Cisco Cloud Object Storage Release 3.5.1 Troubleshooting Guide

November 24, 2015

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco Cloud Object Storage Release 3.5.1 Troubleshooting Guide

Preface

This preface describes who should read the *Cisco Cloud Object Storage Release 3.5.1 Troubleshooting Guide*, how it is organized, and its document conventions. It contains the following sections:

Audience

This guide is for the networking professional managing the Cisco Cloud Object Storage (COS) product. Before using this guide, you should have experience working with Linux platforms, and be familiar with the concepts and terminology of Ethernet, local area networking, clustering and high-availability, and network services like DNS and NTP.

Document Organization

This document contains the following chapters and appendices:

Chapters or Appendices	Descriptions
Chapter 1, “Troubleshooting COS Problems”	Provides information and procedures for troubleshooting general COS problems
Chapter 2, “UCS C3160 (Colusa) Troubleshooting Tips”	Provides CIMC troubleshooting tips and common troubleshooting scenarios for the UCS C3160

Document Conventions

This document uses the following conventions:

Convention	Indication
bold font	Commands and keywords and user-entered text appear in **bold** font.
italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
[ ]	Elements in square brackets are optional.
{x | y | z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
`courier` font	Terminal sessions and information the system displays appear in `courier` font.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Tip Means *the following information will help you solve a problem*. The tips information might not be troubleshooting or even an action, but could be useful information, similar to a Timesaver.

Caution Means *reader be careful*. In this situation, you might perform an action that could result in equipment damage or loss of data.

Timesaver Means *the described action saves time*. You can save time by performing the action described in the paragraph.

Warning **IMPORTANT SAFETY INSTRUCTIONS**

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of

each warning to locate its translation in the translated safety warnings that accompanied this device.

SAVE THESE INSTRUCTIONS

Warning **Statements using this symbol are provided for additional information and to comply with regulatory and customer requirements.**

Related Publications

Refer to the following documents for additional information about COS:

- *Release Notes for Cisco Cloud Object Storage Release 3.5.1*
- *Cisco UCS C3160 Rack Server Installation and Service Guide*
- *Cisco Content Delivery Engine 465 Hardware Installation Guide*
- *Cisco Content Delivery Engine 205/220/250/280/420/460/470 Hardware Installation Guide*
- *Cisco Cloud Object Storage Release 3.5.1 API Guide*
- *Cisco Cloud Object Storage Release 3.5.1 User Guide*
- *Cisco Media Origination System User Guide*

Note COS 3.5.1 has been tested for compatibility with MOS Release 2.4.3. Later releases of COS are expected to be compatible with later versions of MOS. Contact Cisco for the latest information.

- *Open Source Used in COS*

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation* at: http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html.

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation as an RSS feed and delivers content directly to your desktop using a reader application. The RSS feeds are a free service.

Troubleshooting COS Problems

Viewing the Statuses of Primary System Services

The status of the primary system services that provide COS services on a COS node can be viewed by executing the following commands:

```
[root@utah97 ~]# service cassandra status
cassandra is running
[root@utah97 ~]# service cosd status
cosd (pid  9235) is running...
[root@utah97 ~]# service cserver status
cserver is running
```

Service High-Availability

Each COS node leverages the monit framework to provide high-availability of the cos service. In the event that the cosd process crashes, the monit framework will restart the service.

Troubleshooting Swift and Swauth API Errors

If a Swift or Swauth API operation returns a 500 level HTTP error status, this can be an indication of an issue with one of the three primary system services — cassandra-server, cosd, or cserver. These error statuses are sometimes returned due to temporary resource exhaustion on the COS node. However, if the error statuses persist for a long period of time, verify that the primary system services are running. See Viewing the Statuses of Primary System Services, page 1-1.

Information helpful in tracing and analyzing Swift, Swauth, and Cassandra transactions can be found in the following log files collected on each individual COS node:

- /arroyo/log/http.log.<DATE>
  - This log collects HTTP transaction information from Swift and Swauth operations.
  - The following is an example of an entry for an RIO model write:

```
2015-09-03 05:29:34 UTC cosnode-1 : AUDIT : ffff8806c77f0b68 : 172.22.102.214:460006 <-> 192.169.220.2
: RIO WRITE OBJECT : PUT /rio/bucket1/id1 : 0x0000155db8b711 f9e : 202 1024x1 67.1Kbps :
(0:29:39:4:48:121)
```

In this example, the numbers in parenthesis at the end of the output represent the following information, listed in order:

- **<queue-time>:** The time the HTTP request stayed in the queue before initial processing started (0 ms in this example)
- **<initial-meta-data-time>:** The time it took for the initial object metadata creation in cassandra (29 ms in this example)
- **<tcp-receive-time>:** The time it took to receive the entire object over the network from the client (39 ms in this example)
- **<disk-write-completion-time>:** The time it took to write the expanded object (local and remote expansion in parallel) (4 ms in this example)
- **<final-meta-data-time>:** The time it took to finalize the object metadata in cassandra (48 ms in this example)
- **<total-time>:** The total time for the transaction (121 ms)

– The following is an example of an entry for a traditional model reads:

```
27-Jul-2015 22:59:28 UTC :: AUDIT : ffff88075578c850 : 20.0.52.37:52926 <-> 20.0.52.55 : SWIFT READ
OBJECT : GET /v1/AUTH_123/mycontainers2/CISCO24MB_4064 : 0x0000255b56034cb0 :
0-549755813887(549755813888) 1.00Mbps : 200 0-23999999(24000000:0) 432Mbps : (0:3:18:90:0:0:63:507)
```

The following describes some of the key fields in this output:

- 1.00 Mbps: The committed bandwidth
- 200: The HTTP response code
- 0-239999999: The returned range
- 24000000: The returned length
- 432 Mbps: The actual bandwidth
- 0:3:18:90:0:0:63:507: Time taken in various stages of the request, represented as milliseconds, listed in the following order:
  - m_queueDelayTicks
  - m_metadataDelayTicks
  - m_initialDataDelayTicks
  - m_totalDataWaitTicks
  - m_totalWindowWaitTicks
  - m_totalClientWindowClosedTicks
  - m_transmitStartDelayTicks
  - m_totalRequestTicks

– If the HTTP client issuing the REST API write request aborts prematurely, the HTTP response code is -1, as shown in the following example:

```
2015-08-19 15:45:11 UTC cosnode-1 : AUDIT : ffff8806c7000b68 : 172.22.102.214:3566
68 <-> 192.169.220.2 : SWIFT WRITE OBJECT : PUT /v1/AUTH_ea79aa8c-8656-4da9-9f8ee
-a69f49bdaa7f/container90/513G : 0x0000155d3f7dc005 : -1 0-548692869120 609Mbps
: (0:11:7199939:136:446:7200535
```

- /arroyo/log/cosd.log.<DATE>
  – This log records Cassandra transactions that are executed when a client invokes a Swift or Swauth API operation. In this log file watch for "err:", "wrn:", or "ftl: errors. These errors correspond to the standard Unix syslog levels of WARN(wrn), ERR(err), and FATAL(ftl).

- /arroyo/log/protocoltiming.log.<DATE>
  - This log provides information about any network interface issues and any disk issues.

Note When debugging, it may be helpful to raise the HTTP log level of the service in question using the command **echo 9 > /proc/calypso/tunables/http_log_level**.

