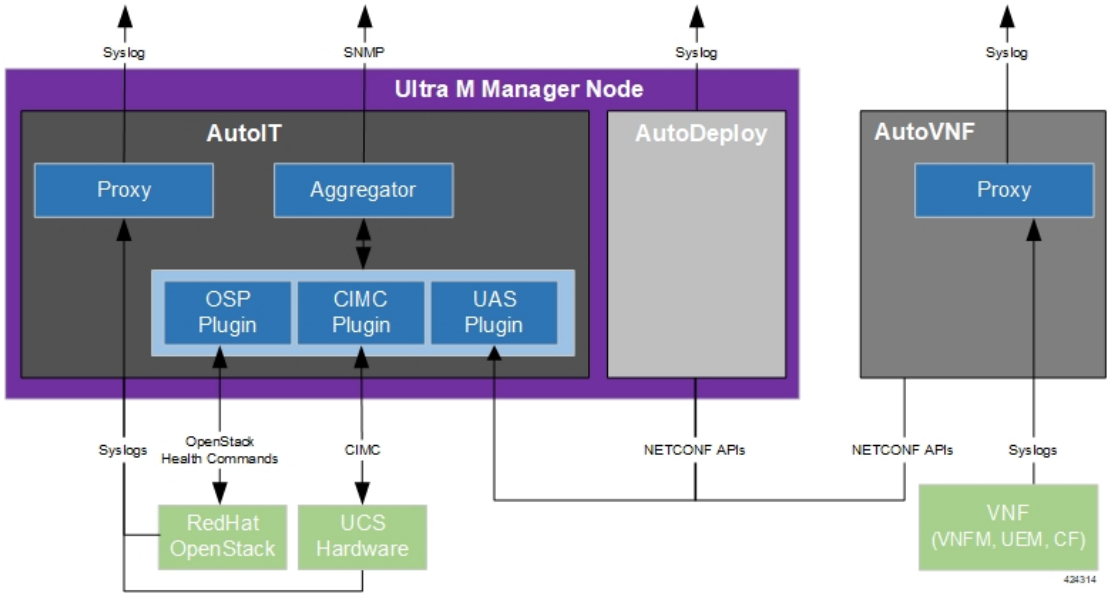
Health Monitoring in the Ultra M Solution

Hyper-Converged Ultra M solution models support a centralized health monitor and management function. This function provides a central aggregation point for events (faults and alarms) and a proxy point for syslogs generated by the different components within the solution as identified in Table 1: Component Event Source Domains, on page 11.

Figure 1: Ultra M Health Monitoring Functions



This functionality is installed with the UAS software modules.

Important The UAS-based health functionality is currently supported only with Ultra M UGP VNF deployments based on OSP 10 or OSP 13 and that leverage the Hyper-Converged architecture. The Ultra M Manager RPM is still distributed separately and is intended only for use in specific deployment scenarios. Contact your local sales or support representative for more information.

Once installed, additional configuration is required based on the desired functionality as described in the following sections:

- Syslog Proxy, on page 2

• Event Aggregation , on page 11
• Configuring Fault Suppression, on page 21

Syslog Proxy

Syslog proxy functionality is supported at the following levels:

- UCS server hardware
- OpenStack services
- UAS software modules
- VNFM, UEM, and CF VNF components

NOTES:

- This functionality is currently supported only with Ultra M UGP VNF deployments based on OSP 10 or OSP 13 and that leverage the Hyper-Converged architecture.
- You must configure a remote collection server to receive and filter log files sent by the Ultra M Manager Node.
  - Take note of the TCP and UDP ports configured on the server for syslogging as the syslog proxy functionality on Ultra M must be configured with the same ports.
  - Ensure that the collection server's IP table rules are configured to accept TCP/UDP connection on the configured port.
- Though you can configure syslogging at any severity level your deployment scenario requires, it is recommended that you only configure syslog levels with severity levels 0 (emergency) through 4 (warning). If the severity level is not set, then by default, the severity level 6 is used.

Important If you wish to enable syslogging for the components that comprise the Ultra M solution but do not wish to use the syslog proxy functionality (e.g. send syslogs directly to an external collection server), refer to Configuring Syslogging to an External Collection Server, on page 7.

Configuring Syslog Proxy for UCS Server Hardware

AutoIT can be configured to serve as a proxy for UCS server hardware syslogs.

Important AutoIT must be configured with information for the syslog collection server at the time it is deployed. Refer to the *Ultra Services Platform Deployment Automation Guide* for more information.

The UCS server list is based on the configuration specified in the VIM Orchestrator and VIM NSD configuration file. As such, syslog proxy functionality for the hardware must be performed after the VIM has been deployed.

Syslog proxy functionality is configured through a fault management descriptor (FMD) configuration file that is comprised of the required NETCONF parameters. Refer to Sample FMD Configuration File for a sample configuration file.

Important Though the FMD configuration can be included in the network service descriptor (NSD) for your VNF, it is recommended that the configuration for this functionality be maintained in a separate, FMD-specific NSD configuration file.

To configure syslog proxy functionality for UCS server hardware:

1. Log on to the primary AutoIT VM as the root user.
2. Prepare the FMD configuration file for your deployment scenario. Your file should include the following parameters:

   ```
   domain hardware
     syslog uas-proxy
     syslog severity <severity_level>
   ```

   Note that the **severity** parameter is optional. The default severity level is 6.

   For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

   Refer to Sample FMD Configuration File for a sample configuration file.
3. Login to the ConfD CLI as the *admin* user.

