.
9	Clear	The occurred condition has been cleared.

For the purposes of the CPS Monitoring and Alert Notifications system, Severity levels of Notice Info and Debug are usually not used.

Warning conditions are often used for proactive threshold monitoring (for example Disk usage or CPU Load) which requires some action on the part of administrators but not immediately.

Conversely, Emergency severity indicates that some major component of the system has failed and that either core policy processing session management or major system functionality is impacted.

Categorization

Combinations of Facility and Severity create many possibilities of notifications (traps) that might be sent. However some combinations are more likely than others. The following table lists some Facility and Severity categorizations.

Table 4: Severity Categorization

Facility.Severity	Categorization	Possibility
Process.Emergency	A single part of an application has dramatically failed.	Possible but in an HA configuration very unlikely.
Hardware.Debug	A hardware component has sent a debug message.	NA
Operating System.Alert	An Operating System (kernel or resource level) fault has occurred.	Possible as a recoverable kernel fault (on a vNIC for instance).
Application.Emergency	An entire application component has failed.	Unlikely but possible (load balancers failing for instance).

It is not possible to quantify every Facility and Severity combination. However greater experience with CPS leads to better diagnostics. The CPS Monitoring and Alert Notification infrastructure provides a baseline for event definition and notification by an experienced engineer.

Emergency Severity Note

Caution Emergency severities are very important! As a general principle CPS does not throw an Emergency-severity trap unless the system becomes inaccessible or unusable in some way. An unusable

system is rare but might occur if multiple failures occur in the operating system virtualization networking or hardware facilities.

SNMP System and Application KPIs

Many CPS system statistics and Key Performance Indicators (KPI) are available via SNMP gets and walks. Both system device level information and application level information is available. This information is documented in the CISCO-QNS-MIB. A summary of the information available is provided in the following sections.

SNMP System KPIs

In this table the system KPI information is provided.

Table 5: SNMP System KPIs

Component	Information
LB01/LB02 PCRFClient01/PCRFClient02 SessionMgr01/SessionMgr02 QNS01/QNS02/QNS03/QNS04	CpuUser CpuSystem CpuIdle LoadAverage1 LoadAverage5 LoadAverage15 MemoryTotal MemoryAvailable SwapTotal SwapAvailable

Note Except for an AIO (All-In-One) deployment all components or devices are VMs.

Details of SNMP System KPIs

The following information is available and is listed per component. The root of these KPIs is .1.3.6.1.4.1.26878.200.3.2.70. MIB documentation provides units of measure.

```
+--ciscoProductsQNSComponents70 (70) |
 +--ciscoProductsQNSComponentsSystemStats (1) |
    +-- -R-- Integer32 componentCpuUser(1) |
    +-- -R-- Integer32 componentCpuSystem(2) |
    +-- -R-- Integer32 componentCpuIdle(3) |
    +-- -R-- Integer32 componentLoadAverage1(4) |
    +-- -R-- Integer32 componentLoadAverage5(5) |
    +-- -R-- Integer32 componentLoadAverage15(6) |
```

```
+-- -R-- Integer32 componentMemoryTotal(7) |
+-- -R-- Integer32 componentMemoryAvailable(8) |
+-- -R-- Integer32 componentSwapTotal(9) |
+-- -R-- Integer32 componentSwapAvailable(10) |
```

SNMP Application KPIs

Current version Key Performance Indicators (KPI) information is available at the OID root of:

```
.1.3.6.1.4.1.26878.200.3.3.70
```

This corresponds to an MIB of:

```
.iso
.identified-organization
.dod
.internet
.private
.enterprise
.broadhop
.broadhopProducts
.ciscoProductsQNS
.ciscoProductsQNSConsolidatedKPIVersion
.ciscoProductsQNSKPI70
```

Summary of SNMP Application KPIs

The following application KPIs are available for monitoring on each node using SNMP Get and Walk utilities:

Table 6: SNMP Application KPIs - Summary

Component	Information
Policy Director (lb01/lb02)	**PCRFProxyExternalCurrentSessions:** It is the total number of active sessions (open connections) which are connected to lbvip01:8443 from external system (lbvip01 has public IP address). It is an active session counter (not cumulative) and as such there is no limit on active sessions. **PCRFProxyInternalCurrentSessions:** It is the total number of active sessions (open connections) which are connected to lbvip02:8080 (lbvip02 has private IP address) from internal VMs such as Policy Server (QNS), sessionmgr, OAM (pcrfclient) and so on. It is an active session counter (not cumulative) and as such there is no limit on active sessions.
OAM (pcrfclient01/pcrfclient02)	----------
Session Manager (sessionmgr01/sessionmgr02)	----------

Component	Information
Policy Server (qns01/qns02/qns03/qns04)	**PolicyCount:** It is the total number of processed policy messages by an individual Policy Server (QNS) VM. There is no limit on policy message processing. **QueueSize:** The number of entries in the processing queue. The default queue size is 500, and is configurable in Policy Builder. You can also see the number of dropped messages in the statistics files. There is a separate queue for each Policy Server (QNS) VM. **FailedEnqueueCount:** Each Policy Server (QNS) VM maintains a queue where it keeps policy messages to be processed in last-in-first-out order. This counter will be incremented when Policy Server (QNS) process fails to add policy message into policy message processing queue. **ErrorCount:** It is the total number of policy messages which got error while processing by an individual Policy Server (QNS) VM. **AggregateSessionCount:** This is the consolidated active subscriber sessions in CPS. The maximum limit of sessions will be based on installed license. It is only active session count not cumulative count. AggregateSessionCount is the consolidated active subscriber sessions in CPS and kpiLBPCRFProxyInternalCurrentSessions is the open connection to lbvip02:8080. FreeMemory

Details of Supported KPIs

The following information is available and is supported in current release. MIB documentation provides units of measure.

```
+--ciscoProductsQNSKPILB(11)
| |
| +-- -R-- String kpiLBPCRFProxyExternalCurrentSessions(1)
| |      Textual Convention DisplayString
| |      Size 0..255
| +-- -R-- String kpiLBPCRFProxyInternalCurrentSessions(2)
|        Textual Convention DisplayString
|        Size 0..255

+--ciscoProductsQNSKPISessionMgr(14)
+--ciscoProductsQNSKPIQNS(15)
| |
| +-- -R-- Integer32 kpiQNSPolicyCount(20)
| +-- -R-- Integer32 kpiQNSQueueSize(21)
| +-- -R-- Integer32 kpiQNSFailedEnqueueCount(22)
| +-- -R-- Integer32 kpiQNSErrorCount(23)
```

```
| +-- -R-- Integer32 kpiQNSAggregateSessionCount(24)
| +-- -R-- Integer32 kpiQNSFreeMemory(25)
```

Threshold based KPI Alarms

CPS can generate SNMP alarms for KPIs after they have reached threshold values. The threshold values are configured in the `/etc/broadhop/kpi_threshold.conf` file. The `kpi_threshold.conf` configuration file contains all the KPI configurations and must be configured to generate the KPI traps. The configuration file must be present on all VMs.

Events generated by the KPI script are locally logged in pcrfclient01/02 in the `/var/log/broadhop/kpi-alarm.log` file. The following table defines the configuration parameters:

Table 7: KPI Configuration Parameters

Parameter	Description
GV_LOG_LEVEL	Log levels are as follows: • 1: DEBUG • 2: INFO • 3: WARN • 4: ERROR for example, GV_LOG_LEVEL= logging.INFO
GV_LOG_FILE	Log file path and log file name. For example, GV_LOG_FILE="/var/log/broadhop/kpi-alarm.log
GV_LOG_FILES	Number of log files to preserve. For example, GV_LOG_FILES=5
GV_LOG_SIZE	Log file size. For example, GV_LOG_SIZE=10 * 1024 * 1024 #10MB
GV_STATS_INTERVAL=300	Statistics collected during last 300 seconds.

Traps generated are logged in the `/var/log/snmp/trap` file on the active Policy Director (LB).

Notifications and Alerting (Traps)

The CPS Monitoring and Alert Notification framework provides the following SNMP notification traps (one-way). Traps are either proactive or reactive. Proactive traps are alerts based on system events or changes that require attention (for example, Disk is filling up). Reactive traps are alerts that an event has already occurred (for example, an application process failed).

For example, if a threshold is crossed snmpd throws a trap to LBVIP on the internal network on port 162. On the Policy Director (load balancer) the snmptrapd process is listening on port 162. When snmptrapd sees trap on 162 it logs it in the file `/var/log/snmp/trap` and throws it again on `corporate_nms_ip` on port 162. This corporate NMS IP is set inside `/etc/hosts` file on LB01 and LB02.

Component Notifications

Components are devices that make up the CPS system. These are systems level traps. They are generated when some predefined thresholds are crossed. User can define these thresholds in `/etc/snmp/snmpd.conf`. For example, for disk full, low memory etc. The snmpd process runs on all VMs. When the process is started, it applies the configuration from `/etc/snmp/snmpd.conf` file. In order to apply changes to snmpd.conf file, snmpd needs to be restarted by executing the following commands:

```
monit stop snmpd

monit start snmpd
```

Component notifications are defined in the BROADHOP-NOTIFICATION-MIB as follows:

```
broadhopQNSComponentNotification NOTIFICATION-TYPE
  OBJECTS { broadhopComponentName,
   broadhopComponentTime,
   broadhopComponentNotificationName,
   broadhopNotificationFacility,
   broadhopNotificationSeverity,
   broadhopComponentAdditionalInfo }
STATUS current
DESCRIPTION "
Trap from any QNS component - i.e. device.

"
::= { broadhopProductsQNSNotifications 1 }
```

Component Notifications that CPS generates are shown in the following list. Any component in the CPS system may generate these notifications.

