Monitoring ESC Health

You can monitor the health of ESC and its services, using one of the following:

- Monitoring Health API
- SNMP Trap

- Monitoring the Health of ESC Using REST API, on page 1
- Monitoring the Health of ESC Using SNMP Trap Notifications, on page 6

Monitoring the Health of ESC Using REST API

ESC provides REST API for any third party software to monitor the health of ESC and its services. Using the API, the third party software can query the health condition of ESC periodically to check whether ESC is in service. In response to the query, API provides status code and messages, see Table 1: ESC Health API Status Code and Messages in Standalone and Active-Standby High Availability, on page 3 for details. In an HA setup the virtual IP (VIP) must be used as the monitoring IP. The return value provides the overall condition of the ESC HA pairs. See the Table 3: Health API Status Messages for Standalone ESC and HA, on page 5 for details.

The REST API to monitor the health of ESC is as follows:

```
GET to https://<esc_vm_ip>:8060/esc/health
```

Note

- The monitoring health API is secured using the existing REST basic HTTP authentication. The user can retrieve the report by using the ESC REST API credentials.
- The ESC Health API port number is changed from 60000 to 8060.

The monitoring health API response with error conditions is as follows:

```
<?xml version="1.0" encoding="UTF-8" ?>
<esc_health_report>
<status_code>{error status code}</status_code>
<message>{error message}</message>
</esc_health_report>
```

The monitoring health API response for local Active/Active is as follows:

```
<?xml version="1.0" encoding="UTF-8" ?>
<esc_health_report>
    <status_code>2010</status_code>
    <message>ESC service is being provided. ESC AA cluster one or mode node(s) not
healthy</message>
    <nodes>
        <node>
            <name>aa-esc-1.novalocal</name>
            <status>HEALTHY</status>
            <datacenter>dc1</datacenter>
            <services>
                <service>
                    <name>escmanager</name>
                    <status>running</status>
                    <is_expected>True</is_expected>
                </service>
                <service>
                    <name>elector</name>
                    <status>leader</status>
                    <is_expected>True</is_expected>
                </service>
                <service>
                    <name>drbd</name>
                    <status>master</status>
                    <is_expected>True</is_expected>
                </service>
                <service>
                    <name>pgsql</name>
                    <status>running</status>
                    <is_expected>True</is_expected>
                </service>
                ...
            </services>
        </node>
        <node>
            <name>aa-esc-2.novalocal</name>
            <status>HEALTHY</status>
            <datacenter>dc1</datacenter>
            <services>
                <service>
                    <name>escmanager</name>
                    <status>running</status>
                    <is_expected>True</is_expected>
                </service>
                <service>
                    <name>elector</name>
                    <status>follower</status>
                    <is_expected>True</is_expected>
                </service>
                <service>
                    <name>drbd</name>
                    <status>backup</status>
                    <is_expected>True</is_expected>
                </service>
                <service>
                    <name>pgsql</name>
                    <status>stopped</status>
                    <is_expected>True</is_expected>
                </service>

                ...
            </services>
        </node>
        <node>
```

```
            <name>aa-esc-3.novalocal</name>
            <status>NOT_HEALTHY</status>
            <datacenter>dc1</datacenter>
            <services>
                <service>
                    <name>escmanager</name>
                    <status>stopped</status>
                    <is_expected>False</is_expected>
                </service>
                <service>
                    <name>elector</name>
                    <status>follower</status>
                    <is_expected>True</is_expected>
                </service>
                <service>
                    <name>vimmanager</name>
                    <status>running</status>
                    <is_expected>True</is_expected>
                </service>
                ...
            </services>
        </node>
    </nodes>
</esc_health_report>
```

XML and JSON responses are also supported for the monitoring health API.

If the API response is successful, an additional field called *stage* is introduced.

```
<?xml version="1.0" encoding="UTF-8" ?>
<esc_health_report>
<status_code>{success status code}</status_code>
<stage>{Either INIT or READY}</stage>
<message>{success message}</message>
</esc_health_report>
```

The stage field has INIT or READY parameters.

INIT: The INIT parameter is the initial stage, where ESC accepts *pre-provisioning* requests such as configuring the config parameters or registering a vim connector.

READY: ESC is ready for any kind of *provisioning* requests such as deploying, undeploying and so on with this parameter.

The status code and messages below provide the health condition of ESC. The status codes with 2000 series imply that the ESC is operational. The status codes with 5000 series imply that at least one ESC component is not in service.

Note The ESC Health API does not check the VIM status because of multi VIM deployment introduced in ESC Release 3.0.

Table 1: ESC Health API Status Code and Messages in Standalone and Active-Standby High Availability

Status Code	Message
2000	ESC services are running.
2010	ESC services are being provided. ESC AA cluster one or more node(s) not healthy.

