Logging

Overview

CPS logs can be divided into two types:

- Application Logs – generated by CPS applications
- VM Logs – generated by the underlying virtual machine operating system

The normal logs on the individual policy server/policy director/OAM (pcrfclient) VMs are:

Table 1: Normal Logs

File	Contains	Useful for
`/var/log/broadhop/qns-1.log`	main detailed policy server (qns) application logs.	finding initialization errors and application level errors.
`/var/log/broadhop/qns-engine-1.log`	detailed event logs.	finding which services a subscriber has, the state of a session, and other detailed information.
`/var/log/broadhop/service-qns-1.log`	the startup logs. If `logback.xml` is incorrectly formatted, all other log statements will go into this log.	startup errors.

Policy Server (QNS) writes policy director (iomgr) and policy server (qns) logs to consolidated logs on pcrfclient01 including:

Table 2: policy director (iomgr) and policy server (qns) logs

File	Contains	Useful for
/var/log/broadhop/consolidated-qns.log	the consolidation of all policy server (qns) logs with the IP of the instance as part of the log event.	finding initialization errors and application level errors.
/var/log/broadhop/consolidated-engine.log	the consolidation of all policy server (qns) engine logs with the IP of the instance as part of the log event.	finding which services a subscriber has, the state of a session, and other detailed information.

Each VM stores their log files locally before they are consolidated on pcrfclient01. The local logs are:

```
/var/log/broadhop/qns-<#>.log
/var/log/broadhop/service-qns-<#>.log
```

CPS Logs

The pcrfclient01 VM also contains the consolidated logs from all of the policy director (LB), policy server (QNS) and OAM (PCRFCLIENT) VMs.

The CPS logs can be divided based on Application/Script that produces the logs:

Application/Script Produces Logs: Deploy Logs

- **Log:** deploy log
  - **Description:** Log messages generated during CPS deployment.
  - **Log file name, format, path:**

    HA/GR: cluman: `/var/log/install_console_YYYYMMDD_HHMMSS.log`
  - **Log config File:** NA
  - **Log Rollover:** No

Application/Script Produces Logs: policy server

- **Log:** policy server (qns) log
  - **Description:** Main and most detailed logging. Contains initialization errors and application level errors.
  - **Log file name, format, path:**

    HA/GR: VM: `/var/log/broadhop/qns-<instance no.log`
  - **Log config File:** `/etc/broadhop/logback.xml`
  - **Log Rollover:** No

- **Log:** policy server (qns) service logs
  - **Description:** Contains start up logs. If `/etc/broadhop/logback.xml` is incorrectly formatted, all logging statements go into this log.
  - **Log file name, format, path:**

    HA/GR: qns0*: `/var/log/broadhop/service-qns-<instance no>.log`
  - **Log config File:** `/etc/broadhop/logback.xml`
  - **Log Rollover:** No
- **Log:** consolidated policy server (qns) logs
  - **Description:** Contains the consolidation of all policy server (qns) logs with the IP of the instance as part of the log event.
  - **Log file name, format, path:**

    HA/GR: pcrfclient0*: `/var/log/broadhop/consolidated-qns.log`
  - **Log config File:** `/etc/broadhop/controlcenter/logback.xml`
  - **Log Rollover:** No
- **Log:** consolidated engine logs
  - **Description:** Contains the consolidation of all policy server (qns) engine logs with the IP of the instance as part of the log event.
  - **Log file name, format, path:**

    HA/GR: `/var/log/broadhop/consolidated-engine.log`
  - **Log config File:** `/etc/broadhop/controlcenter/logback.xml`
  - **Log Rollover:** No
- **Log:** consolidated diagnostics logs
  - **Description:** Contains logs about errors occurred during diagnostics of CPS.
  - **Log file name, format, path:**

    HA/GR: pcrfclient0*: `/var/log/broadhop/consolidated-diag.log`
  - **Log config File:** `/etc/broadhop/controlcenter/logback.xml`
  - **Log Rollover:** No