Table 8: Component Notifications

Notification Name	Severity	Feature
DiskFull	critical	Component
	Message Text: <*diskPath*>: less than <*n*>% free (= REMAINING_DISK_SPACE%) **Description:** Current disk usage has passed a designated threshold. By default, this threshold is set to 10% of total disk space allocated for the partition. This threshold is defined in `/etc/snmp/snmpd.conf` on each VM. This situation could be a sign of logs or database files growing large. For new deployments, this alarm is generated for following file systems in different VMs: • For HA System: • pcrfclient/lb: / • sessionmgr: /, /var/data/session.1 • qns: / • For AIO System: • / For upgraded systems, this alarm is generated for following file systems in different VMs: • For HA System: • pcrf/lb: /, /var, /boot • sessionmgr: /, /home, /boot, /data, /var/data/session.1 • qns: /, /home, /var, /boot • For AIO System: • / • /boot	
	clear	Component

Notification Name	Severity	Feature
	Message Text: <*diskPath*>: clear **Description:** The disk usage has recovered from the designated threshold. For new deployments, this alarm is generated for following file systems in different VMs: • For HA System: • pcrfclient/lb: / • sessionmgr: /, /var/data/session.1 • qns: / • For AIO System: • / For upgraded systems, this alarm is generated for following file systems in different VMs: • For HA System: • pcrf/lb: /, /var, /boot • sessionmgr: /, /home, /boot, /data, /var/data/session.1 • qns: /, /home, /var, /boot • For AIO System: • / • /boot	
LowSwap	critical	Operating System
	Message Text: Running out of swap space ($FreeAvailableSwap) **Description:** Current swap usage has passed a designated threshold. This is a warning.	
	clear	Operating System
	Message Text: Swap space recovered **Description:** Current swap usage has recovered a designated threshold.	

Notification Name	Severity	Feature
HighLoad	warning (1 minute) warning (5 minute) alert (15 minutes)	Component
	Message Text: 1 min Load Average too high (= n.nn) 5 min Load Average too high (= n.nn) 15 min Load Average too high (=n.nn) **Description:** The load average of the system has exceeded the configured threshold for a period of 1/5/15 minutes. The default threshold value is 1.5 * Number of vCPUs (allocated to VM) for each time period as defined in `/etc/snmp/snmpd.conf` file. The value must be integer.	
	clear	Component
	Message Text: Load-1 High load recovered Load-5 High load recovered Load-15 High load recovered **Description:** The load average has recovered from more than configured threshold.	
LinkDown	alert	Operating System
	Message Text: IF-MIB::linkDown <*Interface Name*> **Description:** Not able to connect or ping to the interface. This alarm gets generated for all physical interface attached to the system.	
LinkUp	clear	Operating System
	Message Text: IF-MIB::linkUp <*Interface Name*> **Description:** Able to ping or connect to interface. This alarm gets generated for all physical interface attached to the system.	
Low Memory Alert	critical	Operating System
	Message Text: Current Available Free Memory (total free memory) is less than threshold (Threshold memory) on $HOSTNAME **Description:** The amount of free memory on the VM has dropped below the default threshold of 10% (as a percentage of total memory). To change the default threshold, see Configure Low Memory Threshold, on page 16.	

Notification Name	Severity	Feature
Low Memory Clear	clear	Operating System
	Message Text: Current Available Free Memory (total free memory) is greater than threshold (Threshold memory) on $HOSTNAME **Description:** Low memory alert has been cleared.	
ProcessDown	critical	Component
	Message Text: ${PROCESS_NAME} process is down For example, corosync process is down **Description:** This alarm is generated when the corosync process is stopped or fails. The corosync process manages the virtual IPs between the CPS load balancers in HA and GR deployments.	
ProcessUp	clear	Component
	Message Text: ${PROCESS_NAME} process is up For example, corosync process is up **Description:** The alarm is cleared whenever the corosync process that was down is brought back up.	
HIGH CPU USAGE Alert	critical	Component
	Message Text: CPU Usage is higher than threshold on `hostname`.Threshold=$Threshold%,Current_LOAD=$Current% **Description:** This trap is generated whenever CPU usage on any VM is detected to be higher than the alert threshold value. The system monitors the CPU usage at a specific instant (every 60 second by default), and not over a period of time like for the HighLoad Alert. To change the default threshold or the interval at which the CPU usage is checked, see Configure High CPU Usage Alarm Thresholds and Interval Cycle, on page 16	
HIGH CPU USAGE Clear	clear	Component
	Message Text: CPU Usage is below than lower threshold value on `hostname`.Threshold=$Threshold%,Current_LOAD=$Current% **Description:** This trap is generated whenever CPU usage on any VM is lower than the clear threshold value. It is generated only when High CPU Usage Alert was generated earlier for the VM.	

Each Component Notification contains:

- Name of the Notification being thrown (broadhopComponentNotificationName)
- Name of the device throwing the notification (broadhopComponentName)
- Time the notification was generated (broadhopComponentTime)
- Facility or which layer the notification came from (broadhopNotificationFacility)

- Severity of the notification (broadhopNotificationSeverity)
- Additional information about the notification, which might be a bit of log or other information.

Configure Low Memory Threshold

By default the Low Memory Alert is generated when the available memory of any CPS VM drops below 10% of the Total Memory. To change the default threshold:

Step 1 Modify the following parameter in the Configuration worksheet of the CPS Deployment template spreadsheet.

The CPS Deployment template can be found on the Cluster Manager VM:

`/var/qps/install/current/scripts/deployer/templates/QPS_deployment_config_template.xlsm`

- free_memory_per_alert: Enter a value (0.0-1.0) for the alert threshold. The system will generate an Alert trap whenever the available memory falls below this percentage of total memory for any given VM. Default 0.10 (10% free of the total memory).
- free_memory_per_clear: Enter a value (0.0-1.0) for the clear threshold. The system will generate a low memory clear trap whenever available memory for any given VM is more than 30% of total memory. Default 0.3 (30% of the total memory).

Step 2 Follow the steps in the Update the VM Configuration without Re-deploying VMs section of the *CPS Installation Guide for VMware* to push the new settings out to all CPS VMs.

Configure High CPU Usage Alarm Thresholds and Interval Cycle

To change the default threshold values and interval cycle for the High CPU Usage traps and apply the new values to all CPS VMs:

Step 1 Modify the following parameters in the Configuration worksheet of the CPS Deployment template spreadsheet.

The CPS Deployment template can be found on the Cluster Manager VM:

`/var/qps/install/current/scripts/deployer/templates/QPS_deployment_config_template.xlsm`

Note The alert threshold must be set higher than the clear threshold.

- cpu_usage_alert_threshold: Enter an integer (0-100) for the alert threshold value. The system will generate an Alert trap whenever the CPU usage is higher than this value. Default 80.
- cpu_usage_clear_threshold: Enter an integer (0-100) for the clear threshold value. The system will generate a Clear trap whenever the CPU usage is lower than this value and alert trap already generated. Default 40.
- cpu_usage_trap_interval_cycle: Enter an integer value to be used as an interval period to execute the CPU usage trap script. The interval value in seconds is calculated by multiplying 5 with the given value.

The default cpu_usage_trap_interval_cycle value is 12 which means the script will get executed every 60 seconds.

Step 2 Follow the steps in the Update the VM Configuration without Re-deploying VMs section of the *CPS Installation Guide for VMware* to push the new settings out to all CPS VMs.

Application Notifications

Applications are running processes on a component device that make up the CPS system. These are application level traps. CPS processes (starting with word java when we run "ps -ef") and some scripts (for GR traps) generates these traps.

Application notifications are defined in the BROADHOP-NOTIFICATION-MIB as follows:

```
broadhopQNSComponentNotification NOTIFICATION-TYPE
  OBJECTS { broadhopComponentName,
   broadhopComponentTime,
   broadhopComponentNotificationName,
   broadhopNotificationFacility,
   broadhopNotificationSeverity,
   broadhopComponentAdditionalInfo }
STATUS current
DESCRIPTION "
Notification Trap from any QNS component - i.e. runtime
"
::= { broadhopProductsQNSNotifications 2 }
```

Each Application Notification contains:

- Name of the Notification being thrown (broadhopComponentNotificationName)
- Name of the device throwing the notification (broadhopComponentName)
- Time the notification was generated (broadhopComponentTime)
- Facility or which layer the notification came from (broadhopNotificationFacility)
- Severity of the notification (broadhopNotificationSeverity)
- Additional information about the notification, which might be a bit of log or other information.

Important Currently, third site arbiter supports only Arbiter Down and Arbiter Up traps.

Application Notifications that CPS generates are shown in the following list. Any component in the CPS system may generate these notifications.

Table 9: Application Notifications

Notification Name	Severity	Feature
MemcachedConnectError	error critical	Application
	Message Text: ${HOSTNAME}: Memcached server is in error OR Memcached server is in error : <with exception> **Description:** Generated if attempting to connect to or write to the memcached server causes an exception.	
	clear	Application
	Message Text: ${HOSTNAME}: Memcached server is operational **Description:** Generated if successfully connect to or write to the memcached server.	
ApplicationStartError	alert	Application
	Message Text: ${HOSTNAME}: Feature %s is unable to start. Error - %s **Description:** Generated if an installed feature cannot start.	
	clear	Application
	Message Text: ${HOSTNAME}: Feature %s is Running **Description:** Generated if an installed feature successfully started.	