Status Code	Message
2020	ESC services are running. One or more VIM services (for example, keystone and nova) not reachable. **Note** Not supported from ESC Release 3.0.
2030	ESC services are running, but VIM credentials are not provided. **Note** Not supported from ESC Release 3.0.
2040	ESC services running. VIM is configured, ESC initializing connection to VIM.
2100	ESC services are running, but ESC High-Availability node is not reachable. One or more VIM services (for example, nova) are not reachable. **Note** Not supported from ESC Release 3.0.
5010	ESC service, ESC_MANAGER is not running.
5020	ESC service, CONFD is not running.
5030	ESC service, MONA is not running.
5040	ESC service, VIM_MANAGER is not running.
5090	More than one ESC service (for example, confd and mona) are not running.

Table 2: ESC Health API Status Code and Messages in Active-Active High Availability

Status Code	Message
2000	ESC services are running (Active-Active setup).
2010	ESC services are provided. In ESC Active/Active cluster one or more node(s) are not healthy.
5000	ESC services not being provided, ESC AA cluster not healthy

Note ESC HA mode refers to ESC HA in DRBD setup only. For more information on the ESC HA setup, see the Cisco Elastic Services Controller Install Guide.

The table below describes the status message for standalone ESC and HA with success and failure scenarios. For more information on ESC standalone and HA setup, see the Cisco Elastic Services Controller Install Guide.

Table 3: Health API Status Messages for Standalone ESC and HA

	Success	Partial Success	Failure
Standalone ESC	The response is collected from the monitoring health API and the status code is 2000.	NA	• Monitor cannot get the response from the monitoring health API. • The response is collected from the monitoring health API and the status code returned is in the 5000 series.
ESC in HA (Active-Standby)	The response is collected from the monitoring health API and the status code is 2000.	The response is collected from the monitoring health API and the status code is 2010. This indicates that the ESC standby node cannot connect to ESC master node in ESC HA. However, this does not impact the ESC service to northbound.	• The monitor cannot get the response from the monitoring health API for more than two minutes. **Note** ESC monitoring health API may not be available for a certain period during the HA switchover period. The monitoring software must set a proper threshold to report service failure in this scenario. • The response is collected from the monitoring health API and the status code returned is in the 5000 series.

	Success	Partial Success	Failure
ESC in HA (Active-Active)	The response is collected from the monitoring health API and the status code is 2000.	The response is collected from the monitoring health API and the status code is 2010. This indicates that the ESC services are being provided but one or more nodes are not healthy in the ESC AA cluster. This does not impact the ESC service to northbound.	• For Local Active-Active, if the monitor cannot get the response from the monitoring health API for more than one minute. For Geo Active-Active, if the monitor cannot get the response from the monitoring health API for more than seven minutes (this depends on the configuration in heat template) **Note** ESC monitoring health API may not be available for a certain period during the local and geo switchover period. The monitoring software must set a proper threshold to report service failure in this scenario. • The geo switchover period depends upon the configuration in the heat template. By default, the switchover starts five minutes after the primary datacenter failure. The response is collected from the monitoring health API and the status code returned is 5000. **Note** During switchover, the status code returned will temporarily be 5000 until the new leader becomes healthy.

Monitoring the Health of ESC Using SNMP Trap Notifications

You can also configure notifications on the health of various ESC components via SNMP traps using an SNMP Agent. This Agent is installed as part of the standard ESC installation and supports the SNMP version 2c protocol. The SNMP traps currently support only the state of the ESC product and not of the VNFs managed by ESC. This section describes the steps required to configure the ESC SNMP agent and also cover the events that will be triggered as part of the notifications.

Before you begin

- Ensure the **CISCO-ESC-MIB** and **CISCO-SMI MIB** files are available on your system. These are located in the /opt/cisco/esc/snmp/mibs directory. Download these to your SNMP Manager machine and place them in the $HOME/.snmp/mibs directory.

- Configure SNMP Agent. There are three methods to configure SNMP agent. These methods are discussed in detail in the section below.

Configuring SNMP Agent

In order to receive the SNMP traps, configure the SNMP Agent parameters. The agent can be configured using three different methods described in this section. The best or most applicable method to use depends on your use case.

1. **Enabling and configuring SNMP Agent during ESC installation**:

   - **Standalone or Active/Standby HA setup via BootVM**

     While installing ESC, use the following additional parameters to configure SNMP agent:

     ```
     % bootvm.py <esc_vm_name> --image <image-name> --net <net-name> --enable-snmp-agent
      --ignore-ssl-errors
     --managers "udp:ipv4/port,udp:[ipv6]/port"
     ```

     Note The value for managers is a comma separated list of locations where SNMP traps are delivered in the format "udp:ipv4/port" or "udp:[ipv6]/port". The IP and port must be replaced with the actual values.