Each COS node also includes a cos_stats utility that can be executed from a shell. The utility reports the current operational state of the COS node with respect to Swift and Swauth operations. This utility also provides information on resource and network utilization. The following is a sample output obtained by executing the cos_stats utility.

```
Fri Jun 27 13:23:12 PDT 2014
------------------------------------------------------------------------------
| Operation               |    Active |   Ops/Sec |    Errors |     Total  |
------------------------------------------------------------------------------
| SWAUTH Account Create   |         0 |       0.0 |         0 |          6  |
| SWAUTH Account Delete   |         0 |       0.0 |         0 |          0  |
| SWAUTH Account Meta     |         0 |       0.0 |         0 |          0  |
| SWAUTH User Create      |         0 |       0.0 |         0 |          6  |
| SWAUTH User Delete      |         0 |       0.0 |         0 |          0  |
| SWIFT Container Create  |         0 |       0.0 |         0 |          0  |
| SWIFT Container Delete  |         0 |       0.0 |         0 |          0  |
| SWIFT Container List    |         0 |       0.0 |         0 |          0  |
| SWIFT Container Meta    |         0 |       0.0 |         0 |       3298  |
| SWIFT Object Delete     |         0 |       0.0 |         0 |          0  |
| SWIFT Object List       |         0 |       0.0 |         0 |          0  |
| SWIFT Object Meta       |         0 |       0.0 |         0 |          0  |
| SWIFT Object Read       |      3999 |      1.9K |         0 |      13818  |
| SWIFT Object Write      |         0 |       0.0 |         0 |    6694580  |
| SWIFT Token Get         |         0 |       0.0 |         0 |          6  |
------------------------------------------------------------------------------
|    Eth |  Sessions | TCP RX bps | TCP RX pps | TCP TX bps | TCP TX pps    |
------------------------------------------------------------------------------
|   eth6 |      2004 |   196.320M |   407.518K |     8.511G |   716.957K    |
|   eth7 |      1024 |   170.025M |   352.604K |     8.239G |   688.469K    |
|   eth8 |      1095 |   159.576M |   328.401K |     8.038G |   673.944K    |
|  Total |      4123 |   525.920M |     1.089M |    24.788G |     2.079M    |
------------------------------------------------------------------------------
| Poll:   0% : 823K/s | NetInt:  62% | RX:  37% | TCP TX:  74% | FQ:   0%  |
------------------------------------------------------------------------------
```

Troubleshooting Cassandra Issues

Verify All Cassandra Nodes are Up and Running

To verify that all Cassandra nodes are up and running, on the COS node enter the command **nodetool status**. The status for the Cassandra nodes should show “UN” for Up, Normal. For example:

```
[root@cosnode log]# nodetool status
Datacenter: DC1
===============
Status=Up/Down|/ State=Normal/Leaving/Joining/Moving
--  Address         Load       Tokens  Owns (effective)  Host ID                               Rack
UN  172.22.125.16   66.3 KB    256     16.7%             990771b1-babd-4de6-883c-1748ada16410  RAC1
UN  172.22.125.48   66.22 KB   256     17.5%             0d309a8c-2bfe-4033-a4a0-83d9c4d1baf5  RAC1
```

```
UN  172.22.125.33   66.36 KB   256     17.6%            29c32e41-0ed0-45e2-9f14-ddd52424c27a  RAC1
UN  172.22.125.49   66.16 KB   256     16.1%            6e07235a-c8b2-425c-999e-81966f106584  RAC1
UN  172.22.125.52   246.73 KB  256     18.3%            66f2736c-b370-45f5-ae7d-7f80f709e01d  RAC1
```

VOD/cDVR/LIVE Capture Failed with Storage Failure

Possible reasons for failure:

- COS/NAS storage down or not reachable
- COS/NAS storage writes failed
- Storage network interface down

Correcting the problem:

- Check for possible errors or alarms on the external NAS storage or COS.
- Check for storage usage.
- Check for network connectivity between MCE nodes and the COS/NAS storage.

Impact on the end user:

- The recording will stop and may relocate to another worker.

Verifying the COS-Controller is Running on a PAM HA Node

To verify that all processes are running on a PAM HA node, issue the following API call:

```
GET http:pam_node:5067/v1/roles/leaders
```

Verify that the response includes the cos-controller role:

```
[rabbitmq, pam-installedVm, pam-docserver, cos-controller, pam-dns, service-mgr,
pam-platsrv, pam-zone, controller, pam-vmLoadBalancer, mgmt-docserver, mongo]
```

Unresponsive COS Service Manager GUI

If the COS Service Manager GUI is not responding, you should check COS AIC service status on the COS Service Manager CLI.

Step 1 Login to the COS Service Manager CLI.

Step 2 To view the status of all running services, execute this command:
supervisorctl

Step 3 Verify the status of the cos-controller. This represents the status of the COS AIC.

```
[root@PM-35-C90 supervisor]# supervisorctl
cli_insVmConfig_app              RUNNING    pid 3661, uptime 10 days, 14:51:03
cli_mongoWatcher                 RUNNING    pid 3656, uptime 10 days, 14:51:03
cli_mosdns_app                   RUNNING    pid 3655, uptime 10 days, 14:51:03
cli_pamds_app                    RUNNING    pid 3659, uptime 10 days, 14:51:03
```

```
cli_platformservice_app          RUNNING    pid 3658, uptime 10 days, 14:51:03
cli_restapi_app                  RUNNING    pid 3654, uptime 10 days, 14:51:03
cli_vmLoadBalancer_app           RUNNING    pid 3664, uptime 10 days, 14:51:03
cli_zone_app                     RUNNING    pid 3660, uptime 10 days, 14:51:03
cos-controller                   RUNNING    pid 30717, uptime 0:43:01
docserver_app                    RUNNING    pid 3644, uptime 10 days, 14:51:03
drm-system_app                   RUNNING    pid 3650, uptime 10 days, 14:51:03
mos-controller_app               RUNNING    pid 3648, uptime 10 days, 14:51:03
redis                            RUNNING    pid 3640, uptime 10 days, 14:51:03
roleserver_app                   RUNNING    pid 3653, uptime 10 days, 14:51:03
sm_app                           RUNNING    pid 3646, uptime 10 days, 14:51:03
unified_translogd                RUNNING    pid 3638, uptime 10 days, 14:51:03
```

General Information and Issues

Viewing COS AIC Logs

In addition to the COS Status and COS Events reported in COS Service Manager GUI, detailed COS AIC logs can be found on the COS Service Manager CLI. There are three logs:

- /var/log/opt/cisco/mos/errorlog/COSController-errorlog.current
  - This log records COS AIC execution information.
- /var/log/supervisor/cos-aic.log
  - This log records COS AIC related stdout information. Typically, any node.js related error messages written to stdout are captured in this log.
- /var/log/supervisor/cos-aic.err.log
  - This log records COS AIC related stderr information. Typically, any node.js related error messages written to stderr are captured in this log.

COS AIC Client Logs

The following COS AIC Client log files are available on the COS nodes:

- /arroyo/log/cos-aic-client.log—This is the primary COS AIC Client log file.
- /var/log/cos-aic-client.stderr—This log contains messages related to a COS AIC Client being terminated and is helpful in debugging unexpected crashes.

Linux Log Files

The Linux operating system has the following useful log files:

- /var/log/debugmessages—Syslog messages
- /var/log/messages—Includes useful bootup status messages

CServer Log Files

The COS node has the following useful log files:

- /arroyo/log/c2k.log.<*date*>—This log has information about content read issues. The date extension for the log filename has the format of yyyymmdd (for example, 20090115 is January 15, 2009). To increase the verbosity of this log file, use the following command:

  ```
  # echo "6" > /proc/calypso/tunables/c2k_verbosedump
  ```

- /arroyo/log/protocoltiming.log.<*date*>—Provides information about any network interface issues and any disk issues.
- /arroyo/log/stresstest.log.<*date*>—Provides CPU uptime information.