Notification Name	Severity	Feature
License Usage Threshold Exceeded	critical, error, notice, warning (Configurable)	Application
	Message Text: ${HOSTNAME}: Session Count License Usage at: xxx%, threshold is:xxx% **Description:** The number of sessions on the system has exceeded the configured threshold of sessions allowed by the current license. The threshold value and alarm severity of this alarm is configurable in Policy Builder: Click Fault List in the navigation pane, then create a new fault list or edit the existing fault list. By default, the threshold is set to 90%.	
	clear	Application
	Message Text: ${HOSTNAME}: Session Count License Usage at: xxx%, threshold is:xxx% **Description:** The number of sessions on the system is below the configured threshold of sessions allowed by the current license.	
LicensedSessionCreation	critical	Application
	Message Text: ${HOSTNAME}: Session creation is not allowed **Description:** A predefined threshold of sessions covered by licensing has been passed. This is a warning and should be reported. License limits may need to be increased soon. This message can be generated by an invalid license, but the AdditionalInfo portion of the notification shows root cause.	
	clear	Application
	Message Text: ${HOSTNAME}: Session creation is allowed **Description:** The number of sessions are below the predefined threshold of sessions covered by licensing.	

Notification Name	Severity	Feature
InvalidLicense	emergency	Application
	Message Text: ${HOSTNAME}: xxx license has not been verified yet. **Description:** The system license currently installed is not valid. This prevents system operation until resolved. This is possible if no license is installed or if the current license does not designate values. This may also occur if any of the VMs MAC addresses change.	
	emergency	Application
	Message Text: ${HOSTNAME}: xxx license is Invalid. %s **Description:** License is invalid. For example, if RADIUS feature is installed and the license for the same is not installed, then this alarm is generated. **Note** RADIUS-based policy control is no longer supported in CPS 14.0.0 and later releases as 3GPP Gx Diameter interface has become the industry-standard policy control interface.	
	critical	Application
	Message Text: ${HOSTNAME}: xxx license is Expired. %s **Description:** License has expired.	
	error	Application
	Message Text: ${HOSTNAME}: xxx license will Expire Soon. %s **Description:** License is going to expire soon.	
	critical	Application
	Message Text: ${HOSTNAME}: xxx license has exceeded the allowed parameters. %s **Description:** License has exceeded the allowed parameters.	
	error	Application
	Message Text: ${HOSTNAME}: xxx license is nearing the allowed parameters. %s **Description:** RADIUS AAA proxy server is reachable. **Note** RADIUS-based policy control is no longer supported in CPS 14.0.0 and later releases as 3GPP Gx Diameter interface has become the industry-standard policy control interface.	
	clear	Application

Notification Name	Severity	Feature
	Message Text: ${HOSTNAME}: license is Valid. **Description:** License is valid.	
PolicyConfiguration	error	Application
	Message Text: ${HOSTNAME}: Last policy configuration failed with the following message: xxx **Description:** A change to system policy structure has failed. The AdditionalInfo portion of the notification contains more information. The system typically remains in a proper state and continues core operations. Either make note of this message or investigate more fully.	
	clear	Application
	Message Text: ${HOSTNAME}: Last policy configuration was successful **Description:** A change to system policy structure has passed.	
PoliciesNotConfigured	emergency	Application
	Message Text: ${HOSTNAME}: 1001Policies not configured **Description:** The policy engine cannot find any policies to apply while starting up. This may occur on a new system, but requires immediate resolution for any system services to operate.	
	clear	Application
	Message Text: ${HOSTNAME}: 1001:Policies successfully configured **Description:** The policy engine has successfully configured all the policies while starting up.	

Notification Name	Severity	Feature
DiameterPeerDown	error	Application
	Message Text: ${HOSTNAME}: 3001:Host: %s Realm: %s is down OR ${HOSTNAME}: 3001:Host: %s Realm: %s PeerIP: %s is down OR ${HOSTNAME}: 3001:Host: %s Realm: %s PeerIP: %s Interface: %s is down **Description:** Diameter peer is down.	
	clear	Application
	Message Text: ${HOSTNAME}: 3001:Host: %s Realm: %s is back up OR ${HOSTNAME}: 3001:Host: %s Realm: %s PeerIP: %s is back up OR ${HOSTNAME}: 3001:Host: %s Realm: %s PeerIP: %s Interface: %s is back up **Description:** Diameter peer is up.	
DiameterAllPeersDown	critical	Application
	Message Text: ${HOSTNAME}: 3002:Realm: %s:applicationId: %s:all peers are down **Description:** All Diameter peer connections configured in a given realm are DOWN (i.e. connection lost). The alarm identifies which realm is down. The alarm is cleared when at least one of the peers in that realm is available.	
	clear	Application
	Message Text: ${HOSTNAME}: 3002:Realm: %s:applicationId: %s:peers are up **Description:** The Diameter peer connections configured in a given realm are up.	

Notification Name	Severity	Feature
DiameterStackNotStarted	critical	Application
	Message Text: ${HOSTNAME}: 3004:Error starting diameter stack: <stack uri>. Reason: <error message> **Description:** This alarm is generated when Diameter stack cannot start on a particular policy director (load balancer) due to some configuration issues.	
	clear	Application
	Message Text: ${HOSTNAME}: 3004:Stack <stack uri> is running. **Description:** The Diameter stack has started successfully.	
HA Failover	critical	Application
	Message Text: "${HOSTNAME}: HA Failover done from $previous_member to $PRIMARYNODE of ${SET_NAME}-SET$Loop" **Description:** The primary role of the replica set has been failed over to another member.	
GR Failover	critical	Application
	Message Text: "${HOSTNAME}: Geo Failover done from $previous_member to $PRIMARYNODE of ${SET_NAME}-SET$Loop" **Description:** The primary role of the replica set has been failed over to another member.	
All DB Member of replica set Down	critical	Application
	Message Text: "${HOSTNAME}: All DB members of replica set ${SET_NAME}-SET$Loop are down" **Description:** Not able to connect to any member of the replica set.	
All DB Member of replica set Up	clear	Application
	Message Text: "${HOSTNAME}: All DB members of replica set ${SET_NAME}-SET$Loop are up" **Description:** Able to connect to all members of the replica set.	
No Primary DB Member Found	critical	Application
	Message Text: "${HOSTNAME}: Unable to find primary member for Replica-set ${SET_NAME}-SET$Loop" **Description:** Unable to find primary member for the replica-set.	

Notification Name	Severity	Feature
Primary DB Member Found	clear	Application
	Message Text: "${HOSTNAME}: Found primary member $member for Replica-set ${SET_NAME}-SET$Loop" **Description:** Found primary member for the replica-set.	
DB Member Down	critical	Application
	Message Text: "${HOSTNAME}: DB_Member $member of SET $SET is down" OR "${HOSTNAME}: DB_Member $member_ip:$mem_port ($mem_hostname) of SET $SET is down" **Description:** A secondary member of the replica set is down.	
DB Member Up	clear	Application
	Message Text: "${HOSTNAME}: DB_Member $member of SET $SET is up" OR "${HOSTNAME}: DB_Member $member_ip:$mem_port ($mem_hostname) of SET $SET is up" **Description:** A secondary member of the replica set has come back up.	
Arbiter Down	critical	Application
	Message Text: "${HOSTNAME}: Arbiter $member of SET $SET is down" OR "${HOSTNAME}: Arbiter $member_ip:$mem_port ($mem_hostname) of SET $SET is down" **Description:** The arbiter member of the replica set is not reachable.	
Arbiter Up	clear	Application
	Message Text: "${HOSTNAME}: Arbiter $member of SET $SET is up" OR "${HOSTNAME}: Arbiter $member_ip:$mem_port ($mem_hostname) of SET $SET is up" **Description:** The arbiter member of the replica set is functional.	

Notification Name	Severity	Feature
DB Resync is needed	critical	Application
	Message Text: "${HOSTNAME}: Resync is needed for secondary member $setRepl:$SET_NAME:$DB_MEMBER, this member is lagging behind by $SLAVE_BEHIND_SECS seconds from the primary" **Description:** Generated whenever a manual resynchronization of a database is required to recover from a failure.	
DB Resync is not needed	clear	Application
	Message Text: "${HOSTNAME}: Resync is not needed for member $setRepl:$SET_NAME:$DB_MEMBER" OR "${HOSTNAME}: Resync is not needed for secondary member $setRepl:$SET_NAME:$DB_MEMBER" **Description:** Generated whenever a database changes to 'Good' state from 'Resync is needed' state, it indicates that the database's resynchronization has completed.	
Config Server Down	critical	Application
	Message Text: "${HOSTNAME}: Config_Server $member of SET $SET is down" OR "${HOSTNAME}: Config_Server $member_ip:$mem_port ($mem_hostname) of SET $SET is down" **Description:** The configuration server for the replica set is unreachable. Not valid for non-sharded replica sets.	
Config Server Up	clear	Application
	Message Text: "${HOSTNAME}: Config_Server $member of SET $SET is up" OR "${HOSTNAME}: Config_Server $member_ip:$mem_port ($mem_hostname) of SET $SET is up" **Description:** The configuration server for the replica set is reachable. Not valid for non-sharded replica sets.	