Application/Script Produces Logs: policy server pb

- **Log:** policy server (qns) pb logs
  - **Description:** Policy Builder startup, initialization, warnings, and errors get logged into this log file.
  - **Log file name, format, path:**

HA/GR: pcrfclient0*: `/var/log/broadhop/qns-pb.log`

- **Log config File:** `/etc/broadhop/logback.xml`
- **Log Rollover:** No

- **Log:** service policy server (qns) pb logs
  - **Description:** Policy Builder service logs.
  - **Log file name, format, path:**

    HA/GR: pcrfclient0*: `/var/log/broadhop/service-qns-pb.log`
  - **Log config File:** `/etc/broadhop/logback.xml`
  - **Log Rollover:** No

Application/Script Produces Logs: mongo

- **Log:** MongoDB logs
  - **Description:** Contains useful information about the MongoDB operations including queries, errors, warnings, and users' behavior.
  - **Log file name, format, path:**

    HA/GR: sessionmgr01: `/var/log/mongodb-<port>.log`
  - **Log config File:** `/etc/init.d/sessionmgr-*` (the log options are hard coded into these startup scripts)
  - **Log Rollover:** No

Application/Script Produces Logs: httpd

- **Log:** httpd access logs
  - **Description:** Apache server records all incoming requests and all requests processed to a log file.
  - **Log file name, format, path:**

    HA/GR: pcrfclient0*: `/var/log/httpd/qns-default_access.log`
  - **Log config File:** `/etc/httpd/conf/httpd.conf`
  - **Log Rollover:** Yes

- **Log:** httpd error logs
  - **Description:** All apache errors/diagnostic information about other errors found during serving requests are logged to this file. This apache log file often contain details of what went wrong and how to fix it.
  - **Log file name, format, path:**

HA/GR: pcrfclient0*: `/var/log/httpd/error_log`

- **Log config File:** `/etc/httpd/conf/httpd.conf`
- **Log Rollover:** Yes

Application/Script Produces Logs: license manager

- **Log:** lmgrd logs
  - **Description:** Contains license file related errors.
  - **Log file name, format, path:**

    HA/GR: pcrfclient0*: `/var/log/broadhop/lmgrd.log`
  - **Log config File:** NA
  - **Log Rollover:** No

Application/Script Produces Logs: svn

- **Log:** SVN log
  - **Description:** Displays commit log messages. For more information refer: `/usr/bin/svn log -help`. For example:

    ```
    ./usr/bin/svn log http://lbvip02/repos/run
    ```
  - **Log file name, format, path:**

    HA/GR: NA
  - **Log config File:** NA
  - **Log Rollover:** No

Application/Script Produces Logs: auditd

- **Log:** audit logs
  - **Description:** Contains cron job logs and logs of all SSH sessions established to a CPS VM.
  - **Log file name, format, path:**

    HA/GR: VM: `/var/log/audit/audit.log`
  - **Log config File:** NA
  - **Log Rollover:** Yes

Application/Script Produces Logs: prometheus

- **Log:** prometheus logs
  - **Description:** Contains prometheus logs.
  - **Log file name, format, path:**

    HA/GR: pcrfclient0*: `/var/log/prometheus/prometheus.log` (Present only on pcrfclient VMs)
  - **Log Rollover:** Yes

Application/Script Produces Logs: collectd_exporter

- **Log:** collectd exporter logs
  - **Description:** Contains collectd exporter logs.
  - **Log file name, format, path:**

    HA/GR: pcrfclient0*: `/var/log/prometheus/collectd_exporter.log` (Present only on pcrfclient VMs)
  - **Log Rollover:** Yes

Application/Script Produces Logs: kernel

- **Log:** haproxy
  - **Description:** Contains information about HAProxy and VIP failovers.
  - **Log file name, format, path:**

    HA/GR: pcrfclient0*: `/var/log/messages`
  - **Log config File:** NA
  - **Log Rollover:** Yes

Policy Builder and Control Center Activity Logs

Policy Builder Logging

- Login and logout message in audit logs is now written into separate audit log for easy tracing.
  - File location: `/var/log/broadhop/qns-audit-pb.log`
  - Logs are available in pcrfclient VMs.
  - When the log file reach 20 MB, it gets rotated. Maximum of five latest log files are available at a specific time.