Other CServer log files that may be useful are the following:

- /arroyo/log/controlblocktiming.log.<*date*>
- /arroyo/log/debug.log.<*date*>
- /arroyo/log/decommissioned.log.<*date*>
- /arroyo/log/deleted.log.<*date*>
- /arroyo/log/executiontiming.log.<*date*>
- /arroyo/log/objectRepair.log.<*date*>
- /arroyo/log/serverinfo.log.<*date*>
- /arroyo/log/streamevent.log.<*date*>
- /arroyo/log/systemstats.log.<*date*>

Note The files with the extension <*date*> use the format yyyymmdd. The date is the Coordinated Universal Time (UTC) date.

CServer Error Codes

CServer error codes that appear in the c2k.log.<*date*> file do not necessarily mean an error has occurred. An actual error has “err” listed in the entry, as opposed to “out” or “ntc.” Following is a list of important CServer error and status codes:

Error Codes

- 5—Completion of a task.
- 25—Insufficient resources.

Status Codes

- 0—Content is okay (cnOK).
- 1—Stream has ended (cnEnd).
- 2—Stream has been paused (cnPaused).
- 3—Error has occurred (cnError).
- 4—Next element is being processed (cnNextElement).
- 5—Live content has resumed (cnResumeLive).
- 6—Next content object is being processed (cnNextContent).

- 7—Next iteration is being processed (cnNextIteration).
- 9—There has been a failover (cnFailover).
- 8—Stream has been destroyed (cnDestroyed).

Server Configuration Files

Identifying the Software Versions or Releases

The following sections describe the commands for identifying the software versions on the server.

Linux OS Version

To identify the software version of the Linux operating system (OS) on the Service Manager, enter the following command:

```
# cat /proc/version or "uname -a"
    Linux version 2.6.18-92.el5 (brewbuilder@ls20-bc2-13.build.redhat.com) (gcc version
    4.1.2 20071124 (Red Hat 4.1.2-41)) #1 SMP Tue Apr 29 13:16:15 EDT 2008
```

To identify the software version of the Linux OS on the COS node enter the following commands:

```
# cat /proc/version
    Linux version 2.6.18-53.el5.kernel.2_6_18.2008.10.07.01 (arroyoqa@build-svr) (gcc
    version 4.1.2 20070626 (Red Hat 4.1.2-14)) #1 SMP Mon Nov 17 18:21:51 PST 2008
# uname -a
    Linux stm74 2.6.18-53.el5.kernel.2_6_18.2008.10.07.01 #1 SMP Mon Nov 17 18:21:51 PST
    2008 i686 i686 i386 GNU/Linux
```

CSserver Code

To identify the software version of the CServer on the COS node, perform the following steps:

a. On the COS node, enter the **ls -ltr /lib/modules/** to list all modules and identify the directory of the most recent version:

```
# ls -ltr /lib/modules/
    total 24
    drwxr-xr-x. 7 root root 4096 Oct 12 11:37 2.6.32-358.el6.x86_64
    drwxr-xr-x. 2 root root 4096 Oct 12 12:48 2.6.32-3.5.0_cos0.15
    drwxr-xr-x  2 root root 4096 Oct 23 13:48 2.6.32-3.5.0_cos0.16
    drwxr-xr-x  2 root root 4096 Nov  5 11:14 2.6.32-3.5.0_cos0.18
    drwxr-xr-x  3 root root 4096 Nov 24 10:44 2.6.32-3.5.1_cos0.2
    drwxr-xr-x  2 root root 4096 Nov 24 10:44 2.6.32-3.5.0_cos0.19
```

b. Enter the command **cd /lib/modules/***latest_version*, where *latest_version* is the name of the directory listed in Step a with the highest version number.

```
# cd /lib/modules/2.6.32-3.5.1_cos0.2
```

c. Enter the command **strings avs_cserver.ko | grep 'CServer' | egrep 'Release|Information'** to view the software version of the CServer on the COS node.

```
# strings avs_cserver.ko | grep 'CServer' | egrep 'Release|Information'
    CServer Release 3.5.1-0b9 ENV_ISA_SR prod
```

```
CServer Information ENV_ISA_SR prod (cdsbuild@cds-build7) (gcc 4.4.7 20120313 (Red
Hat 4.4.7-11)) 3.5.1-0b9
```

To view the CServer settings, status, and version, enter the following command:

```
# cat /proc/calypso/status/server_settings
    AVS CServer Information ENV_ISA_SR PROD (arroyoqa@build-svr) (gcc version 4.1.2
    20070626 (Red Hat 4.1.2-14))
    #1-Ncserver-e013-2009-01-20-03 Tue Jan 20 17:54:28 PST 2009

    Server Settings:
        Server is operational1
        Cache2App is operational
        TSCs Per Second is 2333447000

    Network Settings:
        Running in L3 Network Mode
        Allow Jumbo Frames
        Transport/Stream Data Payload: 1316
        Cache/Fill Data Payload: 7680
        Cache/Fill Control Maximum Packet Size: 8048
```

Using ifstats to Monitor Traffic

The **ifstats** command shows real-time traffic on each Ethernet interface on the server.

```
# /home/stats/ifstats
    ifstats  -   11:12:22
    ============================================================
    Int#    R-Mbps    X-Mbps           R-Bytes           X-Bytes
     eth0         0         0          56760511         166307653
     eth1         0         0                 0                 0
     eth2         4       457        3439241508        3497139080
     eth3         4       457        3439172148        3099124288
     eth4         4       457        3441836680        2945489644
     eth5         4       472        3443060380        2736115618
     eth6         4       471        3438423816        2613199736
     eth7         5       464        3440066492        2419935662
     eth8         4       449        3439982812        2266582156
     eth9         4       465        3443251384        2164010982
    eth10         5       465        3439982136        1915437726
    eth11         4       464        3438935192         397577442
    eth12         5       464        3440343164         300903930
    eth13         4       465        3439540716        4454799830
```

Disk Drive Issues

The disk drive order is irrelevant when reinserting disk drives after transporting a chassis, or transferring disk drives from one chassis to another.

To view the statistics of the internal boot drive, the disk drive that contains the software, enter the **df -k** command.

```
# df -k
    Filesystem           1k-blocks      Used Available Use% Mounted on
    /dev/hda1             10317828   3764936   6028776  39% /
    /dev/hda2             20641788   1711372  17881776   9% /arroyo
    /dev/hda3              8254272     32828   7802148   1% /arroyo/db
```

```
/dev/hda6             35641880   1185880  32645480   4% /arroyo/log
none                   1681200         0   1681200   0% /dev/shm
```

To view the statistics of a removable SATA or SCSI disk drive, use the following commands:

```
# cat /proc/calypso/status/diskinfo
    Disk Info:
      Disks(12) Op(12)
      Storage: T(804G) A(21%) U(0)
      BW: (99%) w(1.35M/s) r(0/s)
      I/O Util: w(1:0%) e(0) a(0%)
    Disk[ 1][67.0G] A[20%] B[11x]
    Disk[ 2][67.0G] A[20%] B[0x]
    Disk[ 3][67.0G] A[21%] B[0x]
    Disk[ 4][66.5G] A[22%] B[0x]
    Disk[ 5][67.0G] A[20%] B[0x]
    Disk[ 6][67.0G] A[21%] B[0x]
    Disk[ 7][67.0G] A[20%] B[0x]
    Disk[ 8][67.0G] A[20%] B[0x]
    Disk[ 9][67.0G] A[21%] B[0x]
    Disk[10][67.0G] A[20%] B[0x]
    Disk[11][67.0G] A[20%] B[0x]
    Disk[12][67.0G] A[20%] B[0x]
```

Network

The following commands are useful for checking your network configuration and activity.