Notification Name	Severity	Feature
VM Down	critical	Application
	Message Text: "${HOSTNAME}: unable to connect $member_ip ($member) VM. It is not reachable" **Description:** The administrator is not able to ping the VM.	
VM Up	clear	Application
	Message Text: "${HOSTNAME}: Connected $member_ip ($member) VM. It is reachable" **Description:** The administrator is able to ping the VM.	
QNS Process Down	critical	Application
	Message Text: "${HOSTNAME}: $server (<qns instance id>) server on $VM_HOSTNAME vm is down" **Description:** Policy Server (qns-<instance_id>) java process on particular QNS instance is down.	
QNS Process Up	clear	Application
	Message Text: "${HOSTNAME}: $server (<qns instance id>) server on $VM_HOSTNAME vm is up" **Description:** Policy Server (qns-<instance_id>) java process on particular QNS instance is up.	
Admin User Logged in	info	Application
	Message Text: "${HOSTNAME}: root user logged in on `hostname` terminal $terminal from machine $from_system at $dt" **Description:** root user logged in on %hostname terminal.	
DeveloperMode	error	Application
	Message Text: ${HOSTNAME}: Using Developer mode(100 session limit).To use a license file, remove -Dcom.broadhop.developer.mode from /etc/broadhop/qns.conf **Description:** Generated if developer mode is configured in `qns.conf` file.	
	clear	Application
	Message Text: ${HOSTNAME}: -Dcom.broadhop.developer.mode is disabled **Description:** Generated if developer mode is removed in `qns.conf` file.	

Notification Name	Severity	Feature
ZeroMQConnectionError	error	Application
	Message Text: ${HOSTNAME}: ZMQ Connection Down for %s **Description:** Internal services cannot connect to a required Java ZeroMQ queue. Although retry logic and recovery is available, and core system functions should continue, investigate and remedy the root cause.	
	clear	Application
	Message Text: ${HOSTNAME}: ZMQ Connection Up for %s **Description:** Internal services can connect to a required Java ZeroMQ queue.	
VirtualInterface Down	alert	Application
	Message Text: "${HOSTNAME}: unable to connect ${member}. Not reachable" **Description:**Not able to ping the virtual Interface. This alarm is generated for external VIPs. For example, lbvip01.	
VirtualInterface Up	clear	Application
	Message Text: "${HOSTNAME}: ${member} is up" **Description:** Successfully ping the virtual Interface. This alarm is generated for external VIPs. For example, lbvip01.	
VirtualInterfaceDown	alert	Application
	Message Text: "unable to connect ${member}. Not reachable" **Description:** Not able to ping the internal VIPs.	
VirtualInterfaceUp	clear	Application
	Message Text: "${member} is up" **Description:** Able to ping internal VIPs.	
Site Down	alert	Application
	Message Text: "${HOSTNAME}: Site $site is down" **Description:** Site is down. This alarm is related to GR deployments.	

Notification Name	Severity	Feature
Site Up	clear	Application
	Message Text: "${HOSTNAME}: Site $site is up." OR "${HOSTNAME}: Site $site is up" **Description:** Site is Up. This alarm is related to GR deployments.	
LDAPAllPeersDown	error	Application
	Message Text: ${HOSTNAME}: 1201:<LocalHostname>:LDAP connection down **Description:** All LDAP peers are down.	
	clear	Application
	Message Text: ${HOSTNAME}: 1201:<LocalHostname>:LDAP connection up **Description:** LDAP connection is up.	
LDAPPeerDown	error	Application
	Message Text: ${HOSTNAME}: 1202:<IP Address of the LDAP server>:LDAP connection down **Description:** LDAP peer identified by the IP address is down.	
	clear	Application
	Message Text: ${HOSTNAME}: 1202:<IP Address of the LDAP server>:LDAP connection up **Description:** LDAP peer identified by the IP address is up.	
Percentage of LDAP retry threshold Exceeded	critical	Application
	Message Text: ${HOSTNAME}: Percentage of LDAP retries compared to total LDAP Queries exceeded to $CURRENT_LEVEL% on $HOST VM. **Description:** This alarm indication is generated for LDAP search queries when LDAP retries compared to total LDAP queries exceeds 10% on qnsXX VM. Default Threshold: 10% **Note** The LDAP server Retry Count parameter must be set to a value greater than 1 for this alarm to be generated. In Policy Builder navigate to Plugin Configuration > LDAP Configuration > LDAP Server Configuration > Retry Count.	

Notification Name	Severity	Feature
Percentage of LDAP retry threshold Normal	clear	Application
	Message Text: ${HOSTNAME}: Percentage of LDAP retries compared to total LDAP Queries normal to $CURRENT_LEVEL% on $HOST VM. **Description:** This clear indication is generated for LDAP search queries when LDAP retries copmared to total LDAP queries is normal or has fallen below the threshold value (10%) on qnsXX VM.	
LDAP Requests as percentage of CCR-I Dropped	critical	Application
	Message Text: ${HOSTNAME}: LDAP Requests as percentage of CCR-I dropped to $CURRENT_LEVEL% on $HOST VM. **Description:** This alarm indication is generated for LDAP operations when LDAP requests as percentage of CCR-I (Gx messages) drops below 25% on qnsXX VM. Default Threshold: 25%	
LDAP Requests as percentage of CCR-I Normal	clear	Application
	Message Text: ${HOSTNAME}: LDAP Requests as percentage of CCR-I normal to $CURRENT_LEVEL% on $HOST VM. **Description:** This clear indication is generated for LDAP operations when LDAP requests as a percentage of CCR-I messages is normal or above the 25% threshold on qnsXX VM.	
LDAP Requests Dropped	critical	Application
	Message Text: ${HOSTNAME}: LDAP Requests dropped to $CURRENT_LEVEL on $HOST VM. **Description:** This alarm indication is generated for LDAP operations when LDAP requests drop below 0 on lbXX VM. Default Threshold: 0	
LDAP Requests Normal	clear	Application
	Message Text: ${HOSTNAME}: LDAP Requests normal to $CURRENT_LEVEL on $HOST VM. **Description:** This clear indication is generated for LDAP operations when LDAP requests are normal (above 0) on lbXX VM.	

Notification Name	Severity	Feature
LDAP Query Result Dropped	critical	Application
	Message Text: ${HOSTNAME}: LDAP Query Result dropped to $CURRENT_LEVEL on $HOST VM. **Description:** This alarm indication is generated when LDAP Query Result goes to 0 on qnsXX VM. Default Threshold: 0	
LDAP Query Result Normal	clear	Application
	Message Text: ${HOSTNAME}: LDAP Query Result normal to $CURRENT_LEVEL on $HOST VM. **Description:** This clear indication is generated when LDAP Query Result goes above 0 (above the threshold value) on qnsXX VM.	
Gx Message processing Dropped	critical	Application
	Message Text: ${HOSTNAME}: Gx Message $MSG_TYPE dropped to $CURRENT_LEVEL% on $HOST_VM VM. **Description:** This alarm indication is generated for Gx Message CCR-I, CCR-U and CCR-T when processing of messages drops below 95% on qnsXX VM. The 95% refers to the percentage of responses to the requests within a 60 second period of time. For example, in 60 sec if you receive 100 requests and send 95 responses then your percentage would be 95%. Default threshold: 95%	
Gx Message processing Normal	clear	Application
	Message Text: ${HOSTNAME}: Gx Message $MSG_TYPE normal to $CURRENT_LEVEL% on $HOST_VM VM. **Description:** This clear indication is generated for Gx Message CCR-I, CCR-U and CCR-T when processing of messages is equal or above 95% on qnsXX VM.	
Gx Average Message processing Dropped	critical	Application
	Message Text: ${HOSTNAME}: Gx average Message $MSG_TYPE processing dropped to ${CURRENT_LEVEL}ms on $HOST_VM VM. **Description:** This alarm indication is generated for Gx Message CCR-I, CCR-U and CCR-T when average message processing is above 20ms on qnsXX VM. Default Threshold: 20ms	

Notification Name	Severity	Feature
Gx Average Message processing Normal	clear	Application
	Message Text: ${HOSTNAME}: Gx average Message $MSG_TYPE processing normal to ${CURRENT_LEVEL}ms on $HOST_VM VM. **Description:** This clear indication is generated for Gx Message CCR-I, CCR-U and CCR-T when average message processing is equal or below 20ms on qnsXX VM.	
All SMSC server connections are down	critical	Application
	Message Text: ${HOSTNAME}: 5002:<VMName>:All SMSC servers not reachable **Description:** None of the SMSC servers configured are reachable. This Critical Alarm is generated when the SMSC Server endpoints are not available to submit SMS messages thereby blocking SMS from being sent from CPS.	
Atleast one SMSC server connection is up	clear	Application
	Message Text: ${HOSTNAME}: 5002:<VMName>:Atleast one SMSC server is reachable **Description:** This alarm (Clear) is generated when at least one configured SMSC endpoint server is reachable after a state where none were reachable from the mconfigured list of server endpoints.	
SMSC server connection down	error	Application
	Message Text: ${HOSTNAME}: 5001:<SMSCServer Address>:<SMSC Port>:SMSC Server not reachable **Description:** SMSC Server is not reachable. This alarm is generated when any one of the configured active SMSC server endpoints is not reachable and CPS will not be able to deliver a SMS via that SMSC server.	
SMSC server connection up	clear	Application
	Message Text: ${HOSTNAME}: 5001:<SMSCServer Address>:<SMSC Port>:SMSC server reachable **Description:** This alarm (Clear) is generated when an earlier unreachable SMSC endpoint is now reachable.	
All Email servers not reachable	critical	Application
	Message Text: ${HOSTNAME}: 5004:<VMName>:All Email Servers not reachable **Description:** No email server is reachable. This alarm (Critical) is generated when all configured Email Server Endpoints are not reachable, blocking e-mails from being sent from CPS.	