   - **Active/Active HA setup**

     You can enable the SNMP agent during the Active/Active installation. You can pass the config parameters `ignore_ssl_errors` and list of `managers` to configure the agent on install. It can be defined in the aa-params.yaml or passed on the following command line.

     ```
     openstack stack create name-aa --template aa.yaml -e aa-params.yaml \
     --parameter nameprefix=ESC_AA \
     --parameter image_name=ESC-5_2_0_43 \
     --parameter flavor_name=m1.large \
     ...
     --parameter snmp_agent_startup: auto \
     --parameter snmp_agent_ignore_ssl_errors: true \
     --parameter snmp_agent_managers: [ "udp:ipv4/port,udp:[ipv6]/port" ]
     ```

2. **Enabling and configuring via ESCADM**

   - **Standalone or Active/Standby HA setup**

     Using the escadm tool, you can modify the SNMP agent configuration parameters such as managers and ignoreSslErrors properties.

     ```
     sudo escadm snmp set --ignore_ssl_errors=true
     --managers="udp:ipv4/port,udp:[ipv6]/port"
     ```

   - **Active/Active HA setup**

     Run the following command on all the Leader ready nodes which is the ESC node 1, node 2, node 4, and node 5:

     ```
     sudo escadm snmp set --startup=auto
     ```

Note If a node is deleted and recreated by a stack update, you must rerun the previous command.

Restart ESC services on the SNMP enabled nodes only on the primary datacenter which is node 1 and 2. One node at a time.

```
sudo escadm stop
sudo escadm start
```

Once the leader node is healthy, and SNMP agent is running, you can add the SNMP agent configurations on the leader node as follows.

```
sudo escadm snmp set --ignore_ssl_errors=true
--managers="udp:ipv4/port,udp:[ipv6]/port"
```

Note The `ignore-ssl-errors` parameter is mainly for a developer environment to prevent SSL errors, where self signed certificates are used on the ESC VM.

The value for managers is a comma separated list of locations where SNMP traps are delivered `"udp:ipv4/port" or "udp:[ipv6]/port"` format. The IP and port must be replaced with the actual values.

3. **Updating the configuration file**

   The SNMP agent must already be enabled for this configuration update to take effect.

   The configuration is in the file /opt/cisco/esc/esc_database/snmp.conf. This file is in JSON format. Following is an example:

```
{"sysDescr": "ESC SNMP Agent",
    "listeningPort": "2001",
    "managers": [
        "udp:[ipv4]/port",
        "udp:[ipv6]/port"
    ],
    "ignoreSslErrors": "yes",
    "logLevel": "INFO",
    "sysLocation": "Unspecified",
    "sysName": "system name",
    "pollSeconds": "15",
    "listeningAddress": "0.0.0.0",
    "healthUrl": "https://<esc_vm_ip>:8060/esc/health",
    "sysContact": "root@localhost"}
```

Defining ESC SNMP MIBs

The following table describes the content of ESC MIB. These values are configurable in the snmp.conf file.

Variable	Simple IOD	Description
sysName	SNMPv2-MIB::sysName.0	Specify the name of the ESC machine. The host name is taken by default.
sysDescr	SNMPv2-MIB::sysDescr.0	Specify the name of the SNMP Agent.
sysLocation	SNMPv2-MIB::sysLocation.0	Specify where the ESC machine is located.
sysContact	SNMPv2-MIB::sysContact.0	Specify the Admin contact.

Enabling SNMP Trap Notifications

Use the escadm tool to start the SNMP services.

```
sudo escadm snmp start
```

You can also use esadm tool to stop, get the status, and modify the configurations of the SNMP agent.

```
sudo escadm snmp stop
sudo escadm snmp status
sudo escadm snmp restart
```

Managing SNMP Traps in ESC

This section covers:

- Understanding the SNMP Notification Types in ESC.
- Receiving SNMP Trap Message Directly From the Network
- Managing Trap Endpoints (SNMP Managers)
- Managing ESC SNMP in an HA Environment
- Managing ESC SNMP Agent in an Active/Active Environment
- Managing Self-Signed Certificates in ESC

Procedure

- **Understanding the SNMP Notification Types in ESC:** The following table lists all the events supported by this version of the SNMP agent. These status codes and messages will be returned via a SNMP trap to a registered manager only when there is a change of state of ESC. The status codes with 2000 series imply that the ESC is operational. The status codes with 5000 series imply that at least one ESC component is not in service. For more details on status codes with 2000 series and 5000 series, see section, *Monitoring ESC Health Using REST API*.