To view the ARP table, enter the following command:

```
# arp -a
    jetsam.v.com (111.0.110.151) at 00:00:0C:07:AC:00 [ether] on eth0
    COS17-m1.v.com (111.0.210.170) at 00:30:48:58:5B:A1 [ether] on eth0
    COS17-v1.v.com (111.0.210.171) at 00:30:48:31:53:B2 [ether] on eth0
    ? (111.0.210.175) at 00:30:48:32:0A:5A [ether] on eth0
    COS17-s1.v.com (111.0.210.172) at 00:04:23:D8:89:44 [ether] on eth0
    COS17-s1.v.com (111.0.210.172) at 00:04:23:D8:89:44 [ether] on eth0
```

To view the IP routing table, enter the following command:

```
# netstat -rn
    Kernel IP routing table
    Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
    111.0.210.0     0.0.0.0         255.255.255.0   U         0 0          0 eth0
    111.0.0.0       0.0.0.0         255.0.0.0       U         0 0          0 eth0
    127.0.0.0       0.0.0.0         255.0.0.0       U         0 0          0 lo
    0.0.0.0         111.0.210.1     0.0.0.0         UG        0 0          0 eth0
```

To view the COS subnet table, enter the following command:

```
# cat /arroyo/test/SubnetTable
    network 192.169.75.64 netmask 255.255.255.192 gateway 192.169.75.126
```

Note Local networks and their gateways are specified in the SubnetTable file.

To view the COS Remote Server table, enter the following command:

```
# cat /arroyo/test/RemoteServers
    remote server
    id 141
    ip 111.1.9.20
```

```
ip 111.1.9.21
ip 111.1.9.22
ip 111.1.9.23
ip 111.1.9.24
end remote server

remote server
id 143
ip 111.1.9.25
ip 111.1.9.26
end remote server

remote server
id 144
ip 111.1.9.27
ip 111.1.9.28
ip 111.1.9.29
ip 111.1.9.30
end remote server
```

Interface Information

To view basic interface information, use the **ifconfig** command.

```
# ifconfig eth0
    eth0      Link encap:Ethernet  HWaddr 00:04:23:D8:9A:80
              inet addr:111.0.110.41  Bcast:111.0.110.255  Mask:255.255.255.0
              UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
              RX packets:13946269 errors:0 dropped:0 overruns:0 frame:0
              TX packets:11594110 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:3085199261 (2942.2 Mb)  TX bytes:1317620721 (1256.5 Mb)
              Interrupt:24 Base address:0x3000 Memory:dd240000-0
```

To view interface card settings for physical interfaces use the **ethtool** command.

```
# ethtool eth0
    Settings for eth0:
        Supported ports: [ FIBRE ]
        Supported link modes:   10000baseT/Full
        Supported pause frame use: No
        Supports auto-negotiation: No
        Advertised link modes:  10000baseT/Full
        Advertised pause frame use: No
        Advertised auto-negotiation: No
        Speed: 1000Mb/s
        Duplex: Full
        Port: FIBRE
        PHYAD: 0
        Transceiver: external
        Auto-negotiation: off
        Current message level: 0x00000000 (0)

        Link detected: yes
```

For detailed interface information, view the interface information file in the /proc/net/ directory.

```
# cat /proc/net/adapters/eth0.info
    Description                     Cisco Systems IncÂ®  VIC Ethernet Adapter
    Driver_Name                     enic
    Driver_Version                  2.1.1.66
    Adapter_Name                    eth0
```

```
MAC_addr                        E4:AA:5D:AD:65:11

PCI_Vendor                      0x1137
PCI_Device                      0x0043
PCI_Subsystem_Vendor            0x1137
PCI_Subsystem_Device_ID         0x012e
PCI_Bus                         0x06
PCI_Slot                        0

Uplink_Interface                0
Link                            UP
Speed                           1000 Mb/s

Tx_Packets                      32128987
Tx_Unicast_Packets              32125075
Tx_Multicast_Packets            6
Tx_Broadcast_Packets            3906
Tx_Bytes                        4620479646
Tx_Unicast_Bytes                4620229170
Tx_Multicast_Bytes              492
Tx_Broadcast_Bytes              249984
Tx_Errors                       0
Tx_Dropped                      0

Rx_Packets                      39367510
Rx_Packets_Total                39367510
Rx_Unicast_Packets              30780687
Rx_Multicast_Packets            72
Rx_Broadcast_Packets            8586751
Rx_Bytes                        8261544222
Rx_Unicast_Bytes                7651378709
Rx_Multicast_Bytes              6768
Rx_Broadcast_Bytes              610158745
Rx_Errors                       0
Rx_Over_Errors                  0
Rx_CRC_Errors                   0
Rx_Dropped                      0
Rx_No_Bufs                      0
```

Content Processing Issues

Content Mirroring

To enable content mirroring locally on one COS node, do the following:

Step 1 Add the following line to the /arroyo/test/setupfile file:

```
vault local copy count 2
```

Alternatively, enable local mirroring using the tunables. You can also use the tunables to verify the settings.

```
echo 2 > /proc/calypso/tunables/vaultlocalcopycount
```

Note Using the **echo 2** command to enable local mirroring in the tunable file only changes the local copy count temporarily. The local copy count resets to its original value on reboot. To configure the local copy count permanently for any value other than 1, edit the /arroyo/test/setupfile or use the Service Manager GUI.

Note The Service Manager GUI can overwrite manual changes to the /arroyo/test/setupfile file. For settings that you do *not* want the Service Manager to be able to overwrite, add these settings to the /arroyo/test/aftersetupfile file instead. If the aftersetupfile file does not exist, while logged in as root, create a new blank file in the /arroyo/test folder named aftersetupfile. Later, if you want to use the Service Manager GUI to edit the setting, remove that setting from the aftersetupfile file.

To enable content mirroring between two COS nodes, do the following:

Step 1 In the Service Manager, choose **Service Domain Objects > Policies > Asset Redundancy Policies**.

Step 2 Click **+** to add a new policy.

Step 3 In the Name field, enter a name and from the Policy Type drop-down list, choose **Mirroring**.

Step 4 Click **+** to add a new policy rule.

Step 5 From the Match Tag drop-down list, choose **Local** and in the Number of Copies and Keep Count fields enter **2**.

Step 6 From the Trigger drop-down list for the rule, choose **Start** to trigger the rule at the start of the write or choose **Complete** to trigger the rule at the completion of the write.

Step 7 From the State field of the rule, choose **Enabled** to enable the rule.

Step 8 Click **Save** to save the new rule.

Step 9 Click **Save** to save the new policy.