Notification Name	Severity	Feature
At least one Email server is reachable	clear	Application
	Message Text: ${HOSTNAME}: 5004:<VMName>:At least one Email server is reachable **Description:** At least one email server is reachable.	
Email server is not reachable	error	Application
	Message Text: ${HOSTNAME}: 5003:<Mail Server Address>:<SMTP Port>Email Server not reachable **Description:** Email server is not reachable. This alarm is generated when any of the configured Email Server Endpoints are not reachable. CPS is not able to use the server to send e-mails.	
Email server is reachable	clear	Application
	Message Text: ${HOSTNAME}: 5003:<Mail Server Address>:<SMTP Port>Email Server reachable **Description:** Email server is reachable. This alarm (Clear) is generated when an earlier unreachable Email server endpoint is now reachable.	
Binding Not Available at Policy DRA	Critical, Error, Minor, Warning	Application
	Message Text: Binding DB not accessible or Binding Db not reachable at Policy DRA **Description:** This alarm is generated when IPv6 binding for sessions is not found at Policy DRA. Only one notification is sent out whenever this condition is detected. This is a configurable notification. You can configure whether to send or not to send the notification. For more information, refer to *PolicyDRA Health Check* under *Diameter Configuration* in *CPS Mobile Configuration Guide*.	
	clear	Application
	Message Text: Binding DB Available at Policy DRA or Binding Db reachable at Policy DRA **Description:** The clear severity alarm is generated after the duration of Alarm Clearance Interval (configured under **Diameter Configuration** > **PolicyDRA Health Check** > **Alarm Config** > **Alarm Clearance Interval** in Policy Builder) when the above alarm was generated.	

Notification Name	Severity	Feature
SPR_DB_ALARM	major	Application
	Message Text: 6101:Remote SPR DB:Error adding remote spr db **Description:** This alarm indicates there is an issue in establishing connection to the Remote SPR Databases configured under **USuM Configuration** > **Remote Database Configuration** during CPS policy server (qns) process initialization.	
	clear	Application
	Message Text: 6101:Remote SPR DB: Cleared alarm Error adding remote spr db **Description:** The issue of establishing connection to the Remote SPR database has been resolved.	

Configuration to Generate Invalid License Trap

Note If you change a previously installed valid license and make it invalid, the system will not generate any trap. As system is not monitoring the license files, instead it checks the license entries present in admin database. If the database entries are correct, system will not generate any trap.

Step 1 To generate invalid license trap we need to configure the following parameter in `/etc/broadhop/qns.conf` file.

```
-Dcom.cisco.enforcementfree.mode=false
```

Note When `com.cisco.enforcementfree.mode` is configured as false in addition to license has not been verified yet/license is invalid/has exceeded the allowed parameters following traps will be generated:

- is Expired
- will expire soon
- is nearing the allowed parameters

The traps will be generated only when license expiry date is set in license file.

Step 2 After adding the above entry in `qns.conf` file execute `copytoall.sh` to synchronize the configuration changes to all VMs in the CPS cluster:

```
copytoall.sh /etc/broadhop/qns.conf /etc/broadhop/qns.conf
```

Step 3 After modifying the configuration file to make the changes permanent for future use (when any VM is redeployed or restarted) rebuild etc.tar.gz.

```
/var/qps/install/current/scripts/build/build_etc.sh
```

Step 4 Restart the CPS service.

```
/var/qps/bin/control/restartall.sh
```

Unknown Application Events

All of the alarms generated by different VMs are received by the Policy Director (load balancer) VMs.

On the Policy Director VMs a script called application_trapv1_convert processes the received alarms and generates the new alarm based on the received information and sends it to the external NMS. Unknown alarms can come when application_trapv1_convert is not able to process the received alarm. In this case it will generate one of the below seven unknown alarms.

Table 10: Unknown Application Events

Name	Severity	Facility
ApplicationEvent	None	—
DBEvent	None	—
FailoverEvent	None	—
ProcessEvent	None	—
VMEvent	None	—
None	None	Application
UnKnown	None	None

Note Any unknown alarms should get reported to engineering team to take necessary action against it. Provide the alarm log (`/var/log/snmp/trap`) from the active Policy Director (load balancer) VMs with the ticket number.

Configuration and Usage

All access to system statistics and KPIs should be collected via SNMP gets and walks from the routable IP of the VM. NMS sends the snmpwalk or snmpget request to the routable IP of the VM and gets the response. NMS should know the routable IP addresses of all the VMs available in the setup. System Notifications are sourced from lbvip01.

User can also configure `snmpRouteLan:` parameter which contains the value of a VLAN name which can be used to access the KPIs value provided by SNMP. For more information on the parameter, refer to the *CPS Installation Guide for VMware* or in the *CPS Installation Guide for OpenStack*.

Configuration for SNMP Gets and Walks

By default, SNMPv3 gets and walks can be performed against the routable/public IP addresses of the VMs with the default read-only community string of "broadhop" using standard UDP port 161.

If you want to use SNMPv2 as gets and walks, you need to change the *snmpv3_enable* to `FALSE`.

For more information on SNMP related parameters, refer to general configuration section in the *CPS Installation Guide for VMware* or in the *CPS Installation Guide for OpenStack* for this release.

Configuration for Notifications (traps)

Notifications are logged locally on the Policy Director (load balancer) VMs in the `/var/log/snmp/trap` file as well as forwarded to the NMS destination defined during the installation of CPS.

By default traps are sent to the NMS using the SNMPv2 community string of "broadhop". The standard SNMP UDP trap port of 162 is also used. Both of these values may be changed to accommodate the upstream NMS.

Note If SNMPv3 is enabled, Component Notifications will be sent to NMS via SNMPv3. Application Notifications will be send via SNMPv2.

To change the trap community string for SNMPv2:

1. Configure the snmp_trap_community in Configuration excel sheet on the Cluster Manager VM. For more information, refer to the *Cisco Policy Suite Installation Guide for VMware* for this release. For example:

   ```
   snmp_trap_community cisco
   ```

2. Execute the following command to import csv files into the Cluster Manager VM:

   ```
   /var/qps/install/current/scripts/import/import_deploy.sh
   ```

   This script converts the data to JSON format and outputs it to `/var/qps/config/deploy/import/json/`.

3. Execute `reinit.sh` script to apply the changes to all VMs in the network.

   ```
   /var/qps/install/current/scripts/upgrade/reinit.sh
   ```

To change the destination trap port from 162:

1. To make this change the `/etc/snmp/snmptrapd.conf` file needs to be modified on both lb01 and lb02. In these files append a colon and the destination port to each line containing **`corporate_nms_ip`**. There are a total of 12 lines in each file.

   For example if the NMS destination port were 1162, the line:

   ```
   traphandle DISMAN-EVENT-MIBmteTriggerFired
   ```

   ```
   /etc/snmp/scripts/component_trap_convert corporate_nms_ip
   ```

   becomes

   ```
   traphandle DISMAN-EVENT-MIBmteTriggerFired
   ```

   ```
   /etc/snmp/scripts/component_trap_convert corporate_nms_ip1162
   ```

2. After these changes, save the file and restart the snmptrapd service to enable changes. Run `monit restart snmptrapd` from both Policy Director VMs.

Cluster Manager KPI and SNMP Configuration

This section describes the steps to enable SNMP traps and KPI monitoring of the Cluster Manager so that the customer NMS can monitor the following KPIs:

- Memory usage
- Disk usage
- CPU
- Disk IO

KPIs are reported and recorded on the pcrfclient in the `/var/broadhop/stats` file.

SNMP traps are forwarded to lb01/lb02 and lb01/lb02 forwards the traps to the configured NMS servers in the system.

The following traps are supported for Cluster Manager:

- DiskFull
- HighLoad
- Interface Up/Down
- Swap Usage

Install NET-SNMP

To install NET-SNMP perform the following steps:

Step 1 On the Cluster Manager VM, execute the following command to install NET-SNMP package:

```
yum install --assumeyes --disablerepo=QPS-Repository --enablerepo=QPS-local net-snmp
```

Step 2 To enable run levels for SNMP, execute the following command:

```
chkconfig --level 2345 snmpd on
```

SNMPD Configuration

Note The SNMP configuration mentioned in the following sections is not supported for third site arbiter.

If firewall is configured on Cluster Manager VM, then check if it contains entries for 161 and162 ports.

If the entries for 161 and 162 ports are not there, execute the following command:

```
iptables -A INPUT -i eth0 -p udp -m multiport --ports 161,162 -m comment --comment "100
allow snmp access" -j ACCEPT
```

Check whether IPv6 tables is running and 161 and 162 ports are not there. If the ports are not displayed, then execute the following command:

```
ip6tables -A INPUT -i eth0 -p udp -m multiport --ports 161,162 -m comment --comment "100-6
allow snmp access" -j ACCEPT
```