   ```
   confd_cli -u admin -C
   ```
4. Enter the *admin* user password when prompted.
5. Enter the ConfD configuration mode.

   ```
   config
   ```
6. Load the FMD configuration file.

   ```
   load merge <your_fmd_file_name>.cfg
   commit
   end
   ```
7. Activate the FMD configuration.

   ```
   activate nsd-id <nsd_name> fmd <fmd_name>
   ```

Important The output of this command is a transaction-id which can be used to monitor the deployment progress. If need be, the FMD configuration can be deactivated using the **deactivate** variant of this command.

8. Monitor the progress of the FMD creation by viewing transaction logs:

   ```
   show log <transaction_id> | display xml
   ```

   transaction_id is the ID displayed as a result of the **activate** command executed in step 7, on page 3.

Configuring Syslog Proxy for OpenStack Services

AutoIT can be configured to serve as a proxy for OpenStack service syslogs.

Important AutoIT must be configured with information for the syslog collection server at the time it is deployed. Refer to the *Ultra Services Platform Deployment Automation Guide* for more information.

The list of servers on which OpenStack is running is based on the configuration specified in the VIM Orchestrator and VIM NSD configuration file. As such, syslog proxy functionality for the hardware must be performed after the VIM has been deployed.

If syslogging is enabled, syslogs for the following OpenStack services are proxied:

- Nova
- Cinder
- Keystone
- Glance
- Ceph monitor (Controller nodes only)
- Ceph OSD (OSD Compute nodes only)

Syslog proxy functionality is configured through a fault management descriptor (FMD) configuration file that is comprised of the required NETCONF parameters. Refer to Sample FMD Configuration File for a sample configuration file.

Important Though the FMD configuration can be included in the network service descriptor (NSD) for your VNF, it is recommended that the configuration for this functionality be maintained in a separate, FMD-specific NSD configuration file.

To configure syslog proxy functionality for UCS server hardware:

1. Log on to the primary AutoIT VM as the root user.
2. Prepare the FMD configuration file for your deployment scenario. Your file should include the following parameters:

   ```
   domain vim
     syslog uas-proxy
     syslog severity <severity_level>
   ```

   Note that the **severity** parameter is optional. The default severity level is 6.

   For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

   Refer to Sample FMD Configuration File for a sample configuration file.
3. Login to the ConfD CLI as the *admin* user.

   ```
   confd_cli -u admin -C
   ```

4. Enter the *admin* user password when prompted.
5. Enter the ConfD configuration mode.

   config
6. Load the FMD configuration file.

   load merge *<your_fmd_file_name>*.**cfg**

   commit

   end
7. Activate the FMD configuration.

   activate nsd-id *<nsd_name>* **fmd** *<fmd_name>*

Important The output of this command is a transaction-id which can be used to monitor the deployment progress. If need be, the FMD configuration can be deactivated using the **deactivate** variant of this command.

8. Monitor the progress of the FMD creation by viewing transaction logs:

   show log *<transaction_id>* **| display xml**

   transaction_id is the ID displayed as a result of the **activate** command executed in step 7, on page 5.

Configuring Syslogging for UAS Software Modules

Each UAS software module can be configured to send logs and syslogs to one or more external collection servers.

AutoDeploy and AutoIT

Logs and syslogs are sent directly to one or more external syslog collection servers configured when these modules are first installed. The configured collection servers are also the receivers for UCS server hardware and OpenStack services for which AutoIT is a proxy.

The following logs are sent:

- **AutoDeploy:**
  - /var/log/upstart/autodeploy.log
  - /var/log/syslog
- **AutoIT:**
  - /var/log/upstart/autoit.log
  - /var/log/syslog

In order to support syslogging functionality, additional operators were added to the *boot_uas.py* script used to install these modules:

- **--syslog-ip**<*ext_syslog_server_address*>

- **--port**<*syslog_port_number*>
- **--severity**<*syslog_severity_to_send*>

Refer to the *Ultra Services Platform Deployment Automation Guide* for more information on deploying AutoIT and AutoDeploy.

AutoVNF

AutoVNF serves as the syslog proxy for the VNFM, UEM, and CF VNF components (VNFCs). It also sends its own logs to the same external syslog collection server:

- /var/log/upstart/autovnf.log
- /var/log/syslog

Syslogging for the AutoVNF module is configured through the AutoVNF VNFC configuration within the VNF Rack and VNF NSD configuration file. Refer to the *Ultra Services Platform Deployment Automation Guide* for more information.

AutoVNF should always configure the external syslog server. For AutoVNF, the information and instructions provided in those sources also remain identical but with the exception of the parameters used in the corresponding VNFC section of the VNF Rack and VNF NSD configuration file.

```
syslog server <ip_address>
syslog port <tcp_udp_port>
syslog severity <severity_level>
```

Note that the **port** and **severity** parameters are optional. The default values of **port** and **severity** parameters are 514 and 6 respectively.

For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

Configuring Syslog Proxy for the VNFM, UEM, and CF VNFCs

AutoVNF can be configured as the syslog proxy for the following VNFM, UEM, and CF VNF component (VNFC) logs:

- **VNFM (ESC):** /var/log/messages

Important *escmanager* and *mona* logs are not configured as part of syslog automation. ESC can be manually configured to send these logs to the syslog proxy or to an external syslog collection server. Refer to Manual ESC escmanager and mona Log Configuration, on page 9 for more information.

- **UEM:**
  - /var/log/em/vnfm-proxy/vnfm-proxy
  - /var/log/em/ncs/ncs-java-vm
  - /var/log/em/zookeeper/zookeeper

- /var/log/syslog

- **CF:** All syslogs configured within the StarOS-based VNF.