Step 10 Verify the change has propagated by looking at /arroyo/test/setupfile and /arroyo/log/protocoltiming.log.<*date*> files.

```
# grep mirror /arroyo/test/setupfile
    vault mirror copies 2

# grep LocalMirror /arroyo/log/protocoltiming.log.11202007
    -LocalMirror Active=0:0 comp=0% obj=0.0/s read=0b/s write=0b/s copies=1
    -LocalMirror Active=0:0 comp=0% obj=0.0/s read=0b/s write=0b/s copies=1
```

Erasure Coding Troubleshooting

Note The total number of data and parity stripes *cannot* exceed one less than the total number of available servers. This implies that for 1 data and 1 parity stripe, you need a minimum of at least 3 servers. For example, if you have 8 servers, at a maximum you can configure 6 Data and 1 Parity stripes, or 4 Data and 2 Parity stripes.

Watch for warnings in the /arroyo/log/protocoltiming*.log file before issuing any writes. If you do not have enough servers in your cluster, there will be a warning message in the protocoltiming*.log file on the last sample. If you receive this warning message, you must reduce the number of data or parity stripes so there are enough servers present to stripe data to.

Note To verify, you must use the GOID that is associated with a Swift Write object and use the **stripequery** command to /proc with that GOID.

RIO model writes have a max of 32 GB object with a 1 byte minimum and traditional model writes have a max of 512 GB write with a 1 byte minimum.

The following example shows how to check the striping for an RIO model write:

Step 1 Enter the following command to write four copies of 2 Mb object:

```
time curl -v -X PUT -H "X-Rio-CopyCount: 4"
http://192.169.220.25/rio/bucket1/thierryg/2Mx4.ts -T ./2M
```

Step 2 Enter the following command to perform a Distributed Erasure Coding (DEC) stripequery using the Goid returned in Step 1:

```
ssh -o "BatchMode yes" 172.22.125.210 "echo 'stripequery 0x155d3f7dc003' >
/proc/calypso/test/filesystemtestcommand"  2>&1
```

Step 3 Enter the following command to check the file system log for striping:

```
ssh -o "BatchMode yes" 172.22.125.210 "tail -n 30 /arroyo/log/filesystemtest.log.20150819"
2>&1
```

The following example shows how to check the striping for a traditional model write:

Step 1 Create an account:

```
time curl -v -X PUT  -H "X-Auth-Admin-User: .super_admin" -H "X-Auth-Admin-Key: rootroot"
http://192.169.220.2/auth/v2/account90
```

Step 2 Create a user:

```
time curl -v -X PUT -H "X-Auth-Admin-User: .super_admin" -H "X-Auth-Admin-Key: rootroot"
-H "X-Auth-User-Key: rootroot" -H "X-Auth-User-Reseller-Admin: true"
http://192.169.220.2/auth/v2/account90/user90
```

Step 3 Get an authorization token and storage URL:

```
time curl -v -X GET  -H "X-Auth-User: account90:user90"  -H "X-Auth-Key: rootroot"
http://192.169.220.2/v1.0
```

Step 4 Create a container using the token and storage URL returned in Step 3:

```
time curl -v -X PUT -H "X-Auth-Token: AUTH_tkfec0e31bf1514a47bf29dddba697f8a6"
http://192.169.220.2/v1/AUTH_ea79aa8c-8656-4da9-9f8e-a69f49bdaa7f/container90
```

Step 5 Enter the following command to write a 512G object:

```
time curl -v -X PUT -H "X-Auth-Token: AUTH_tkfec0e31bf1514a47bf29dddba697f8a6"
http://192.169.220.2/v1/AUTH_ea79aa8c-8656-4da9-9f8e-a69f49bdaa7f/container90/512G -T 512G
```

Step 6 Get the Goid:

```
time curl -v -I -H "X-Auth-Token: AUTH_tkfec0e31bf1514a47bf29dddba697f8a6"
http://192.169.220.2/v1/AUTH_ea79aa8c-8656-4da9-9f8e-a69f49bdaa7f/container90/512G
```

Step 7 Enter the following command to perform a Distributed Erasure Coding (DEC) stripequery using the Goid returned in Step 1:

```
ssh -o "BatchMode yes" 172.22.125.210 "echo 'stripequery 0x155d3f7dc003' >
/proc/calypso/test/filesystemtestcommand" 2>&1
```

Step 8 Enter the following command to check the file system log for striping:

```
ssh -o "BatchMode yes" 172.22.125.210 "tail -n 30 /arroyo/log/filesystemtest.log.20150819"
2>&1
```

UCS C3160 (Colusa) Troubleshooting Tips

The Cisco Integrated Management Controller (CIMC) is the management service for the Cisco UCS C3160. CIMC runs within the server.

You can use a web-based GUI or SSH-based CLI to access, configure, administer, and monitor the server. Almost all tasks can be performed in either interface. The results of tasks performed in one interface are automatically displayed in another.

This document provides some CIMC troubleshooting tips and common troubleshooting scenarios for the Cisco UCS C3160.

UCS C3160 Troubleshooting Tips

This session provides common troubleshooting tips on the UCS C3160.

Obtaining Show Tech Support to TAC

Perform this task when requested by the Cisco Technical Assistance Center (TAC). This utility creates a summary report containing configuration information, logs, and diagnostic data that will help TAC in troubleshooting and resolving a technical issue.

This show tech support is available from GUI and CLI to tftp upload a tech support file for offline analysis.

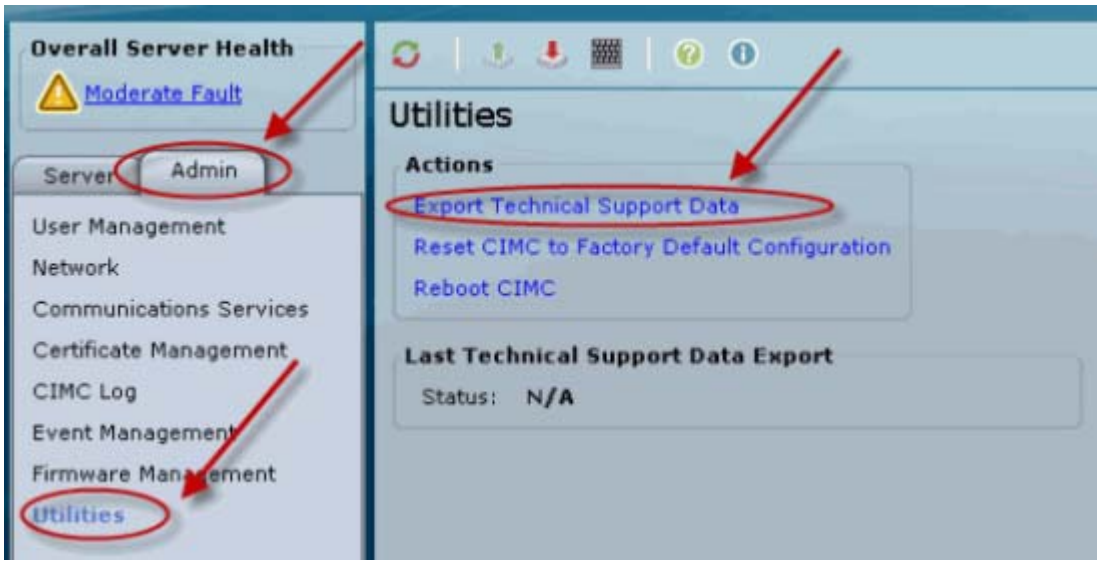
To obtain show tech via GUI, do the following:

Step 1 In the Navigation pane, click the **Admin** tab.

Step 2 From the **Admin** tab, click **Utilities**.

Step 3 In the Actions area of the Utilities pane, click **Export Technical Support Data**.