Status Code	SNMP Agent-specific Message
5100	An HTTP error was received when using the ESC Monitor API
5101	The ESC Monitor replied, but the data could not be understood.
5102	The Agent could not create a network connection to the ESC Monitor API.
5199	An unhandled error occurred (details will be included in the message).
5210	"**AA LEADER node change**". In an AA environment where a node has become the LEADER, the agent on the node will send this notification. Only for local leader change.
5200	"**HA MASTER node change**". In an A/S HA environment where a node has become the MASTER node the agent sends this notification.
5220	"**Geo AA Primary datacenter change**" In a GEO A/A environment, after GEO switchover, when a node becomes the LEADER, the agent on the node will send this notification. Only for GEO leader change.

- **Receiving SNMP trap messages directly from the network**: Directly receive SNMP trap messages from the network, by using basic SNMP UNIX tools such as, snmpget snmpwalk and snmptrapd. An example usage:

```
snmptrapd -m ALL -f -Lo -c snmptrapd.conf <port>
```

This will start an SNMP trap daemon on port 12113. Make sure the Cisco and ESC MIB's are present in ~/.snmp/mibs. The referenced snmptrapd.conf looks like this:

```
disableAuthorization yes
authCommunity   log,execute,net public
# traphandle default /Users/ahanniga/bin/notify.sh esc

createUser myuser MD5 mypassword DES myotherpassword

format2 %V\n% Agent Address: %A \n Agent Hostname: %B \n Enterprise OID: %N \n Trap
Sub-Type: %q \n Community/Infosec Context: %P \n Uptime: %T \n PDU Attribute/Value Pair
 Array:\n%v \n -------------- \n
```

The trap will contain two entries: statusCode and statusMessage. The trap will be sent when the status changes

```
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (3971) 0:00:39.71
SNMPv2-MIB::snmpTrapOID.0 = OID: CISCO-ESC-MIB::statusNotif
SNMPv2-MIB::sysDescr.0 = STRING: ESC SNMP Server
CISCO-ESC-MIB::escStatusCode.0 = STRING: "2000"
CISCO-ESC-MIB::escStatusMessage.0 = STRING: "ESC services are running."
```

- **Managing Trap Endpoints (SNMP Managers)**: The SNMP Agent monitors its configuration file for changes and reloads when a change is made. Add or remove manager endpoints to the configuration file and the new configuration will be used in future traps.
- **Managing ESC SNMP Agent in an HA Environment**: Two or more ESC nodes can be deployed in a HA configuration and the SNMP agent does support this configuration. However, consider the following points in an HA deployment:
  - Only one ESC node (the master node) can send SNMP traps
  - The SNMP Agent must be up if the backup node becomes the master.
  - Any changes made to the master configuration must also be applied on backup nodes.
  - If a node becomes the master node due to failover, this will generate a trap.
- **Managing ESC SNMP Agent in an AA Environment**: SNMP agent service is also supported in local or GEO ESC Active/Active setup. Following are the considerations in an Active/Active deployment:
  - SNMP Agent runs and sends traps on the leader node only.
  - Traps are sent in the following scenarios:
    - On ESC health API status code change. The SNMP Agent polls the Health Monitor API for AA, if there is a change in the status code returned, it is sent as a trap to its subscribers.
    - After local switchover by the node which becomes the new Leader to signify local switchover.
    - After GEO switchover by the node which becomes Leader in new GEO Primary datacenter.
  - Changes made to the configuration in leader node is carried forward by new leader after switchover.
- **Managing Self-Signed Certificates**: When ESC is deployed and the SNMP agent uses ESC Health APIs, it is recommended that a root trusted certificate is installed on the server. If the environment is a known and trusted one then it is possible to ignore these errors using the configuration parameter "ignoreSslErrors". However, if you did want to keep this setting to its more secure default it is possible to install a self-signed certificate by importing the ESC certificate into the JVM trust store. The following section describes the procedure to do so.
  a) Add esc as an alternative name for localhost. In the file "/etc/hosts:" add the following (or ensure that "esc" is added to the end):

     ```
     127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4 esc
     ```

  b) In the SNMP Agent configuration file "/opt/cisco/esc/esc_database/snmp.conf" the healthUrl must point to esc.

     ```
     "healthUrl": "https://esc:8060:/esc/health"
     ```

  c) Import the certificate into the truststore. Following is an example of importing the certificate, assuming $JAVA_HOME is/usr/lib/jvm/jre-1.8.0-openjdk.x86_64:

     ```
     cd /opt/cisco/esc/esc-config
     sudo openssl x509 -inform PEM -in server.pem -outform DER -out server.cer
     sudo keytool -importcert -alias esc -keystore $JAVA_HOME/lib/security/cacerts
     -storepass changeit -file server.cer
     ```