Syslogging for the VNFM, UEM, and CF is configured through their respective VNFC configurations within the VNF Rack and VNF NSD configuration file. Refer to the *Ultra Services Platform Deployment Automation Guide* for more information.

The following parameters should be configured for each VNFC:

```
syslog uas-proxy
syslog severity <severity_level>
```

Note that the **severity** parameter is optional. The default severity level is 6.

For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

Configuring Syslogging to an External Collection Server

Syslogging for the Ultra M solution components can be enabled without leveraging the syslog proxy functionality. In this scenario, syslogs are sent directly from each component to an external collection server.

Important Regardless of the domain level at which you're configuring syslogging functionality for, you must ensure that the external collection server to which your sending syslogs is reachable over the network by the component sending the syslog.

UCS Server Hardware

The instructions for configuring UCS servers to send syslogs to an external collection server are identical to those described in Configuring Syslog Proxy for UCS Server Hardware, on page 2 with the exception of the parameters used in the FMD configuration file.

To configure external collection servers for UCS server hardware, use the following parameters:

```
domain hardware
  syslog server <ip_address>
  syslog port <tcp_udp_port>
  syslog severity <severity_level>
```

Note that the **port** and **severity** parameters are optional. The default values of **port** and **severity** parameters are 514 and 6 respectively.

For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

Important Though multiple external collection servers can be configured, the UCS server hardware support a maximum of two servers. If more than two servers are configured in the FMD, only the first two are configured on the UCS servers. Additionally, only one severity level can be configured on the UCS servers. It is used for both configured collection servers.

OpenStack Services

The instructions for configuring OpenStack services to send syslogs to an external collection server are identical to those described in Configuring Syslog Proxy for OpenStack Services, on page 4 with the exception of the parameters used in the FMD configuration file.

To configure external collection servers for OpenStack services, use the following parameters:

```
domain vim
  syslog server <ip_address>
  syslog port <tcp_udp_port>
  syslog severity <severity_level>
```

Note that the **port** and **severity** parameters are optional. The default values of **port** and **severity** parameters are 514 and 6 respectively.

For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

UAS Software Modules

The information and instructions provided in Configuring Syslogging for UAS Software Modules, on page 5 and in the *Ultra Services Platform Deployment Automation Guide* that pertain to AutoDeploy and AutoIT configure them to communicate with external collection servers.

To configure external collection servers for the AutoVNF, use the following parameters:

```
  syslog server <ip_address>
  syslog port <tcp_udp_port>
  syslog severity <severity_level>
```

Note that the **port** and **severity** parameters are optional. The default values of **port** and **severity** parameters are 514 and 6 respectively.

For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

VNFM, UEM, and CF VNF Components

The instructions for configuring the VNFM, UEM, and CFs to send syslogs to an external collection server are identical to those described in Configuring Syslog Proxy for the VNFM, UEM, and CF VNFCs, on page 6 and in the *Ultra Services Platform Deployment Automation Guide* with the exception of the parameters used in the corresponding VNFC section of the VNF Rack and VNF NSD configuration file.

To configure external collection servers for the VNFCs, use the following parameters for each VNFC:

```
  syslog server <ip_address>
  syslog port <tcp_udp_port>
  syslog severity <severity_level>
```

Important

- Though multiple external collection servers can be configured, the ESC-based VNFM supports the configuration of only a single server. If multiple servers are configured in the VNFC, only the first is configured on the ESC-based VNFM. Additionally, all severity levels are enabled for the ESC-based VNFM regardless of the severity specified in the configuration.
- The ESC *escmanager* and *mona* logs are not configured as part of syslog automation. ESC can be manually configured to send these logs to the syslog proxy or to an external syslog collection server. Refer to Manual ESC escmanager and mona Log Configuration, on page 9 for more information.
- For the CF component within the VNF, neither the syslog port or the syslog severity need to be configured. The default syslog port of 514 and the default severity of 7, debug, is used.

  For more information on the syslog severity supported, refer to the *Cisco Ultra Services Platform NETCONF API Guide*.

Manual ESC *escmanager* and *mona* Log Configuration

ESC's *escmanager* and *mona* logs are not configured as part of syslog automation. However, ESC can be manually configured to send these logs to either the syslog proxy server (i.e. AutoVNF) or to an external collection server.

To manually configure ESC to send these logs:

1. Log on to the active ESC VNFM VM as the user *admin*.
2. Navigate to the */etc/rsyslog.d* directory.

   `cd /etc/rsyslog.d`
3. Create a configuration file for the *escmanager* log called *00-escmanager.conf*. The file should have the following configuration information which includes the IP address of the syslog server (either the syslog proxy server or the external collection server):

   ```
   $ModLoad imfile
   $InputFileName /var/log/esc/escmanager.log
   $InputFileTag escmanager:
   $InputFileStateFile stat-escmanager
   $InputRunFileMonitor

   $template escmanager_log, "%syslogtag:::% %msg%"

   if $programname == 'escmanager' then @@<syslog-server-ip>:<port-number>;escmanager_log

   if $programname == 'escmanager' then stop
   ```

   <*syslog-server-ip*> is the one of the following:

   - AutoVNF HA VIP address if you want the logs sent to the syslog proxy server, OR
   - IP address of the external syslog collection server.

   <*port-number*> is the TCP/UDP port used for syslog. For the syslog proxy functionality, the default port of 514 is used.