Figure 2-1 Utilities



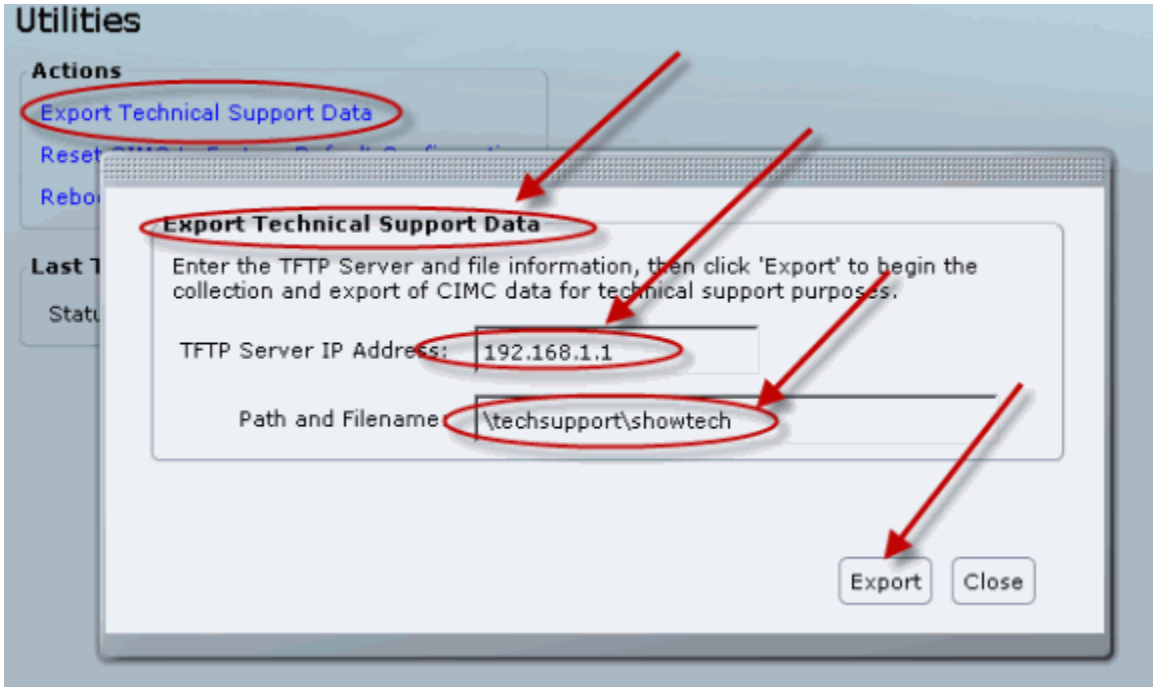
Step 4 In the Export Technical Support Data dialog box, complete these fields:

TFTP Server IP Address field - The IP address of the TFTP server on which the support data file should be stored.

Path and Filename field - The file name in which the support data should be stored on the server. When you enter this name, include the relative path for the file from the top of the TFTP tree to the desired location.

Step 5 Click **Export**.

Figure 2-2 Utilities - Export Technical Support Data



To obtain show tech via CLI command, do the following:

Generate show techsupport then provide the generated report file to Cisco TAC.

SanDiego# scope cimc SanDiego /cimc # scope

firmware log network

tech-support

SanDiego /cimc # scope tech-support

SanDiego /cimc/tech-support # set tftp-ip 192.168.1.1 SanDiego /cimc/tech-support *# set path \techsupport\showtech SanDiego /cimc/tech-support *#commit

SanDiego /cimc/tech-support *#start

Following are the explanations of some of the key fields within the showtech:

- var/ - Contains detailed logs, and status of all monitored services. It also contains services information files such as the configuration of SOL and IPMI sensor alarms.
- var/log - This contains the rolling volatile log messages.
- obfl/ - This contains the rolling non-volatile log messages.
- met/ - Non-volatile configuration and SEL.
- tmp/ - The show techsupport text files, along with BIOS techsupport text files.
- Text files in tmp - These contain all process, network, system, mezzanine, and bios state information.

- mctool - Gets basic information on the State of the CIMC to USC management API.
- network - See current network configuration and socket information.
- obfl - Live obfl
- messages - Live /var/log/messages file
- alarms - What sensors are in alarm.
- sensors - Current sensor readings from IPMI.
- power - The current power state of the x86.

Display of System Event Log Events

To display the System Event Log (SEL) events, do the following:

Step 1 In the Navigation pane, click the **Server** tab.

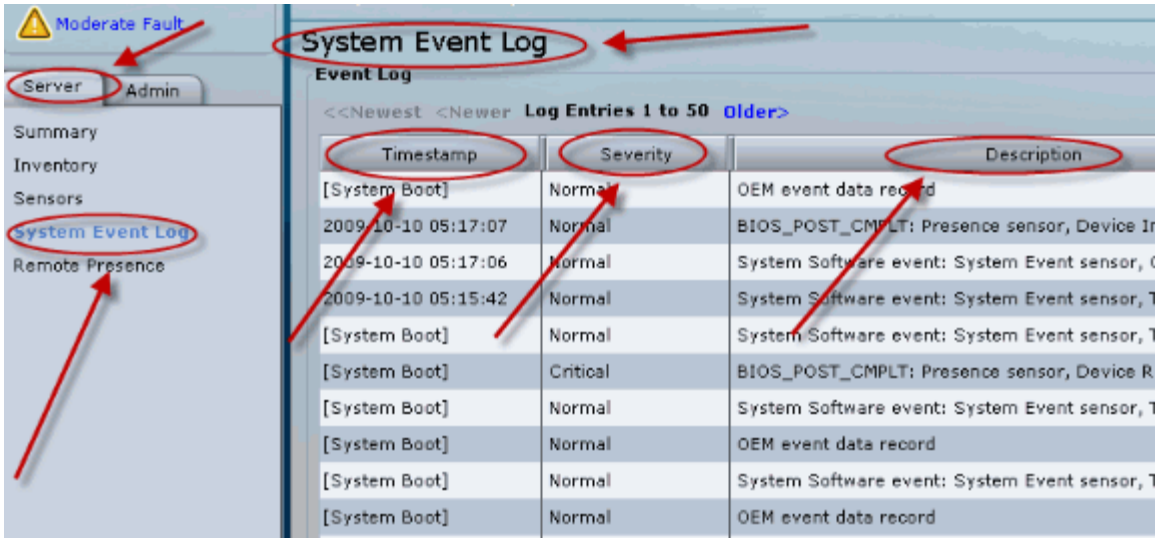
Step 2 From the Server tab, click **System Event Log**.

Step 3 Review the following information for each system event in the log.

Step 4 (Optional) From the Entries Per Page drop?down list, select the number of system events to display on each page.

Step 5 (Optional) Click <Newer and Older> to move backward and forward through the pages of system events, or click <<Newest to move to the top of the list. By default, the newest system events are displayed at the top if the list. Cisco CIMC.