You need to enable the log information in `/etc/broadhop/logback-pb.xml` file.

```
============================================================
<!-- UI Activity Loggers -->
<logger name="com.broadhop.client.WorkspaceChooserDialog" level="info"><appender-ref
ref="UI-ACTIVITY" /></logger>
<logger name="com.broadhop.client.ui.framework.handlers.ExitHandler"
level="info"><appender-ref ref="UI-ACTIVITY" /></logger>
<!-- UI Activity Loggers -->
============================================================
```

- **Policy Builder publish logs in audit database:** User name is updated into the existing audit database entry.

  Enable Audit Database

  1. Enable the audit database logging by configuring the parameter in `/etc/broadhop/pb/pb.conf` file.

     ```
     -Dua.client.submit.audit=true
     ```

  2. Select the checkbox, **Log Read Request** in Policy Builder under **Systems** > ***system name*** > **Plugin Configuration** > **Audit Configuration**. For more information, see *Audit Configuration* sections in the *CPS Mobile Configuration Guide*.

Control Center Logging

- Login and logout message in audit logs is now written into separate audit log for easy tracing.
  - File location: `/var/log/broadhop/qns-audit-1.log`
  - Logs are available in Policy Server (QNS) VM.
  - When the log file reach 20 MB, it gets rotated. Maximum of five latest log files are available at a specific time.

  You need to enable the log information in `/etc/broadhop/logback-pb.xml` file.

  ```
  <!-- CC Login Logout -->
    <logger name="com.broadhop.ui.security.server.SessionConcurrencyManager"
  level="info"><appender-ref ref="UI-ACTIVITY" /></logger>
  <!-- CC Login Logout -->
  ```

- By default, Control Center activity logs are captured in Audit database.

Basic Troubleshooting Using CPS Logs

- Review the policy server (qns) engine logs on pcrfclient01/02:

  HA/GR: `/var/log/broadhop/consolidated-engine.log`

  These logs display issues or problems in the subscriber or services. If the event is not found in the engine logs, check the policy server (qns) logs to look for anomalies.

- Determine when the call was supposed to occur in order to narrow down the issue.
- grep usernames, MAC addresses, IP addresses, or other relevant data to find required information.

Logging Level and Effective Logging Level

Logging level and the actual effective logging level can be two different levels because of the following logback logging rules:

- When a logging level is set, if the logging level of the parent process is higher than the logging level of the child process, then the effective logging level of the child process is that of the parent process. That is, even though the logging level of the child process is set, it cannot be below the logging level of the parent process and is automatically overridden to the higher logging level of the parent process.
- There is a global "root" logging level that each process can inherit as an effective default logging level.
  - HA deployments default all logging to 'warn' level.
- Each logging level prints the output of the lower logging levels.

  The following table displays the logging level and the message types printed.

Table 3: Logging Level and Effective Logging Level

Level	Message Types Printed
All	Equivalent to Trace and some more messages.
Trace	Trace, Debug, Info, Warn, & Error
Debug	Debug, Info, Warn, & Error
Info	Info, Warn, & Error
Warn	Warn & Error
Error	Error
Off	-

The following table describes the different logging levels and what they should be used for:

Table 4: Logging Levels

Logging Level	Description	Valid Use Case	Invalid Use Case
Error	Error conditions that break a system feature. The error logging level should not be used for call flow errors.	Database is not available.	Subscriber not found.
Warn	Helps to understand the early signs that will prevent the system from functioning in the near future OR are triggered by unexpected preconditions in a method.	Retrieved more then one Gx QoS profile.	Warnings should not be used for individual call flows. No service found for session.