Important The server IP address and port number must be identical to those configured at the VNFC-level for the VNFM.

4. Create a configuration file for the *mona* log called *02-mona.conf*. The file should have the following configuration information which includes the IP address of the syslog server (either the syslog proxy server or the external collection server):

```
$ModLoad imfile
$InputFileName /var/log/esc/mona/mona.log
$InputFileTag mona:
$InputFileStateFile stat-mona
$InputRunFileMonitor

$template mona_log, "%syslogtag:::% %msg%"

if $programname == 'mona' then @@<syslog-server-ip>:<port-number>;mona_log

if $programname == 'mona' then stop
```

<*syslog-server-ip*> is the one of the following:

- AutoVNF HA VIP address if you want the logs sent to the syslog proxy server, OR
- IP address of the external syslog collection server.

<*port-number*> is the TCP/UDP port used for syslog. For the syslog proxy functionality, the default port of 514 is used.

Important The server IP address and port number must be identical to those configured at the VNFC-level for the VNFM.

5. Change the file permissions for the *escmanager.log* file.

```
ls -al /var/log/esc/escmanager.log
-rw-r--r--. 1 esc-user esc-user 12671993 Sep 12 23:32 /var/log/esc/escmanager.log
sudo chmod 666 /var/log/esc/escmanager.log
ls -al /var/log/esc/escmanager.log
-rw-rw-rw-. 1 esc-user esc-user 12671993 Sep 12 23:32 /var/log/esc/escmanager.log
```

6. Change the file permissions for the *mona.log* file.

```
ls -al /var/log/esc/mona/mona.log
-rw-r--r--. 1 esc-user esc-user 3937424 Sep 13 01:10 /var/log/esc/mona/mona.log
sudo chmod 666 /var/log/esc/mona/mona.log
ls -al /var/log/esc/mona/mona.log
-rw-rw-rw-. 1 esc-user esc-user 3940388 Sep 13 01:11 /var/log/esc/mona/mona.log
```

7. Restart the syslog service.

```
sudo service rsyslog restart
```

8. Repeat steps 1, on page 9 through 7, on page 10 on the standby ESC VNFM VM.

Event Aggregation

The AutoIT module within the Ultra M Manager Node can be configured to aggregate events received from different Ultra M components as identified in Table 1: Component Event Source Domains, on page 11.

Important This functionality is currently supported only with Ultra M UGP VNF deployments based on OSP 10 or OSP 13 and that leverage the Hyper-Converged architecture. In pre-6.2 releases, this functionality was made available through the Ultra M Manager utility. The Ultra M Manager RPM is still distributed separately and is intended only for use in specific deployment scenarios. Contact your local sales or support representative for more information.

Table 1: Component Event Source Domains

Solution Component Domain	Event Source Type	Details
hardware (UCS server hardware)	CIMC	Reports on events collected from UCS C-series hardware via CIMC-based subscription. These events are monitored in real-time.
vim (VIM (Overcloud))	OpenStack service health	Reports on OpenStack service fault events pertaining to: • Failures (stopped, restarted) • High availability • Ceph / storage • Neutron / compute host and network agent • Nova scheduler (VIM instances) Refer to Table 2: Monitored OpenStack Services, on page 13 for a complete list of services.
uas (UAS AutoVNF)	UAS cluster	Reports on UAS service fault events pertaining to: • Service failure (stopped, restarted) • High availability
vnfm (ESC-based VNFM)	ESC (VNFM) event notifications	Reports on ESC-based VNFM service fault events pertaining to: • Service failure (stopped, restarted) **Important** Events on a per-VNFM VM level.

Solution Component Domain	Event Source Type	Details
vnf-em (UEM)	USP management component events	Reports on UEM service fault events pertaining to: • Service failure (stopped, restarted) • High availability • Internal application errors (e.g. SCM, LCM, etc.)
vnf (VNF VM Status)	ESC (VNFM) event notifications	Reports on VNF VM deployment state events generated by ESC (the VNFM). The following events are supported: • VM_DEPLOYED • VM_ALIVE • VM_UNDEPLOYED • VM_REBOOTED • VM_RECOVERY_REBOOT • VM_RECOVERY_UNDEPLOYED • VM_RECOVERY_DEPLOYED • VM_RECOVERY_COMPLETE • VM_STOPPED **Important** This feature was introduced in 6.0. It was not fully qualified and made available only for testing purposes. In 6.0, AutoVNF monitors for event notifications from ESC in real time. Though AutoVNF updates the VNFR for the VNF and VNFC the event pertains to upon receipt of an event, it does not generate a corresponding SNMP trap. It is fully qualified and fully functional as of the 6.2 release.

Table 2: Monitored OpenStack Services

Node Type	OpenStack Module	OpenStack Services
Controller	cinder	• openstack-cinder-api.service, • openstack-cinder-scheduler.service
	glance	• openstack-glance-api.service, • openstack-glance-registry.service
	heat-engine	openstack-heat-engine.service
	heat-api	• openstack-heat-api-cfn.service, • openstack-heat-api-cloudwatch.service, • openstack-heat-api.service
	heat	• openstack-heat-api-cfn.service, • openstack-heat-api-cloudwatch.service, • openstack-heat-api.service
	nova	• openstack-nova-api.service, • openstack-nova-conductor.service, • openstack-nova-consoleauth.service, • openstack-nova-novncproxy.service, • openstack-nova-scheduler.service
	swift-object	• openstack-swift-object-auditor.service, • openstack-swift-object-replicator.service, • openstack-swift-object-updater.service, • openstack-swift-object.service
	swift-account	• openstack-swift-account-auditor.service, • openstack-swift-account-reaper.service, • openstack-swift-account-replicator.service, • openstack-swift-account.service
	swift-container	