For SNMPv2

1. Add the following content to `/etc/snmp/snmpd.conf` file on the Cluster Manager:

```
com2sec local     localhost           <snmp_trap_community>
com2sec6 local     localhost          <snmp_trap_community>
rocommunity <snmp_ro_community>
rocommunity6 <snmp_ro_community>
group MyRWGroup v1         local
group MyRWGroup v2c        local
view all    included  .1                                80
access MyRWGroup ""      any       noauth    exact  all    all    none
syslocation Unknown (edit /etc/snmp/snmpd.conf)
syscontact Root  (configure /etc/snmp/snmp.local.conf)
master agents
agentAddress udp:161,udp6:161

trapcommunity  <snmp_trap_community>
agentSecName meme
rouser meme


 # Send all traps upstream - Don't change this password or it breaks the framework.
 # v1 and v2 traps _could_ be sent for all but only need v2 trap.
 trap2sink lbvip02 <snmp_trap_community>


##########
#
# Local Stats
#
ignoreDisk /proc
ignoreDisk /proc/sys/fs/binfmt_misc
ignoreDisk /var/lib/nfs/rpc_pipefs
ignoreDisk /dev/shm
ignoreDisk /dev/pts
disk / 10%


swap 102400

load 6 6 6
```

```
#linkUpDownNotifications yes

notificationEvent  linkUpTrap    linkUp   ifIndex ifAdminStatus ifOperStatus
notificationEvent  linkDownTrap  linkDown ifIndex ifAdminStatus ifOperStatus


monitor  -S -u meme -r 60 -e linkUpTrap -o ifDescr  "Generate linkUp" ifOperStatus != 2
monitor  -u meme -r 60 -e linkDownTrap -o ifDescr "Generate linkDown" ifOperStatus == 2

# Note: alert!=0, clear==0 and messages must be unique or snmpd errors.
monitor -u meme -r 60 -o dskPath -o dskErrorMsg "DiskFullAlert" dskErrorFlag != 0
monitor -S -u meme -r 60 -o dskPath -o dskErrorMsg "DiskFullClear" dskErrorFlag == 0
monitor -u meme -r 60 -o memErrorName -o memSwapErrorMsg "LowSwapAlert" memSwapError !=
 0
monitor -S -u meme -r 60 -o memErrorName -o memSwapErrorMsg "LowSwapClear" memSwapError
 == 0
monitor -u meme -r 60 -o laNames -o laErrMessage  "HighLoadAlert" laErrorFlag != 0
monitor -S -u meme -r 60 -o laNames -o laErrMessage  "HighLoadClear" laErrorFlag == 0


##########
#
# BROADHOP-QNS-MIB Proxy Configuration
#
##########
# proxy -v <version> -c <community> <local_host> <map_to> <map_from>
#
# NOTE:  Most values are listed twice.  This is to cover the snmp get requirement
#        for scalar values.  Snmp get for scalar values (ie. not a table) is
#        required to return for both x.y OID and .x.y.0 OID values.  This only
#        effects <map_to> values.

##########
#
# System Stats
#


#
# LB
#
# User, System and Idle CPU (UCD-SNMP-MIB ss)

proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.1.0
.1.3.6.1.4.1.2021.11.9.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.2.0
.1.3.6.1.4.1.2021.11.10.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.3.0
.1.3.6.1.4.1.2021.11.11.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.1
.1.3.6.1.4.1.2021.11.9.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.2
.1.3.6.1.4.1.2021.11.10.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.3
.1.3.6.1.4.1.2021.11.11.0
# 1, 5 and 15 Minute Load Averages (UCD-SNMP-MIB la)
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.4
.1.3.6.1.4.1.2021.10.1.5.1
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.5
.1.3.6.1.4.1.2021.10.1.5.2
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.6
.1.3.6.1.4.1.2021.10.1.5.3
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.4.0
```

```
.1.3.6.1.4.1.2021.10.1.5.1
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.5.0
.1.3.6.1.4.1.2021.10.1.5.2
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.6.0
.1.3.6.1.4.1.2021.10.1.5.3
# Memory Total, Memory Available, Swap Total, Swap Available (UCD-SNMP-MIB mem)
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.7
.1.3.6.1.4.1.2021.4.5.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.8
.1.3.6.1.4.1.2021.4.6.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.9
.1.3.6.1.4.1.2021.4.3.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.10
.1.3.6.1.4.1.2021.4.4.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.7.0
.1.3.6.1.4.1.2021.4.5.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.8.0
.1.3.6.1.4.1.2021.4.6.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.9.0
.1.3.6.1.4.1.2021.4.3.0
proxy -v 2c -c <snmp_ro_community> localhost .1.3.6.1.4.1.26878.200.3.2.70.1.10.0
.1.3.6.1.4.1.2021.4.4.0
```

2. Replace the string in <*tag*> with the actual value. You can check the `snmpd.conf` from other VMs to get the values for tags. For example, `/etc/snmp/snmpd.conf` file on lb01.
3. You can also update the configuration parameter such as `load 6 6 6` to some other value based on number of vCPUs present on Cluster Manager.

Note Formula is 1.5 * no_of_vCPUs. Consider only the integer value from the output.

Here is an sample `snmpd.conf` file configuration:

```
com2sec local     localhost           cisco123
com2sec6 local     localhost           cisco123
rocommunity cisco_ro
rocommunity6 cisco_ro
group MyRWGroup v1         local
group MyRWGroup v2c        local
view all    included  .1                                80
access MyRWGroup ""      any       noauth    exact  all    all    none
syslocation Unknown (edit /etc/snmp/snmpd.conf)
syscontact Root  (configure /etc/snmp/snmp.local.conf)
master agentx
agentAddress udp:161,udp6:161

trapcommunity  cisco123
agentSecName meme
rouser meme


 # Send all traps upstream - Don't change this password or it breaks the framework.
 # v1 and v2 traps _could_ be sent for all but only need v2 trap.
 trap2sink lbvip02 cisco123


##########
#
# Local Stats
#
```

```
ignoreDisk /proc
ignoreDisk /proc/sys/fs/binfmt_misc
ignoreDisk /var/lib/nfs/rpc_pipefs
ignoreDisk /dev/shm
ignoreDisk /dev/pts
disk / 90%


swap 102400

load 6 6 6
#linkUpDownNotifications yes

notificationEvent  linkUpTrap    linkUp   ifIndex ifAdminStatus ifOperStatus
notificationEvent  linkDownTrap  linkDown ifIndex ifAdminStatus ifOperStatus

monitor  -S -u meme -r 60 -e linkUpTrap -o ifDescr  "Generate linkUp" ifOperStatus != 2
monitor  -u meme -r 60 -e linkDownTrap -o ifDescr "Generate linkDown" ifOperStatus == 2

# Note: alert!=0, clear==0 and messages must be unique or snmpd errors.
monitor -u meme -r 60 -o dskPath -o dskErrorMsg "DiskFullAlert" dskErrorFlag != 0
monitor -S -u meme -r 60 -o dskPath -o dskErrorMsg "DiskFullClear" dskErrorFlag == 0
monitor -u meme -r 60 -o memErrorName -o memSwapErrorMsg "LowSwapAlert" memSwapError !=
 0
monitor -S -u meme -r 60 -o memErrorName -o memSwapErrorMsg "LowSwapClear" memSwapError
 == 0
monitor -u meme -r 60 -o laNames -o laErrMessage  "HighLoadAlert" laErrorFlag != 0
monitor -S -u meme -r 60 -o laNames -o laErrMessage  "HighLoadClear" laErrorFlag == 0


##########
#
# BROADHOP-QNS-MIB Proxy Configuration
#
##########
# proxy -v <version> -c <community> <local_host> <map_to> <map_from>
#
# NOTE:  Most values are listed twice.  This is to cover the snmp get requirement
#        for scalar values.  Snmp get for scalar values (ie. not a table) is
#        required to return for both x.y OID and .x.y.0 OID values.  This only
#        effects <map_to> values.

##########
#
# System Stats
#


#
# User, System and Idle CPU (UCD-SNMP-MIB ss)

proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.1.0
.1.3.6.1.4.1.2021.11.9.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.2.0
.1.3.6.1.4.1.2021.11.10.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.3.0
.1.3.6.1.4.1.2021.11.11.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.1
.1.3.6.1.4.1.2021.11.9.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.2
.1.3.6.1.4.1.2021.11.10.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.3
.1.3.6.1.4.1.2021.11.11.0
# 1, 5 and 15 Minute Load Averages (UCD-SNMP-MIB la)
```

```
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.4
.1.3.6.1.4.1.2021.10.1.5.1
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.5
.1.3.6.1.4.1.2021.10.1.5.2
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.6
.1.3.6.1.4.1.2021.10.1.5.3
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.4.0
.1.3.6.1.4.1.2021.10.1.5.1
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.5.0
.1.3.6.1.4.1.2021.10.1.5.2
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.6.0
.1.3.6.1.4.1.2021.10.1.5.3
# Memory Total, Memory Available, Swap Total, Swap Available (UCD-SNMP-MIB mem)
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.7
.1.3.6.1.4.1.2021.4.5.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.8
.1.3.6.1.4.1.2021.4.6.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.9
.1.3.6.1.4.1.2021.4.3.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.10
.1.3.6.1.4.1.2021.4.4.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.7.0
.1.3.6.1.4.1.2021.4.5.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.8.0
.1.3.6.1.4.1.2021.4.6.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.9.0
.1.3.6.1.4.1.2021.4.3.0
proxy -v 2c -c cisco_ro localhost .1.3.6.1.4.1.26878.200.3.2.70.1.10.0
.1.3.6.1.4.1.2021.4.4.0
```

4. After updating the `snmpd.conf` file, execute the following commands from Cluster Manager.

```
mkdir /etc/snmp/mibs;scp root@qns01:/etc/snmp/mibs/* /etc/snmp/mibs
scp root@qns01:/etc/sysconfig/snmpd /etc/sysconfig/snmpd
scp root@qns01:/etc/logrotate.d/snmpd /etc/logrotate.d/snmpd
scp root@qns01:/etc/monit.d/snmpd /etc/monit.d/
service monit restart
```