Logging Level	Description	Valid Use Case	Invalid Use Case
Info	Helps to understand the life cycle of components and subsystems, such as plug-ins and databases.	Troubleshooting low-level application issues.	Info should not be used for individual call flows.
Debug	Helps to understand the flow of the code execution at Class/Method level. i.e. in _createIsgDeviceSession({log...)	Troubleshooting low-level application issues.	NA
Trace	Helps to understand the values of the statement and branch of logics within the method for troubleshooting.	Troubleshooting low-level application issues.	NA

You can configure target and log rotation for consolidated logs in the control center's log configuration file `/etc/broadhop/controlcenter/logback.xml`.

The following parameters can be configured for target VM and port.

```
<appender name="SOCKET-BASE" class="ch.qos.logback.classic.net.SocketAppender>"
    <RemoteHost>${logging.controlcenter.host:-lbvip02}</RemoteHost>
    <Port>${logging.controlcenter.port:-5644}</Port>
    <ReconnectionDelay>10000</ReconnectionDelay>
    <IncludeCallerData>false</IncludeCallerData>
</appender>
```

The configuration above is used to redirect consolidated logs to lbvip02 VM on port 5644 with reconnection delay.

Consolidated log rotation is configured using the following configuration in `/etc/broadhop/controlcenter/logback.xml`.

```
<rollingPolicy
    class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">
    <fileNamePattern>
${com.broadhop.log.dir:-/var/log/broadhop}/consolidated-diag.%i.log.gz
    </fileNamePattern>
    <minIndex>1</minIndex>
    <maxIndex>5</maxIndex>
  </rollingPolicy>
<triggeringPolicy
    class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">
    <maxFileSize>100MB</maxFileSize>
  </triggeringPolicy>
```

Using the above configuration, 100 MB log files are generated and after that, log files rotate from index 1 to 5. This configuration will require 500 MB total available disk space.

Note Do not set `maxFileSize` greater than 100MB as this impacts performance in order to compress the log files.

Do not set `maxIndex` greater than 13, which is the limitation on the logging framework used by CPS.

When the 100 MB log file trigger condition is met, the order in which CPS system performs the file operations is:

- log.5.gz > deleted

- log.4.gz > log.5.gz
- log.2.gz > log.3.gz
- log.1.gz > log.2.gz
- Current > log.1.gz

Similar configurations can be applied for policy server (qns) logs in `/etc/broadhop/logback.xml`.

Consolidated Application Logging

Consolidated logging is a function of all of the CPS VMs, and sends CPS application logs to a central server (either pcrfclient01 or pcrfclient02) to aid the debugging process. The following procedure describes how to configure the consolidated logging function.

Step 1 Edit the `logback.xml` file that is present in the `/etc/broadhop` directory and the `logback.xml` file that is present in the `/etc/broadhop/controlcenter` directory.

Start by viewing the `/etc/broadhop/logback.xml` file. It must have a section that looks similar to this:

```
<!-- Configure Loggers -->
<!-- Hide 'Could not load class...' noise. -->
<logger
name="org.springframework.osgi.extensions.annotation.ServiceRefe
renceDependencyBeanFactoryPostProcessor" level="error" />
<logger name="org.springframework" level="warn" />
<logger name="com.broadhop.resource.impl" level="warn" />
<logger name="com.danga" level="warn" />
<logger name="httpclient.wire" level="warn" />
<logger name="org.apache.commons.httpclient" level="warn" />
<logger name="sun.rmi.tranrsport.tcp" level="warn" />
<logger name="org.apache.activemq.transport.InactivityMonitor" level="warn" />
<!-- Configure default Loggers -->
<root level="warn">
<appender-ref ref="FILE" />
<appender-ref ref="SOCKET" />
</root>
```

The level can be configured to error, warn, info, or debug in the order of least logging to most logging. When debugging an issue or during initial installation. We recommend that you set the logging level to debug. To change the logging level, change one of the levels or add additional categories, for which you must contact a Cisco support representative.