Node Type	OpenStack Module	OpenStack Services
		• openstack-swift-container-auditorservice, • openstack-swift-container-replicatorservice, • openstack-swift-container-updaterservice, • openstack-swift-container.service
	swift-proxy	openstack-swift-proxy.service
	swift	All above swift services
	ntpd	ntpd.service
	mongod	mongod.service
	memcached	memcached
	neutron-dhcp-agent	neutron-dhcp-agent.service
	neutron-l3-agent	neutron-l3-agent.service
	neutron-metadata-agent	neutron-metadata-agent.service
	neutron-openvswitch-agent	neutron-openvswitch-agent.service
	neutron-server	neutron-server.service
	httpd	httpd.service
OSD Compute	ceph-mon.target	ceph-mon.target
	ceph-radosgw.target	ceph-radosgw.target
	ceph.target	ceph.target
	openvswitch.service	openvswitch.service
	neutron-sriov-nic-agent	neutron-sriov-nic-agent.service
	neutron-openvswitch-agent	neutron-openvswitch-agent.service
	ntpd	ntpd.service
	nova-compute	openstack-nova-compute.service
	libvirtd	libvirtd.service

Node Type	OpenStack Module	OpenStack Services
Compute	ceph-mon.target	ceph-mon.target
	ceph-radosgw.target	ceph-radosgw.target
	ceph.target	ceph.target
	openvswitch.service	openvswitch.service
	neutron-sriov-nic-agent	neutron-sriov-nic-agent.service
	neutron-openvswitch-agent	neutron-openvswitch-agent.service
	ntpd	ntpd.service
	nova-compute	openstack-nova-compute.service
	libvirtd	libvirtd.service

Faults can be enabled or disabled at various levels as described in Configuring Fault Suppression, on page 21.

Events received from the solution components, regardless of the source type, are mapped against the Ultra M SNMP MIB (CISCO-ULTRAM-MIB.my, refer to Ultra M MIB). The event data is parsed and categorized against the following conventions:

- **Fault code:** Identifies the area in which the fault occurred for the given component. Refer to the "CFaultCode" convention within the Ultra M MIB for more information.
- **Severity:** The severity level associated with the fault. Refer to the "CFaultSeverity" convention within the Ultra M MIB for more information. Since the Ultra M Manager Node aggregates events from different components within the solution, the severities supported within the Ultra M Manager Node MIB map to those for the specific components. Refer to Ultra M Component Event Severity and Fault Code Mappings for details.
- **Domain:** The component in which the fault occurred (e.g. UCS hardware, VIM, UEM, etc.). Refer to the "CFaultDomain" convention within the Ultra M MIB for more information.

UAS and OpenStack events are monitored at the configured polling interval as described in Table 3: SNMP Fault Entry Table Element Descriptions, on page 17. At the polling interval, the Ultra M Manager Node:

1. Collects data from UAS and OpenStack.
2. Generates/updates .log and .report files and an SNMP-based fault table with this information. It also includes related data about the fault such as the specific source, creation time, and description.
3. Processes any events that occurred:
   a. If an error or fault event is identified, then a .error file is created and an SNMP trap is sent.
   b. If the event received is a clear condition, then an informational SNMP trap is sent to "clear" an active fault.
   c. If no event occurred, then no further action is taken beyond Step 2.

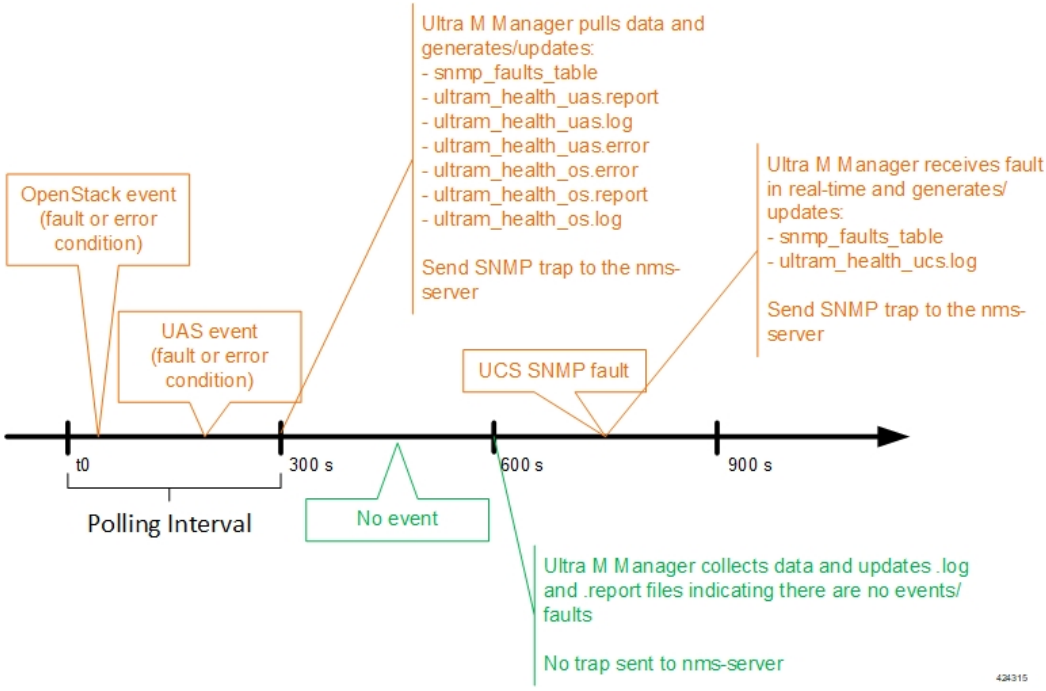
UCS and ESC VM events are monitored and acted upon in real-time. When events occur, the Ultra M Manager generates a .log file and the SNMP fault table. In the case of VM events reported by ESC, upon receipt of an event, AutoVNF updates the VNFR for the VNF and VNFC the event pertains to. In parallel, it passes the event information to the Ultra M Manager functionality within AutoIT. The Ultra M Manager then generates corresponding SNMP traps for each event.

