Diagnostics Configuration

- Overview of Cisco UCS Manager Diagnostics, on page 1
- Creating a Diagnostics Policy, on page 1
- Configuring a Memory Test for a Diagnostics Policy, on page 2
- Deleting a Diagnostic Policy, on page 4
- Running a Diagnostics Test on a Server, on page 5
- Stopping a Diagnostics Test, on page 5
- Diagnostics Troubleshooting, on page 6

Overview of Cisco UCS Manager Diagnostics

The Cisco UCS Manager diagnostics tool enables you to verify the health of the hardware components on your servers. The diagnostics tool provides a variety of tests to exercise and stress the various hardware subsystems on the servers, such as memory and CPU. You can use the tool to run a sanity check on the state of your servers after you fix or replace a hardware component. You can also use this tool to run comprehensive burn-in tests before you deploy a new server in your production environment.

When a system is new, a default diagnostics policy is created in org scope. This default policy is named default and it cannot be deleted. The user will receive an error message if they try to delete it. The default diagnostic policy is the preferred way to execute the same set of tests across all servers. Any diagnostic policy, including the default can be customized.

The default policy only has one memory test. The default parameters of the memory test can be modified. In addition, the memory test within the default diagnostics policy can be deleted. If it does not have a memory test, the diagnostic policy will not run.

Creating a Diagnostics Policy

Before you begin

You must log in as a user with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope org**	Enters the organization configuration mode.
Step 2	UCS-A /org # **create diag-policy** *<diag-policy>*	Creates a diagnostic policy. **Note** The diagnostic policy name can contain up to 16 characters.
Step 3	UCS-A /org/diag-policy # **commit buffer**	

Example

The following example shows how to create and set description for a diagnostic policy:

```
UCS-A# scope org
UCS-A /org # create diag-policy new-policy
UCS-A /org/diag-policy* # commit-buffer
```

Configuring a Memory Test for a Diagnostics Policy

Before you begin

You must log in as a user with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope org**	Enters the organization configuration mode.
Step 2	UCS-A /org # **create diag-policy-name** *<diag-polic-name>*	Creates a custom diagnostic policy. The diagnostic policy can contain up to 16 characters.
Step 3	UCS-A /org/diag-policy-name* # **commit buffer**	Commits the transaction to the system configuration.
Step 4	UCS-A /org/diag-policy # **create memory-test** *<memory-test <test order>*	Creates a custom memory test for the diagnostic policy. The memory test ID can range from 1 to 64. The memory test has the following values which the user can set:

<table>
<tr><th></th><th>Command or Action</th><th>Purpose</th></tr>
<tr><td></td><td></td><td><table>
<tr><th>Name</th><th>Description</th></tr>
<tr><td>Order</td><td>The order in which the tests will be executed.</td></tr>
<tr><td>CPU Filter</td><td>Sets the CPU filter to all CPUs or to a specified CPU.</td></tr>
<tr><td>Loop Count</td><td>Sets the loop count to the specified iterations. The range is from 1-1000.</td></tr>
<tr><td>Memory Chunk Size</td><td>Sets the memory chunk to 5mb-chunk or big-chunk.</td></tr>
<tr><td>Memory Size</td><td>Sets the memory size to a specific value.</td></tr>
<tr><td>Pattern</td><td>Sets the memory test to butterfly, killer, prbs, prbs-addr, or prbs-killer.</td></tr>
</table></td></tr>
<tr><td>Step 5</td><td>UCS-A /org/diag-policy/memory-test* # set cpu-filter {all cpus | p0-p1-cpus}</td><td>Sets the CPU filter to all CPUs or on the core 0 and 1 CPUs. Values are all cups or p0-p1-cpus.</td></tr>
<tr><td>Step 6</td><td>UCS-A /org/diag-policy/memory-test* # set memchunksize {5mb-chunk | big-chunk}</td><td>Sets the memory chunk size to the specified value in GiB. Values are 5mb-chunk or big-chunk</td></tr>
<tr><td>Step 7</td><td>UCS-A /org/diag-policy/memory-test* # set memsize {0-4096 | all}</td><td>Sets the memory size to the specified value. The available values are 0-4096 or all</td></tr>
<tr><td>Step 8</td><td>UCS-A /org/diag-policy/memory-test* # set pattern {butterfly |killer |prbs |prbs-addr |prbs-killer}</td><td>Sets the memory test to the specified pattern. Available patterns are butterfly, killer, prbs, prbs-addr, or prbs-killer.</td></tr>
<tr><td>Step 9</td><td>UCS-A /org/diag-policy/memory-test* # set loopcount 1-1000</td><td>Sets the loop count to the specified iterations. The loop count can range from 1 to 1000.</td></tr>
<tr><td>Step 10</td><td>UCS-A /org/diag-policy/memory-test* # commit-buffer</td><td>Commits the transaction to the system configuration.</td></tr>
<tr><td>Step 11</td><td>UCS-A /org/diag-policy/memory-test # exit</td><td>Exits from the memory test scope.</td></tr>
<tr><td>Step 12</td><td>UCS-A /org/diag-policy # show configuration</td><td>Displays the configuration values set for the memory test of the custom diagnostic policy.</td></tr>
</table>