View the `/etc/broadhop/controlcenter/logback.xml` file. It must have a section that looks similar to this:

```
<!-- Configure Remote Logger -->
<logger name="remote" level="info" additivity="false">
<appender-ref ref="CONSOLIDATED-FILE" />
<appender-ref ref="CONSOLIDATED-JMX" />
</logger>
```

Step 2 If you do not want to have a default effective logging level, then set the root level to off, as shown:

```
<!-- Configure default Loggers -->
<root level="off">
<appender-ref ref="FILE" />
```

```
<appender-ref ref="SOCKET" />
</root>
```

In `/etc/logrotate.d`, the logrotation configuration files are present where the rotation time and size are defined.

If any of the logfile is not rotated within the defined time/size and file is increasing continuously, then perform the following steps to solve the issue:

1. Move the logfile to different a location for backup.
2. Restart the particular process to create the new file for rotation to work.

   For example, the log for particular mongo process does not work and size is increased very huge to approx 17 GB.

   ```
   [root@ARBITER02 log]# ls -lrth *27720*
   -rw-r--r-- 1 root root 3.5M Dec 10  2018 mongodb-27720.log.4.gz
   -rw-r--r-- 1 root root 180M Dec 11  2018 mongodb-27720.log.1
   -rw-r--r-- 1 root root  17G Sep 12 08:38 mongodb-27720.log
   [root@ARBITER02 log]#
   ```

3. To resolve this issue, move the mongo process file to another location for data backup and restart sessionmgr-27720 process to start the log rotation.

Enable Debug Logs

By default, Cisco recommends to keep log level as WARN or ERROR. Sometimes for analysis the user may need more detailed logging. For this, the user needs the log level based on Cisco recommendation on case-to-case basis.

The following are the various top-level loggers for which the user may need to change log level on case-to-case basis. These loggers must be defined in `/etc/broadhop/logback.xml` file.

To make sure that all changes are controlled from one VM, synchronize all changes made in the Cluster Manager to all the other VMs.

```
SSHUSER_PREFERROOT=true copytoall.sh <path of file where changes have been made> <path of file in other VMs where changes are to be reflected>
```

For example,

```
SSHUSER_PREFERROOT=true copytoall.sh /etc/broadhop/logback.xml /etc/broadhop/logback.xml
```

- For Diameter issues: com.broadhop.diameter2
- For CDR/EDR issues: com.broadhop.policyintel
- For Custom Reference Data issues: com.broadhop.custrefdata
- For Notifications issues: com.broadhop.notifications
- For Session Manager Cache issues: com.broadhop.policy.mdb.cache
- For Control Center issues: com.broadhop.controlcenter
- For Fault Management issues: com.broadhop.faultmanagement

- For LDAP issues: com.broadhop.ldap
- For SPR issues: com.broadhop.spr
- For Unified API issues: com.broadhop.unifiedapi
- For audit issues: com.broadhop.audit
- For policy related issues: com.broadhop.policy
- For any CPS logs issues for which the log level is not overridden by other loggers: com.broadhop
- For CER/CEA DWR/DWA stack level message debugging: jdiameter logs with org.jdiameter
- For PB API issues: com.broadhop.client.api, com.broadhop.client.publish, com.broadhop.client.api.publish.svnImpl, com.broadhop.client

Note For consolidated logs make sure that the configuration specified in Control Center is correct to forward logs to OAM (pcrfclient) VMs.

Note Do not set the root log level to anything higher than 'warn' in a production system. If needed, adjust the individual loggers listed in `logback.xml`.

The levels debug or info usually have logs rollover very quickly. After the log rolls over, the information is lost. For this reason, warn or error generates a substantially smaller amount of logging, and gives you the ability to look for issues in the system over a longer period of time.

Step 1 On the CPS node where you require debug logs, edit the `/etc/broadhop/logback.xml` file.