Active faults are reported “only” once and not on every polling interval. As a result, there is only one trap as long as this fault is active. Once the fault is “cleared”, an informational trap is sent.

Important UCS events are considered to be the “same” if a previously received fault has the same distinguished name (DN), severity, and lastTransition time. UCS events are considered as “new” only if any of these elements change.

These processes are illustrated in Figure 2: Ultra M Manager Node Event Aggregation Operation, on page 16. Refer to About Ultra M Manager Log Files for more information.

Figure 2: Ultra M Manager Node Event Aggregation Operation

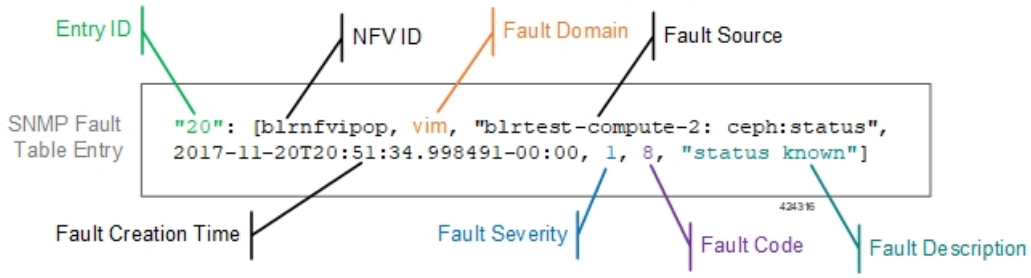


An example of the snmp_faults_table file is shown below and the entry syntax is described in Figure 3: SNMP Fault Table Entry Description, on page 17:

```
"0": [3 "neutonoc-osd-compute-0: neutron-sriov-nic-agent.service" 1 8 "status known"] "1":
 [3 "neutonoc-osd-compute-0: ntpd" 1 8 "Service is not active state: inactive"] "2": [3
"neutonoc-osd-compute-1: neutron-sriov-nic-agent.service" 1 8 "status known"] "3": [3
"neutonoc-osd-compute-1: ntpd" 1 8 "Service is not active state: inactive"] "4": [3
"neutonoc-osd-compute-2: neutron-sriov-nic-agent.service" 1 8 "status known"] "5": [3
"neutonoc-osd-compute-2: ntpd" 1 8 "Service is not active state: inactive"]
```

Refer to About Ultra M Manager Log Files for more information.

Figure 3: SNMP Fault Table Entry Description



Each element in the SNMP Fault Table Entry corresponds to an object defined in the Ultra M SNMP MIB as described in Table 3: SNMP Fault Entry Table Element Descriptions, on page 17. (Refer also to Ultra M MIB.)

Table 3: SNMP Fault Entry Table Element Descriptions

SNMP Fault Table Entry Element	MIB Object	Additional Details
Site ID	cultramSiteId	Identify fault at site level
Entry ID	cultramFaultIndex	A unique identifier for the entry
NFV ID	cultramNFVIdenity	Ultra M PoD on which this fault is occurring
Fault Domain	cultramFaultDomain	The component area in which the fault occurred. Refer to Table 1: Component Event Source Domains, on page 11 for information on domains supported in this release.
Fault Source	cultramFaultSource	Information identifying the specific component within the Fault Domain that generated the event. The format of the information is different based on the Fault Domain. Refer to Table 4: cultramFaultSource Format Values, on page 19 for details.
Fault Creation Time	cultramFaultCreationTime	The date and time when the fault was occurred.

SNMP Fault Table Entry Element	MIB Object	Additional Details
Fault Severity	cultramFaultSeverity	The severity associated with the fault as one of the following: • **emergency(1) :** System level FAULT impacting multiple VNFs/Services • **critical(2) :** Critical Fault specific to VNF/Service • **major(3) :** component level failure within VNF/service. • **alert(4) :** warning condition for a service/VNF, may eventually impact service. • **informational(5) :** informational only, does not impact service Refer to Ultra M Component Event Severity and Fault Code Mappings for details on how these severities map to events generated by the various Ultra M components.
Fault Code	cultramFaultCode	A unique ID representing the type of fault as. The following codes are supported: • **other(1) :** Other events • **networkConnectivity(2) :** Network Connectivity Failure Events • **resourceUsage(3) :** Resource Usage Exhausted Event • **resourceThreshold(4) :** Resource Threshold crossing alarms • **hardwareFailure(5) :** Hardware Failure Events • **securityViolation(6) :** Security Alerts • **configuration(7) :** Config Error Events • **serviceFailure(8) :** Process/Service failures Refer to Ultra M Component Event Severity and Fault Code Mappings for details on how these fault codes map to events generated by the various Ultra M components.
Fault Description	cultramFaultDescription	A message containing details about the fault.