For SNMPv3

1. Add the following content to `/etc/snmp/snmpd.conf` file.

```
rouser cisco_snmpv3
rouser cisco_snmpv3_trap
com2sec local localhost cisco_snmpv3
group MyRWGroup usm local
group MyRWGroup usm cisco_snmpv3
view all included .1 80
access MyRWGroup "" any noauth exact all all none
syslocation Unknown (edit /etc/snmp/snmpd.conf)
syscontact Root (configure /etc/snmp/snmp.local.conf)
master agentx
agentSecName cisco_snmpv3_trap
trapsess -v 3 -u cisco_snmpv3_trap -a SHA -m 0xf8798c43bd2f058a14ffde26f037fbc5d44f434e
 -x AES -m
0xf8798c43bd2f058a14ffde26f037fbc5d44f434e -l authPriv lbvip02
##########
#
# Local Stats
#
ignoreDisk /proc
ignoreDisk /proc/sys/fs/binfmt_misc
ignoreDisk /var/lib/nfs/rpc_pipefs
ignoreDisk /dev/shm
```

```
ignoreDisk /dev/pts
disk / 10%
disk /var 10%
disk /boot 10%
swap 102400
#load = 1.5 * vCPUs (allocated to VM)
load 9 9 9
#linkUpDownNotifications yes
notificationEvent linkUpTrap linkUp ifIndex ifAdminStatus ifOperStatus
notificationEvent linkDownTrap linkDown ifIndex ifAdminStatus ifOperStatus
monitor -S -u cisco_snmpv3_trap -r 60 -e linkUpTrap -o ifDescr "Generate linkUp"
ifOperStatus !=
2
monitor -u cisco_snmpv3_trap -r 60 -e linkDownTrap -o ifDescr "Generate linkDown"
ifOperStatus ==
2
# Note: alert!=0, clear==0 and messages must be unique or snmpd errors.
monitor -u cisco_snmpv3_trap -r 60 -o dskPath -o dskErrorMsg "DiskFullAlert" dskErrorFlag
 != 0
monitor -S -u cisco_snmpv3_trap -r 60 -o dskPath -o dskErrorMsg "DiskFullClear"
dskErrorFlag == 0
monitor -u cisco_snmpv3_trap -r 60 -o memErrorName -o memSwapErrorMsg "LowSwapAlert"
memSwapError
!= 0
monitor -S -u cisco_snmpv3_trap -r 60 -o memErrorName -o memSwapErrorMsg "LowSwapClear"
memSwapError == 0
monitor -u cisco_snmpv3_trap -r 60 -o laNames -o laErrMessage "HighLoadAlert" laErrorFlag
 != 0
monitor -S -u cisco_snmpv3_trap -r 60 -o laNames -o laErrMessage "HighLoadClear"
laErrorFlag == 0
monitor -u cisco_snmpv3_trap -r 60 -o memAvailReal -o memTotalReal "LowMemoryAlert"
memAvailReal<
1633390
monitor -S -u cisco_snmpv3_trap -r 60 -o memAvailReal -o memTotalReal "LowMemoryClear"
memAvailReal
>= 1633390
##########
#
# System Stats
#
# User, System and Idle CPU (UCD-SNMP-MIB ss)
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.1.0 .1.3.6.1.4.1.2021.11.9.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.2.0 .1.3.6.1.4.1.2021.11.10.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.3.0 .1.3.6.1.4.1.2021.11.11.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.1 .1.3.6.1.4.1.2021.11.9.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.2 .1.3.6.1.4.1.2021.11.10.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
```

```
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.3 .1.3.6.1.4.1.2021.11.11.0
# 1, 5 and 15 Minute Load Averages (UCD-SNMP-MIB la)
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.4 .1.3.6.1.4.1.2021.10.1.5.1
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.5 .1.3.6.1.4.1.2021.10.1.5.2
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.6 .1.3.6.1.4.1.2021.10.1.5.3
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.4.0 .1.3.6.1.4.1.2021.10.1.5.1
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.5.0 .1.3.6.1.4.1.2021.10.1.5.2
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.6.0 .1.3.6.1.4.1.2021.10.1.5.3
# Memory Total, Memory Available, Swap Total, Swap Available (UCD-SNMP-MIB mem)
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.7 .1.3.6.1.4.1.2021.4.5.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.8 .1.3.6.1.4.1.2021.4.6.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.9 .1.3.6.1.4.1.2021.4.3.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.10 .1.3.6.1.4.1.2021.4.4.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.7.0 .1.3.6.1.4.1.2021.4.5.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.8.0 .1.3.6.1.4.1.2021.4.6.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.9.0 .1.3.6.1.4.1.2021.4.3.0
proxy -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -m
0x7a64eefbf13e918c77b41fada0b55cf8338d6cc8 -x AES -m 0x7a64eefbf13e918c77b41fada0b55cf8
 -l authPriv
localhost .1.3.6.1.4.1.26878.200.3.2.70.1.10.0 .1.3.6.1.4.1.2021.4.4.0
```

Note For snmptrap, puppet executes the script `/var/broadhop/initialize_snmpv3_trap.sh`. The script `/var/broadhop/initialize_snmpv3_trap.sh` is starting and stopping snmptrapd twice.

```
[root@lb01 broadhop]# ./initialize_snmpv3_trap.sh
Stopping monit: [ OK ]
Stopping snmpd: [ OK ]
Stopping snmptrapd: [ OK ]
Starting snmptrapd: [ OK ]
Stopping snmptrapd: [ OK ]
Starting snmptrapd: [ OK ]
Starting snmpd: [ OK ]
Starting monit: Starting Monit 5.17.1 daemon with http interface at [localhost]:2812
[ OK ]
[root@lb01 broadhop]#
```

2. Replace the string in <*tag*> with the actual value. You can check the `snmpd.conf` from other VMs to get the values for tags. For example, `/etc/snmp/snmpd.conf` file on lb01.
3. You can also update the configuration parameter such as `load 6 6 6` to some other value based on number of vCPUs present on Cluster Manager.

Note Formula is 1.5 * no_of_vCPUs. Consider only the integer value from the output.

Here is an sample `snmpd.conf` file configuration:

4. After updating the snmpd.conf file, execute the following commands from Cluster Manager.

```
mkdir /etc/snmp/mibs;scp root@qns01:/etc/snmp/mibs/* /etc/snmp/mibs
scp root@qns01:/etc/sysconfig/snmpd /etc/sysconfig/snmpd
scp root@qns01:/etc/logrotate.d/snmpd /etc/logrotate.d/snmpd
scp root@qns01:/etc/monit.d/snmpd /etc/monit.d/
service monit restart
```

Validation and Testing

This section describes the commands for validation and testing of the CPS SNMP infrastructure. You can use these commands to validate and test your system during setting up or configuring the system. Our examples use MIB values because they are more descriptive but you may use equivalent OID values if you like particularly when configuring an NMS.

The examples here use Net-SNMP snmpget snmpwalk and snmptrap programs. Detailed configuration of this application is outside the scope of this document but the examples assume that the three Cisco MIBs are installed in the locations described on the man page of snmpcmd (typically the `/etc/snmp/mibs` directories).

Run all tests from a client with network access to the Management Network or from lb01 lb02 (which are also on the Management Network).

Component Statistics

Component statistics can be obtained on a per statistic basis with snmpget. For example, to get the current available memory on pcrfclient01, use the following commands:

For SNMPv2

```
snmpget -v 2c -c broadhop -M /etc/snmp/mibs:/usr/share/snmp/mibs -m
+BROADHOP-MIB:CISCO-QNS-MIB
pcrfclient01 .1.3.6.1.4.1.26878.200.3.2.70.1.8
```

An example of the output from this command is:

```
CISCO-QNS-MIB::componentMemoryAvailable = INTEGER: 4551356
```

Interpreting this output means that 4551356 MB of memory are available on this component machine.

All available component statistics in an MIB node can be "walked" via the snmpwalk command. This is very similar to snmpget as above. For example, to see all statistics on lb01 use the command:

```
snmpwalk -v 2c -c broadhop -M /etc/snmp/mibs:/usr/share/snmp/mibs -m
+BROADHOP-MIB:CISCO-QNS-MIB
lb01 .1.3.6.1.4.1.26878.200.3.2.70
```

An example of the output from this command is:

```
CISCO-QNS-MIB::componentCpuUser = INTEGER: 34
CISCO-QNS-MIB::componentCpuUser.0 = INTEGER: 34
CISCO-QNS-MIB::componentCpuSystem = INTEGER: 3
CISCO-QNS-MIB::componentCpuSystem.0 = INTEGER: 3
CISCO-QNS-MIB::componentCpuIdle = INTEGER: 61
CISCO-QNS-MIB::componentCpuIdle.0 = INTEGER: 61
CISCO-QNS-MIB::componentLoadAverage1 = INTEGER: 102
CISCO-QNS-MIB::componentLoadAverage1.0 = INTEGER: 102
CISCO-QNS-MIB::componentLoadAverage5 = INTEGER: 101
CISCO-QNS-MIB::componentLoadAverage5.0 = INTEGER: 101
CISCO-QNS-MIB::componentLoadAverage15 = INTEGER: 109
CISCO-QNS-MIB::componentLoadAverage15.0 = INTEGER: 109
CISCO-QNS-MIB::componentMemoryTotal = INTEGER: 12198308
CISCO-QNS-MIB::componentMemoryTotal.0 = INTEGER: 12198308
CISCO-QNS-MIB::componentMemoryAvailable = INTEGER: 4518292
CISCO-QNS-MIB::componentMemoryAvailable.0 = INTEGER: 4518292
CISCO-QNS-MIB::componentSwapTotal = INTEGER: 0
CISCO-QNS-MIB::componentSwapTotal.0 = INTEGER: 0
CISCO-QNS-MIB::componentSwapAvailable = INTEGER: 0
CISCO-QNS-MIB::componentSwapAvailable.0 = INTEGER: 0
```

For SNMPv3

```
snmpwalk -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -A Cisco-12345 -x AES -X
Cisco-12345 -l
authPriv -M /etc/snmp/mibs:/usr/share/snmp/mibs -m +BROADHOP-MIB:CISCO-QNS-MIB pcrfclient01
.1.3.6.1.4.1.26878.200.3.2.70.1
snmpget -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -A Cisco-12345 -x AES -X Cisco-12345
 -l
authPriv -M /etc/snmp/mibs:/usr/share/snmp/mibs -m +BROADHOP-MIB:CISCO-QNS-MIB pcrfclient01
.1.3.6.1.4.1.26878.200.3.2.70.1.2.0
```

Application KPI

Application KPI can be obtained on a per statistic basis with snmpget in a manner much like obtaining Component Statistics. As an example to get the aggregate number of sessions currently active on qns01 use the following commands:

For SNMPv2

```
snmpget -v 2c -c broadhop -M /etc/snmp/mibs:/usr/share/mibs -m +BROADHOP-MIB:CISCO-QNS-MIB
 qns01
.1.3.6.1.4.1.26878.200.3.3.70.15.24
```

An example of the output from this command would be:

```
iso.3.6.1.4.1.26878.200.3.3.70.15.24 = STRING: "0"
```

Interpreting this output means that 0 sessions are active on qns01.