The default root logger level would be currently set to WARN. It must be changed to debug, as shown.

```
<!-- Configure default Loggers -->
<root level="debug">
<appender-ref ref="FILE" />
<appender-ref ref="SOCKET" />
</root>
```

Step 2 The specific component for which you require the debug log should be set to "debug" in the appropriate line. For example:

For Control Center:

On pcrfclient01, update the logback.xml on `/etc/broadhop/controlcenter/`.

```
<logger name="com.broadhop.controlcenter" level="debug"/>
And
<root level="debug">
    <appender-ref ref="FILE" />
  </root>
```

For Audit:

```
<logger name="com.broadhop.audit" level="debug"/>
```

For Balance:

```
<logger name="com.broadhop.balance" level="debug"/>
```

For SPR:

```
<logger name="com.broadhop.spr" level="debug"/>
```

For Congestion Reference Data:

```
<logger name="com.broadhop.CongestionRefData" level="debug"/>
```

For LDAP:

```
<logger name="com.broadhop.ldap" level="debug"/>
```

For DRA:

```
<logger name="com.broadhop.dra" level="debug"/>
```

For POP-3 Authentication:

```
<logger name="com.broadhop.pop3auth" level="debug"/>
```

For Scheduled Events:

```
<logger name="com.broadhop.scheduledevents" level="debug"/>
```

For Diameter:

```
<logger name="com.broadhop.diameter2" level="debug"/>
```

For CDR/EDR:

```
<logger name="com.broadhop.policyintel" level="debug"/>
```

For Custom Reference Data:

```
<logger name="com.broadhop.custrefdata" level="debug"/>
```

For Notification:

```
<logger name="com.broadhop.notifications" level="debug"/>
```

Session Manager Cache:

```
<logger name="com.broadhop.policy.mdb.cache" level="debug"/>
```

Step 3 Save and exit.

Step 4 Run the following command to synchronize changes to all CPS VMs:

```
/var/qps/bin/update/syncconfig.sh
```

Enable Unified API Request and Response Logging

The following procedure describes how to enable logging to debug Unified API requests and responses. This level of logging is usually sufficient for the majority of debugging.

Step 1 On the Cluster Manager VM, add the following entry to `/etc/broadhop/logback.xml`:

```
<logger name="com.broadhop.unifiedapi.soap.servlet" level="debug"/>
```

Step 2 Copy the updated `/etc/broadhop/logback.xml` file to all other CPS VMs:

```
/var/qps/install/current/scripts/bin/control/copytoall.sh /etc/broadhop/logback.xml
```

Step 3 Search the logs for the following phrases to locate valid API requests/responses:

```
request to server:
response from server:
```

The logs will include a string containing the XML sent on the request and response for Unified API calls. This XML will NOT contain the SOAP wrapper information, such as the namespace info and envelope, header, and body tags. It will only include the inner XML that policy server (QNS) actually processes.

The SOAP wrapper tags would need to be added to paste this into SoapUI and submit it. However, this is easily done by using SoapUI to create a sample request after reading the WSDL and then just pasting in the piece from the log in the appropriate place in the XML in SoapUI.

Note Set the following parameter in the `qns.conf` file to output the Unified API logs in formatted XML instead of a continuous string. You must restart the policy server (qns) processes after modifying `qns.conf` file.

```
-Dpretty.print.responses=true
```

Rsyslog Log Processing

Rsyslog Overview

Rsyslog logs Operating System (OS) data locally on each VM (`/var/log/messages`) using the `/etc/rsyslog.conf` and `/etc/rsyslog.d/*conf` configuration files.

rsyslog outputs all WARN level logs on CPS VMs to `/var/log/warn.log` file.

On all nodes, Rsyslog forwards the OS system log data to lbvip02 via UDP over the port defined in the logback_syslog_daemon_port variable as set in the CPS deployment template (Excel spreadsheet). To download the most current CPS Deployment Template (`/var/qps/install/current/scripts/deployer/templates/QPS_deployment_config_template.xlsm`), refer to the *CPS Installation Guide for VMware* or *CPS Release Notes* for this release.

Refer to http://www.rsyslog.com/doc/ for more details and Rsyslog documentation.

Rsyslog-proxy

A second instance of Rsyslog called Rsyslog-proxy is installed only on Policy Director (LB) nodes. Rsyslog-proxy is only installed if the syslog_managers_list variable is set in the CPS Deployment Template.