Figure 2-3 System Event Log



Display of Sensor Readings

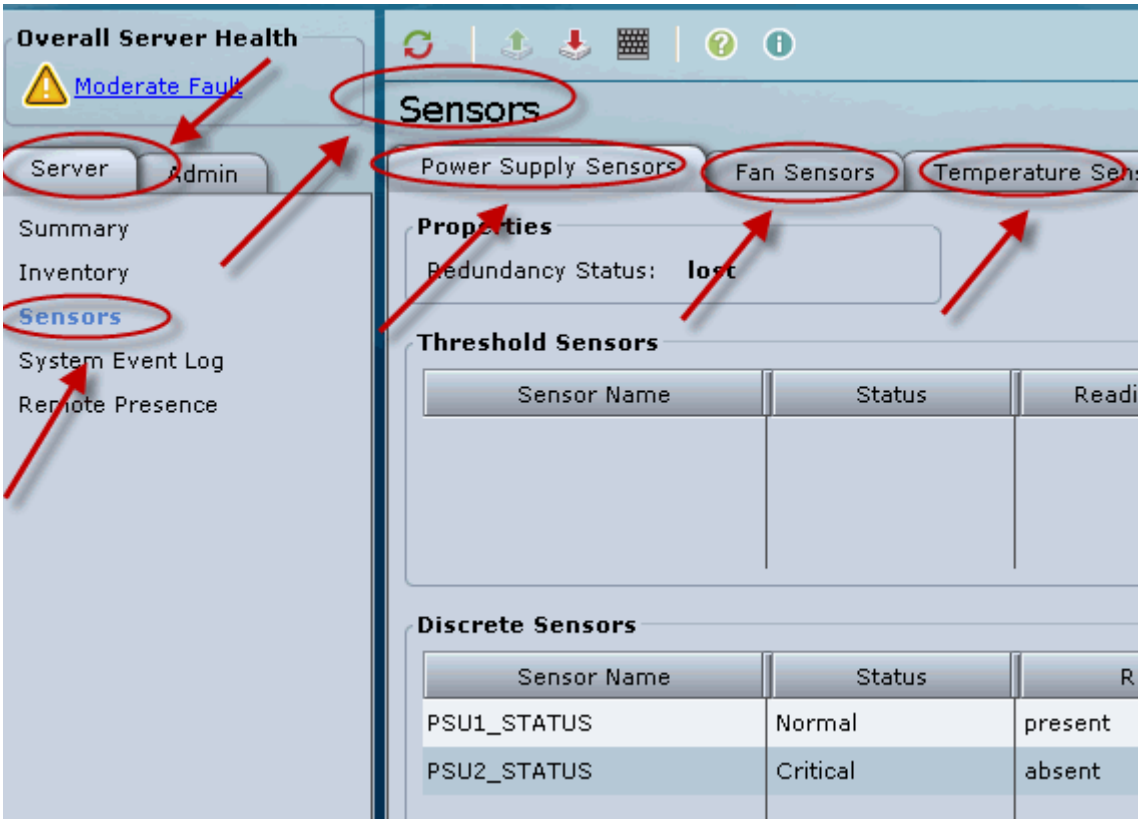
To display the sensor readings, do the following:

Step 1 In the Navigation pane, click the **Server** tab.

Step 2 From the **Server** tab, click **Sensors**.

Step 3 View various sensors by clicking the desired sensor.

Figure 2-4 Sensors



Display of CIMC Log

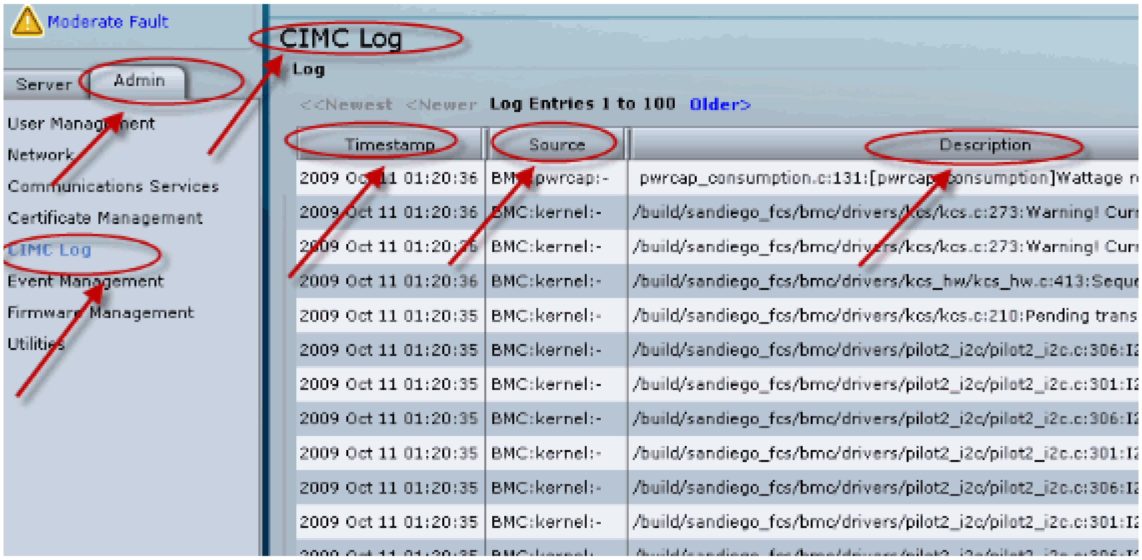
To display the CIMC log, do the following:

Step 1 In the Navigation pane, click the **Admin** tab.

Step 2 From the **Admin** tab, click **CIMC Log**.

Step 3 From the Entries Per Page drop?down list, select the number of CIMC events to display on each page.

Figure 2-5 CIMC Log



Common Troubleshooting Scenarios

Power-On Related

If there is no Standby Power to UCS C250 M1 Extended-Memory Rack-Mount Server, do the following:

Step 1 Check that the AC power cord is ok.

Step 2 Failure in Power Supply Unit.

If the Server Host does not power up, do the following:

Step 1 Check front I/O board connection.

Step 2 Check Power Sequencer fault LEDs.

Step 3 Power Supply unit failure (PS Failure LED blinking).

If the Server powers on with no video, do the following:

Step 1 Check that the front I/O dongle is properly seated.

Step 2 Check the front I/O cable connection to Motherboard.

Step 3 Memory subsystem failure.

Common Troubleshooting Scenarios Host Does not Boot

If the host does not boots up, do the following:

Step 1 Verify front I/O dongle is seated correctly.

Step 2 Check Front I/O cable connection.

Step 3 Reseat/Replace Dimm(s).

Step 4 Verify BIOS is not corrupt.

Step 5 Verify host power rails are good.

Step 6 Check CPU sockets for bent pins.

Step 7 Verify Powerok signals are ok.

Step 8 Verify Resets are good.

Common Troubleshooting Scenarios - BMC

BMC booted. Look for Blade health LED to come on which indicates that the BMC has started.

Step 1 Check that the Standby power rails are ok.

Step 2 Check that the BMC bios is not corrupt.

Step 3 Check that the BMC clock is ok.

Step 4 Check that standby power is ok and resets are valid.

If the BMC Ethernet cannot communicate, do the following:

Step 1 Check the flex cable connections to Mother Board and Rear I/O.

Accessing CIMC on the UCS C3160

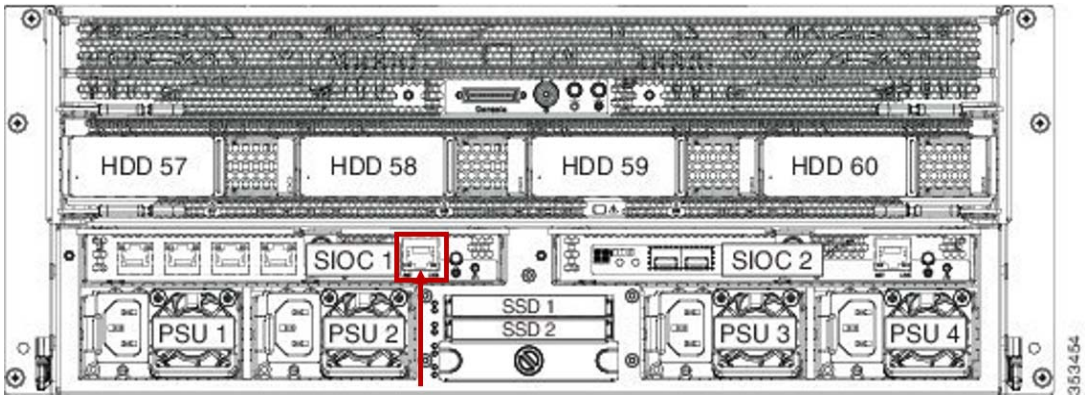
For UCS C3160, you can use the CIMC to manage your device. However, you have to configure the CIMC.