Table 4: cultramFaultSource Format Values

FaultDomain	Format Value of cultramFaultSource
Hardware (UCS Servers)	**Node:** *<UCS-SERVER-IP-ADDRESS>*, **affectedDN:** *<FAULT-OBJECT-DISTINGUSIHED-NAME>* Where: <*UCS-SERVER-IP-ADDRESS*> *:* The management IP address of the UCS server that generated the fault. <*FAULT-OBJECT-DISTINGUSIHED-NAME*> *:* The distinguished name of the affected UCS object.
UAS	**Node:** *<UAS-MANAGEMENT-IP>* Where: <*UAS-MANAGEMENT-IP*> *:* The management IP address for the UAS instance.
VIM (OpenStack)	*<OS-HOSTNAME>*: *<SERVICE-NAME>* Where: <*OS-HOSTNAME*> : The OpenStack node hostname that generated the fault. <*SERVICE-NAME*> : Then name of the OpenStack service that generated the fault.

SNMP Version Support

The following commands are supported for both SNMP Version 2 and Version 3:

- GET
- Walk
- GETNEXT
- GETBULK

The following security algorithms are supported for SNMP Version 3:

Table 5: Supported SNMP Version 3 Security Algorithms

Protocol	Algorithms
Authentication	• usmNoAuthProtocol • usmHMACMD5AuthProtocol • usmHMACSHAAuthProtocol

Protocol	Algorithms
Privacy	• usmNoPrivProtocol • usmDESPrivProtocol • usm3DESEDEPrivProtocol • usmAesCfb128Protocol • usmAesCfb192Protocol • usmAesCfb256Protocol

For SNMP Version 3, the SNMP Engine ID is generated in accordance with RFC 3411:

```
(80000000 OR HEX value of enterprise ID) + 04 + (HEX value of Administratively Assigned
String)
```

Important The name of the network service descriptor (NSD) in which fault management functionality is configured is used as the 'Administratively Assigned String'. For deployment scenarios that require the Ultra M Manager RPM for fault maangement functionality, the name of the UCS cluster is used.

SNMP configuration is based on parameters configured in the fault management descriptor (FMD) along with other parameters pertaining to Ultra M health monitoring. Refer to Configuring Event Aggregation, on page 20 for more information on configuring and activating the FMD. Refer to the *Cisco Ultra Services Platform NETCONF API Guide* for more information on the specific parameters that comprise the FMD.

Configuring Event Aggregation

Event aggregation functionality is configured through NETCONF API-based remote procedure calls invoked via AutoIT. In either scenario, the parameters related to this functionality are defined by/within the fault management descriptor (FMD). When the VNF is deployed, the FMD configuration is merged into the existing NSD configuration. (Refer to the *Cisco Ultra Services Platform NETCONF API Guide* for details on the parameters supported within the FMD.)

Though the FMD configuration can be included in the NSD configuration file, it is recommended that the configuration for this functionality be maintained in a separate, FMD-specific NSD configuration file.

Important The instructions in this section assume that the Ultra M solution has been completely deployed prior to proceeding. This includes the VIM Orchestrator, the VIM, the UAS components, and the VNF.

To enable this functionality on the Ultra M solution:

1. Log on to the primary AutoIT VM as the root user.
2. Prepare the FMD configuration file for your deployment scenario. Your file should include the following parameters:
   - SNMP user configuration
   - Fault management descriptor (FMD) configuration

- Domain configuration (e.g. hardware, vim, uas, etc.)
- SNMP version and receiver configuration

Refer to Sample FMD Configuration File for a sample configuration file. Refer to the *Cisco Ultra Services Platform NETCONF API Guide* for a complete list of supported parameters.

3. Login to the ConfD CLI as the *admin* user.

   confd_cli -u admin -C

4. Enter the *admin* user password when prompted.
5. Enter the ConfD configuration mode.

   config

6. Load the FMD configuration file.

   load merge *<your_fmd_file_name>*.**cfg**

   commit

   end

7. Activate the FMD configuration.

   activate nsd-id *<nsd_name>* **fmd** *<fmd_name>*

Important The output of this command is a transaction-id which can be used to monitor the deployment progress. If need be, the FMD configuration can be deactivated using the **deactivate** variant of this command.

8. Monitor the progress of the FMD creation by viewing transaction logs:

   show log *<transaction_id>* **| display xml**

   transaction_id is the ID displayed as a result of the **activate** command executed in step 7, on page 21.

Configuring Fault Suppression

AutoIT can be configured to monitor the fault events for a specified domain. The fault suppression functionality for VNFC(s) must be performed after the VIM has been deployed.

Important AutoIT must be configured with information for the event (fault and alarm) monitoring at the time it is deployed. Refer to the *Ultra Services Platform Deployment Automation Guide* for more information.

Fault suppression functionality is configured through a fault management descriptor (FMD) configuration file that is comprised of the required NETCONF parameters.

Depending on the configuration of this functionality, the faults can be automatically suppressed at the following levels:

- UCS server:

- **UCS cluster**: All events for all UCS nodes are suppressed.
- **UCS fault object distinguished names (DNs)**: All events for one or more specified UCS object DNs within are suppressed.
- **UCS faults**: One or more specified UCS faults are suppressed.

Important Fault suppression can be simultaneously configured at both the UCS object DN and fault levels.

- UAS and VNF components:
  - **UAS component cluster**: All events for all UAS components are suppressed.
  - **UAS component events**: One or more specified UAS component events are suppressed.

When faults are suppressed, event monitoring occurs as usual and the log report file shows the faults. However, suppressed faults are not reported over SNMP. Within the log file, suppress faults are preceded by the word "Skipping".