Rsyslog-proxy is the main log forwarding process and is configured in `/etc/rsyslog-proxy.conf` on LB01/LB02 VMs.

- It receives OS system log data from all the nodes via UDP over the PORT defined in the logback_syslog_daemon_port variable. The default port number is 6514.

- The `/etc/broadhop/controlcenter/logback.xml` file on OAM (pcrfclients) is configured to send logs to rsyslog-proxy via UDP using the logback SyslogAppender. See Configuration of Logback.xml, on page 17 for more information.
- Rsyslog-proxy forwards the OS system log data and CPS log data to logstash via TCP on PORT 6513 with a UDP backup.
- By default, Rsyslog-proxy does not log any syslog data to local files on the OAM (PCRFClient) VMs. To configure the system to output consolidated log files for syslog data on the OAM (PCRFclients), see Enable Consolidated Syslog Output to Files on OAM VMs, on page 16.
- It receives CPS JSON formatted log data via TCP on PORT 5544. Rsyslog-proxy forwards that to logstash via TCP on PORT 5543 with a UDP backup.
- It receives SNMP events via TCP on PORT 7546. rsyslog-proxy forwards that to logstash via TCP on PORT 7545 with a UDP backup.
- Rsyslog-proxy sends all OS system log data and CPS log data to any number of remote servers via UDP or TCP in case the encryption is enabled. (The remote servers must be configured to receive traffic but that is not a part of the scope of this document.)

Configuration for HA Environments

Configuration of Rsyslog for High Availability CPS environments is performed using the CPS Deployment Template.

Refer to the following information available in the template tabs.

Configuration Variables

The following variables can now be set in the CPS Deployment Template:

- syslog_managers_list — space separated list of remote logging servers (tuple protocol:hostname:port). Only UDP is currently supported.
- syslog_managers_ports — comma separated list of the remote logging server ports (must match the ports in the syslog_managers_list).
- logback_syslog_daemon_addr — hostname of the internal UDP server that rsyslog-proxy runs to receive incoming logs from CPS and OS (defaults to lbvip02).
- logback_syslog_daemon_port — incoming port for rsyslog-proxy (defaults to 6514).

Note If the syslog_managers_list variable is empty, the rsyslog-proxy instance is not installed or configured.

Additional Hosts Tab

The following parameter can be configured in the Additional Hosts tab of the CPS Deployment Template file:

Table 5: Parameters in Additional Hosts Tab

corporate_syslog_ip	syslog_manager	<IP ADDR>

Configuration Tab

The following parameters can be configured in the Configuration tab of the CPS Deployment Template file:

syslog_managers_list	udp:corporate_syslog_ip:<PORT>
syslog_managers_ports	<PORT>
logback_syslog_daemon_addr	lbvip02
logback_syslog_daemon_port	6514

- lbvip02 is the default address for logback to send data.
- 6514 is the default port for logback to send data.

Enable Consolidated Syslog Output to Files on OAM VMs

By default, consolidated syslog logs from all VMs are not written to local files on the OAM (PCRFClient) VMs. The following procedure describes how to configure the system to output consolidated log files for syslog data on the OAM (PCRFclients).

Step 1 On the Cluster Manager VM, edit the following file:

```
/etc/puppet/modules/qps/templates/logstash/logstash.conf
```

Step 2 Add the following section highlighted below:

```
output {
  if [type] == "snmp-event-log" or [type] == "qps" {
    udp {
      host => "127.0.0.1"
       port => 2121
    }
  }
  if [type] == "syslog" {
    file {
      message_format => "%{[message_remainder]}"
      codec => "plain"
      path => "/var/log/broadhop/syslog/consolidated-messages.log"
    }
  }
}
```

Step 3 The directory in the 'path' above must exist on pcrfclient01/pcrfclient02 VMs and the directory must be owned by 'logstash:logstash'. If needed, SSH to each OAM (pcrfclient) to create the directory. Use the following command to change ownership of this directory:

```
chown -R logstash:logstash <dir>
```

Step 4 Once the configuration is in place on the Cluster Manager VM, run the following command to prepare the VMs using this new configuration:

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

Step 5 Run the following command to propogate the changes to all VMs:

```
pupdate
```

Step 6 To control how often these log files are overwritten, edit the file `/etc/logrotate.d/logstash` on pcrfclient01/02 VMs with the following content.