To configure the CIMCIP, do the following:

Step 1 Attach an Ethernet cable to the SIOC 1, 1-Gb Ethernet dedicated management port of the UCS C3160. SIOC 1 is on the left side as you look at the server rear.

Note Use only this SIOC 1 management port; do not use the SIOC 2 management port.

Figure 2-6 Attaching the Ethernet Cable

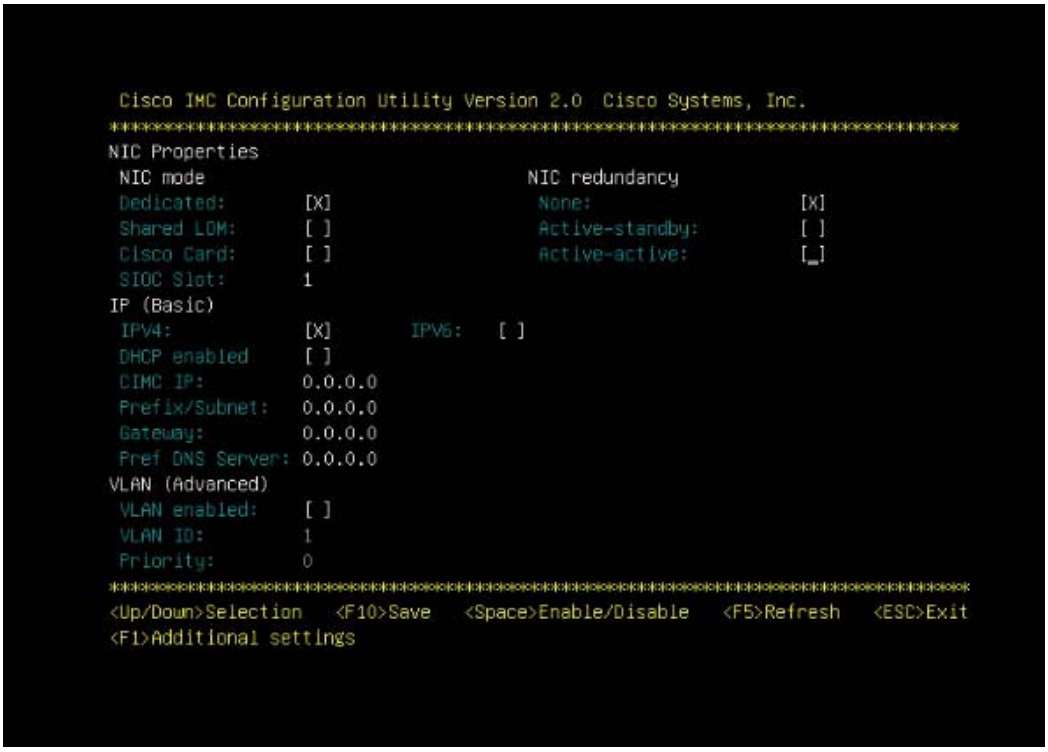


Step 2 Power on the server and press F8 when the CIMC configuration screen appears.

Step 3 Configure the device with the options shown below:

- For the NIC mode, press the Space Bar to choose **Dedicated** if you are using the dedicated IPMI ethernet port.
- Enter the CIMC IP, Prefix/Subnet mask, and gateway addresses.
- For the NIC Redundancy, press the Space Bar to choose **None**.
- Press **F10** to save the configuration.
- Press **Esc** to exit the configuration.

Figure 2-7 Configuring the CIMC IP



Note When configuring CIMC, pay close attention to the 'NIC Mode' and 'NIC Redundancy' settings. If these are set incorrectly, CIMC configures itself to use network ports other than the dedicated CIMC port which will cause issues.

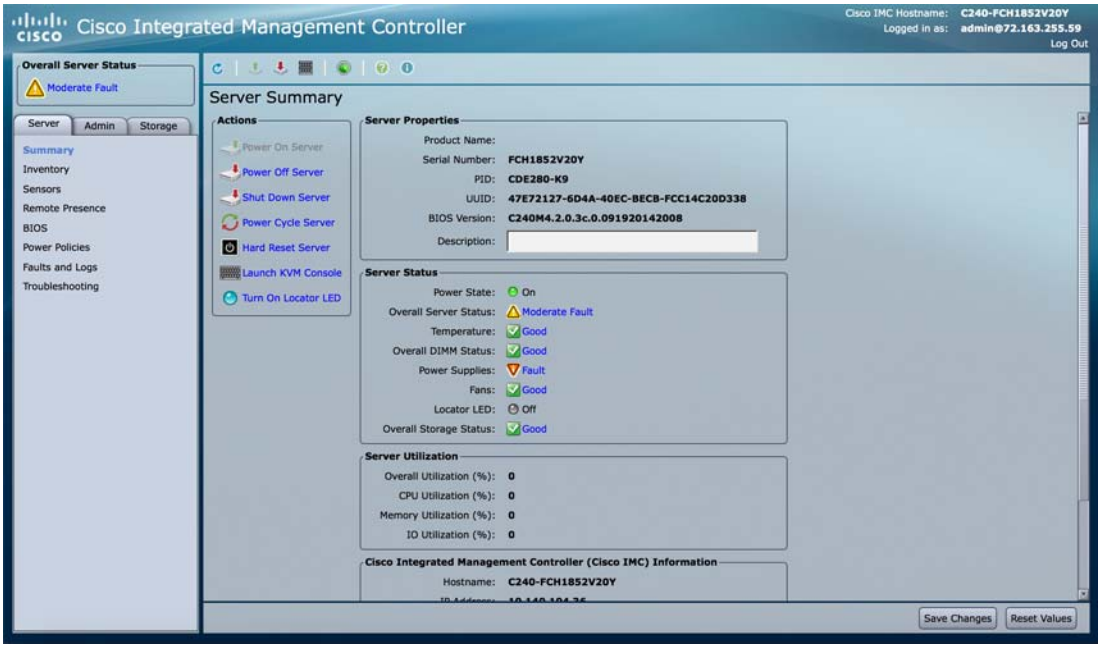
Step 4 Ping the IP address associated with the CIMC port.

Step 5 After verifying the CIMC interface is reachable; verify that the CIMC Web Interface is available by pointing your Web-browser to its IP address:

https://<*CIMC IP*>

Step 6 Enter **admin** as the username and the default password as **password**.

Figure 2-8 CIMC Main Webpage



Connecting to the console

To connect to the console, do the following:

Step 1 Enter the IP address of the server with admin as the username and then the password (same as CIMC login web page):

SE1$ ssh -l admin

10.140.104.17

admin@10.140.104.17's

password: XXXXX C240-

FCH1852V20Y#

Step 2 Enter the following command:

C240-FCH1852V20Y# connect host