Suppressing UCS Faults

AutoIT can be configured to suppress UCS hardware faults based on fault ID or affected fault object distinguished names (DNs).

UCS incorporates the concept of DN where each entity is been assigned unique ID or namespace. Suppressing events for a given UCS fault object distinguished name (DN) stops the reporting of all events related to the DN. Suppression can be enabled for one or more DNs.

Important Fault suppression can be simultaneously configured at both the UCS object DN and fault levels.

Fault suppression functionality is configured through a fault management descriptor (FMD) configuration file that is comprised of the required NETCONF parameters. Refer to Sample FMD Configuration File for a sample configuration file.

Important Though the FMD configuration can be included in the network service descriptor (NSD) for your VNF, it is recommended that the configuration for this functionality be maintained in a separate, FMD-specific NSD configuration file.

Refer to the latest Cisco UCS Faults and Error Messages Reference Guide for more information https://www.cisco.com/c/en/us/support/servers-unified-computing/unified-computing-system/products-system-message-guides-list.html .

To configure fault suppression functionality for UCS server hardware:

1. Log on to the primary AutoIT VM as the root user.

2. Prepare the FMD configuration file for your deployment scenario. Your file should include the following parameters:

   ```
   domain hardware
     monitoring suppress-hw-affected-dn [ <dn_1>…<dn_n> ]
     monitoring suppress-hw-fault-id [ <fault_id_1>…<fault_id_n> ]
   ```

   The operators **suppress-hw-affected-dn** and **suppress-hw-fault-id** are optional. If these are not configured, the faults can be raised.

   Refer to Sample FMD Configuration File for a sample configuration file.

   For information related to UCS faults, see https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/ts/faults/reference/3-0/UCSFaultsErrorsRef_3-0/UCS_SEMs_3-0.html.

3. Login to the ConfD CLI as the *admin* user.

   ```
   confd_cli -u admin -C
   ```

4. Enter the *admin* user password when prompted.

5. Enter the ConfD configuration mode.

   ```
   config
   ```

6. Load the FMD configuration file.

   ```
   load merge <your_fmd_file_name>.cfg
   commit
   end
   ```

7. Activate the FMD configuration.

   ```
   activate nsd-id <nsd_name> fmd <fmd_name>
   ```

Important The output of this command is a transaction-id which can be used to monitor the deployment progress. If need be, the FMD configuration can be deactivated using the **deactivate** variant of this command.

8. Monitor the progress of the FMD creation by viewing transaction logs:

   ```
   show log <transaction_id> | display xml
   ```

   transaction_id is the ID displayed as a result of the **activate** command executed in step 7, on page 5.

Suppressing UAS Faults

AutoIT can be configured to suppress UAS faults based on UAS components (AutoVNF, UEM, and the VNFM (ESC)) or type of failure within the component. Fault suppression functionality is configured through a fault management descriptor (FMD) configuration file that is comprised of the required NETCONF parameters. Refer to Sample FMD Configuration File for a sample configuration file.

Important Though the FMD configuration can be included in the network service descriptor (NSD) for your VNF, it is recommended that the configuration for this functionality be maintained in a separate, FMD-specific NSD configuration file.

The health check must be enabled for UAS, VNFM, VNF and UEM before configuring fault suppression functionality.

To configure fault suppression functionality for UAS components:

1. Log on to the primary AutoIT VM as the root user.
2. Prepare the FMD configuration file for your deployment scenario. Your file should include the following parameters:

```
domain uas
  monitoring suppress-uas-fault [ overall ]
domain vnfm
  monitoring suppress-uas-fault [ overall ]
domain vnf-EM
  monitoring suppress-uas-fault [ api-endpoint ha-event cluster-ha]
domain vnf
  monitoring suppress-uas-fault [ overall ]
```

   Refer to Sample FMD Configuration File for a sample configuration file.

   NOTES:

   - **suppress-uas-fault**: This operator is optional and it accepts enum value in the YANG model.
   - **overall**: Suppresses faults for all the configured domains.
   - **api-endpoint**: This is applicable only to UEM domain. The fault is raised when EM applications/internal components are not in healthy state i.e. SCM/SLA/VNFM-PROXY is down.
   - **ha-event**: This is applicable to UAS and UEM domain. The fault is raised when EM HA endpoint is changed i.e. during HA switch over (one of the UAS VMs in a cluster rebooted and a switchover/failover has been performed resulting in the election of a new master).
   - **cluster-ha**: This is applicable to UAS and UEM domain. The fault is raised when EM VMs are failed to form HA cluster.
   - The **api-endpoint** operator is not applicable for the UAS and ESC domains as the UAS/ESC health check procedure takes care of these errors and corresponding recovery procedures.

3. Login to the ConfD CLI as the *admin* user.

```
confd_cli -u admin -C
```

4. Enter the *admin* user password when prompted.
5. Enter the ConfD configuration mode.

```
config
```

6. Load the FMD configuration file.

```
load merge <your_fmd_file_name>.cfg
commit
end
```

7. Activate the FMD configuration.

```
activate nsd-id <nsd_name> fmd <fmd_name>
```

Important The output of this command is a transaction-id which can be used to monitor the deployment progress. If need be, the FMD configuration can be deactivated using the **deactivate** variant of this command.

8. Monitor the progress of the FMD creation by viewing transaction logs:

```
show log <transaction_id> | display xml
```

transaction_id is the ID displayed as a result of the **activate** command executed in step 7, on page 5.