Similarly, all available KPI in an MIB node can be "walked" via the snmpwalk command. This is very similar to snmpget as above. As an example, to see all statistics on qns01, use the following command:

```
snmpwalk -v 2c -c broadhop -M /etc/snmp/mibs:/usr/share/mibs -m +BROADHOP-MIB:CISCO-QNS-MIB
 qns01
.1.3.6.1.4.1.26878.200.3.3.70.15
```

An example of the output from this command would be:

```
iso.3.6.1.4.1.26878.200.3.3.70.15.20 = STRING: "0"
iso.3.6.1.4.1.26878.200.3.3.70.15.20.0 = STRING: "0"
iso.3.6.1.4.1.26878.200.3.3.70.15.21 = STRING: "0"
iso.3.6.1.4.1.26878.200.3.3.70.15.21.0 = STRING: "0"
iso.3.6.1.4.1.26878.200.3.3.70.15.22 = STRING: "0"
iso.3.6.1.4.1.26878.200.3.3.70.15.22.0 = STRING: "0"
iso.3.6.1.4.1.26878.200.3.3.70.15.23 = STRING: "0"
iso.3.6.1.4.1.26878.200.3.3.70.15.23.0 = STRING: "0"
iso.3.6.1.4.1.26878.200.3.3.70.15.24 = STRING: "0"
iso.3.6.1.4.1.26878.200.3.3.70.15.24.0 = STRING: "0"
iso.3.6.1.4.1.26878.200.3.3.70.15.25 = STRING: "1434914488"
iso.3.6.1.4.1.26878.200.3.3.70.15.25.0 = STRING: "1434914488"
```

For SNMPv3

```
snmpwalk -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -A Cisco-12345 -x AES -X
Cisco-12345 -l
authPriv -M /etc/snmp/mibs:/usr/share/mibs -m +BROADHOP-MIB:CISCO-QNS-MIB qns01
.1.3.6.1.4.1.26878.200.3.3.70
snmpget -e 0x0102030405060708 -v 3 -u cisco_snmpv3 -a SHA -A Cisco-12345 -x AES -X Cisco-12345
 -l
authPriv -M /etc/snmp/mibs:/usr/share/mibs -m +BROADHOP-MIB:CISCO-QNS-MIB qns01
.1.3.6.1.4.1.26878.200.3.3.70.15.25.0
```

Alarm Notifications/Traps

Testing and validating alarms notifications requires slightly more skill than testing SNMP gets and walks. Recall that the overall architecture is that all components and applications in the CPS system are configured to send notifications to lb01 or lb02 via lbvip02, the Internal Network IP.

These systems log the notification locally in `/var/log/snmp/trap` and then "re-throw" the notification to the destination configured by corporate_nms_ip. Two testing and troubleshooting methods can be performed: confirming notifications are being sent properly from system components to lb01 or lb02, and confirming that notifications can be sent upstream to the NMS.

Testing Individual Traps

Chapter 1 in the *CPS Troubleshooting Guide* includes procedures to test each CPS trap individually.

Troubleshooting

Note For information about troubleshooting SNMP notifications and traps, refer to *Cisco Policy Suite Troubleshooting Guide*.

The scenarios mentioned in the following sections are applicable only for Application notifications.

Here are some scenarios:

Why the clear notifications come from different sources that the alert notification sent out from?

A: In case of alarms, CPS supports high availability by running the monitoring scripts on both the pcrfclient01 and pcrfclient02. To illustrate this point here is a sample output from pcrfclient01 and pcrfclient02.

pcrfclient01:

```
[root@pcrfclient01 ~]# monsum
The Monit daemon 5.17.1 uptime: 6h 6m

Process 'whisper'                   Running
Process 'snmpd'                     Running
Program 'kpi_trap'                  Status ok
Program 'db_trap'                   Status ok
Program 'failover_trap'             Status ok
Program 'qps_process_trap'          Status ok
Program 'admin_login_trap'          Status ok
Program 'vm_trap'                   Status ok
Program 'qps_message_trap'          Status ok
Program 'ldap_message_trap'         Status ok
```

pcrfclient02:

```
[root@pcrfclient02 ~]# monsum
The Monit daemon 5.17.1 uptime: 5h 47m

Process 'whisper'                   Running
Process 'snmpd'                     Running
Program 'kpi_trap'                  Status ok
Program 'db_trap'                   Status ok
Program 'failover_trap'             Status ok
Program 'qps_process_trap'          Status ok
Program 'admin_login_trap'          Status ok
Program 'vm_trap'                   Status ok
Program 'qps_message_trap'          Status ok
Program 'ldap_message_trap'         Status ok
```

- The monitoring scripts are responsible for detecting conditions that can lead to raising or clearing a trap.
- Once a condition that can lead to an alarm is detected by both the pcrfclients, both pcrfclient01 & pcrfclient02 individually raise an event towards HA-Proxy.
- The HA-Proxy forwards both the events to the Fault Management System(**FMS**).
- The FMS raises a trap for the first event it receives and discards the second event.
- When pcrfclient01 raises an alert, it is because the event sent by pcrfclient01 reaches the FMS first. Event sent by pcrfclient02 is ignored by FMS.

- When pcrfclient02 clears an alarm, it is because the corresponding event sent by the pcrfclient02 reaches the FMS first. Event sent by pcrfclient01 is ignored by FMS.

How to match alarm and clear for the same event, from different sources?

A: Every Alarms/Clear generated from CPS system has the following varbinds:

- **broadhopComponentName:** The broadhopComponentName object is used to provide the name of the individual system device being trapped.
- **broadhopComponentTime:** The broadhopComponentTime object is used to provide the date and time associated with the occurrence of the problem being trapped.
- **broadhopComponentNotificationName:** The broadhopComponentNotificationName object is used to provide the name of the notification.
- **broadhopNotificationFacility:** This object determines the facility or layer which notifications are sourced.
- **broadhopNotificationSeverity:** This object determines the severity or level of sourced notifications.
- **broadhopComponentAdditionalInfo:** This object is used to provide any additional information about the problem being trapped.

To match the alarm and clear from different host, user can use the following field information:

- broadhopComponentNotificationName
- broadhopNotificationSeverity
- broadhopComponentAdditionalInfo

Note Ignore the text before the first colon (:) from the additional info field.

Host Independent Alarms: Alarm and clear can come from different host.

- All DB Member of replica set Up
- All DB Member of replica set Down
- Primary DB Member Found
- No Primary DB Member Found
- VirtualInterface Up (External VIPs)
- VirtualInterface Down
- VirtualInterfaceDown (Internal VIPs)
- VirtualInterfaceUp
- License Usage Threshold Exceeded
- LicensedSessionCreation
- InvalidLicense

- PolicyConfiguration
- PoliciesNotConfigured
- DiameterAllPeersDown
- ZeroMQConnectionError
- DeveloperMode

How to match if alarm and clears coming from same source?

A: To match the alarm and clear from same host, user can use the following field information:

- broadhopComponentNotificationName
- broadhopNotificationSeverity
- broadhopComponentAdditionalInfo
- broadhopComponentName

Host Dependent Alarms: Alarm and clear come from the same host.

- DB Member Up
- DB Member Down
- Arbiter Up
- Arbiter Down
- Config Server Up
- Config Server Down
- DB Resync is not needed
- DB Resync is needed
- QNS Process Up
- QNS Process Down
- VM Up
- VM Down
- Site Up
- Site Down
- LDAPAllPeersDown
- LDAPPeerDown
- Percentage of LDAP retry threshold Exceeded
- Percentage of LDAP retry threshold Normal
- LDAP Requests as percentage of CCR-I Dropped

- LDAP Requests as percentage of CCR-I Normal
- LDAP Requests Dropped
- LDAP Requests Normal
- LDAP Query Result Dropped
- LDAP Query Result Normal
- Gx Message processing Dropped
- Gx Message processing Normal
- Gx Average Message processing Dropped
- Gx Average Message processing Normal
- All SMSC server connections are down
- Atleast one SMSC server connection is up
- SMSC server connection down
- SMSC server connection up
- All Email servers not reachable
- At least one Email server is reachable
- Email server is not reachable
- Email server is reachable
- MemcachedConnectError
- ApplicationStartError
- DiameterPeerDown
- DiameterStackNotStarted

Information Alarm (Alarms without clear indication)

There are no clear trap for the following alarms:

- HA Failover
- GR Failover
- Admin User Logged in