Note The path and filename specified below should match the 'path' value in `/etc/puppet/modules/qps/templates/logstash/logstash.conf`.

```
/var/log/broadhop/syslog/*.log
/var/log/logstash/*.log
{
        daily
        rotate 7
        copytruncate
        compress
        delaycompress
        missingok
        notifempty
}
```

Configuration of Logback.xml

The `/etc/broadhop/controlcenter/logback.xml` file on OAM (pcrfclients) is configured to send logs to rsyslog-proxy via UDP using the logback SyslogAppender.

Refer to http://logback.qos.ch/manual/appenders.html#SyslogAppender for the Syslog Appender documentation.

The following appender forwards all CPS logs to a remote server.

```
<appender name='SYSLOG' class='ch.qos.logback.classic.net.SyslogAppender'>
  <syslogHost>lbvip02</syslogHost><!--#SAP#-->
  <port>6514</port><!--#SAP#-->
  <suffixPattern>[qps] [%d{yyyy-mm-dd'T'HH:mm:ss.SSSZ}] %msg</suffixPattern>
  <facility>LOCAL0</facility>
</appender>
```

Rsyslog Customization

CPS provides the ability to configure forwarding of consolidated syslogs from rsyslog-proxy on Policy Director VMs to remote syslog servers (refer to *CPS Installation Guide for VMware*). However, if additional customizations are made to rsyslog configuration to forward logs to external syslog servers in customer's network for monitoring purposes, such forwarding must be performed via dedicated action queues in rsyslog. In the absence of dedicated action queues, when rsyslog is unable to deliver a message to the remote server, its main message queue can fill up which can lead to severe issues, such as, preventing SSH logging, which in turn can prevent SSH access to the VM.

In the example below, rsyslog is configured to forward syslogs related to 'authpriv' onto a remote syslog server (for example, 10.10.10.1). The forwarding is done via a dedicated 'disk-assisted in-memory' action queue:

```
## Action queue for remote syslog forwarding
## The action queue config is specified above the
## directive to forward syslogs to remote server
$ActionQueueType LinkedList
$ActionQueueFileName remote
$ActionQueueSize 10000
$ActionQueueHighWatermark 8000
```

```
$ActionQueueLowWatermark 2500
$ActionQueueMaxDiskSpace 1G
$ActionQueueTimeoutEnqueue 0

authpriv.*;auth.info  @10.10.10.1
```

Refer to rsyslog documentation for further details on action queue configuration: http://www.rsyslog.com/doc/v5-stable/concepts/queues.html

Viewing Logs Without Superuser Privileges

TACACS+ users who do not have superuser privileges can access all the files on the systems and some of the files (sudosh logs) that contain sensitive data. Currently read-only/admin users can read the sudosh logs.

Only qns-ro and qns-admin users are allowed to view log files at specific paths according to their role and maintenance requirement. Access to logs are allowed only using the following paths:

- /var/log/
- /var/log/broadhop/scripts/
- /var/log/httpd
- /var/log/redis
- /var/log/broadhop

Commands such as `cat`, `less`, `more`, and `find` cannot be executed using sudo in CPS 10.0.0 or higher releases.

To read any file, execute the following script using sudo:

```
$ sudo /var/qps/bin/support/logReader.py -r h -n 2 -f /var/log/puppet.log
```

where,

- `-r`: Corresponds to tail (t) ,tailf (tf), and head (h) respectively
- `-n`: Determines number of lines to be read. It works with the -r option. This is an optional parameter.
- `-f`: Determines the complete file path to be read.

Note

- Non-root users cannot view the sudosh logs.
- Support to read gunzipped files is also available.