Example

The following example shows how to create a memory test for a diagnostic policy:

```
UCS-A# scope org
UCS-A /org # create diag-policy P2
UCS-A /org/diag-policy* # commit-buffer
UCS-A /org/diag-policy # create memory-test 1
UCS-A /org/diag-policy/memory-test* # set cpu-filter all-cpus
UCS-A /org/diag-policy/memory-test* # set memchunksize big-chunk
UCS-A /org/diag-policy/memory-test* # set memsize all
UCS-A /org/diag-policy/memory-test* # set pattern butterfly
UCS-A /org/diag-policy/memory-test* # set loopcount 1000
UCS-A /org/diag-policy/memory-test* # commit-buffer
UCS-A /org/diag-policy/memory-test # exit
UCS-A /org/diag-policy # show configuration
enter diag-policy P2
enter memory-test 1
set cpu-filter all-cpus
set loopcount 1000
set memchunksize big-chunk
set memsize all
set pattern butterfly
exit
set descr ""
set policy-owner local
exit
UCS-A /org/diag-policy #
```

Deleting a Diagnostic Policy

Before you begin

You must log in as a user with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope org**	Enters the organization configuration mode.
Step 2	UCS-A /org # delete **diag-policy** *<diag-policy>*	Deletes the specified diagnostic policy.
Step 3	UCS-A /org* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete a diagnostic policy:

```
UCS-A # scope org
UCS-A /org # delete diag-policy P2
UCS-A /org* # commit-buffer
UCS-A /org #
```

Running a Diagnostics Test on a Server

Before you begin

You must log in with admin priveleges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope server** *chassis-id /server-id*	Enters chassis server scope for the specified server.
Step 2	UCS-A /chassis/server # **scope diag**	Enters the diagnostic mode.
Step 3	UCS-A /chassis/server/diag # **set diag-policy-name** <*diag-policy-name*>	Associates the specified diagnostic policy with the server.
Step 4	UCS-A /chassis/server/diag* # **commit-buffer**	Commits the transaction to the system configuration.
Step 5	UCS-A /chassis/server/diag # **show**	Displays the server diagnostic details.
Step 6	UCS-A /chassis/server/diag # **start**	Runs the diagnostic test on the server.
Step 7	UCS-A /chassis/server/diag* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to run a diagnostic test on server 1/7:

```
UCS-A # scope server 1/7
UCS-A /chassis/server # scope diag
UCS-A /chassis/server/diag # set diag-policy-name P1
UCS-A /chassis/server/diag* # commit-buffer
UCS-A /chassis/server/diag # show
Oper State    Diag Overall Progress    Diag Policy Name
-----------   ----------------------   ----------------
Completed     100                      P1
UCS-A /chassis/server/diag # start
UCS-A /chassis/server/diag* # commit-buffer
UCS-A /chassis/server/diag #
```

Stopping a Diagnostics Test

Before you begin

You must log in as a user with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope server**	Enters the server configuration mode.
Step 2	UCS-A chassis/server # **scope diag**	Enters diagnostics configuration mode.
Step 3	UCS-A chassis/server/diag # **stop**	Stops the diagnostic policy.
Step 4	UCS-A /chassis/server/diag* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to stop a diagnostic policy:

```
UCS-A# scope server 1/2
UCS-A /chassis/server # scope diag
UCS-A /chassis/server/diag # stop
UCS-A /chassis/server/diag* # commit-buffer
```

Diagnostics Troubleshooting

Issue	Steps to Debug
If the BIOS detects a bad DIMM, the DIMM is disabled and is not visible to the Diagnostics operation.	Refer to memory-related faults in addition to the diagnostics operation results.
If the DIMM blacklisting feature is enabled and a DIMM is blacklisted, it is not visible to the Diagnostics operation.	Refer to memory-related faults in addition to the diagnostics operation results.

The Diagnostics operation may not execute successfully, if the server has bad DIMMs which prevent the server from booting.	NA
The Diagnostics operation can fail, if an uncorrectable error causes a server reboot.	NA
A Diagnostics operation failure can occur if there are memory errors that cause the Diagnostics operation to hang.	NA

The Diagnostics operation can be interrupted by external events, such as a managed endpoint failover or a critical UCSM process restart. In these cases, the Diagnostics operation is cancelled and the Memory Tests are marked as failed.	The failure is triggered by external events. Retry the Diagnostics operation.
A Memory test fails with the error: Uncorrectable errors detected.	Check for server faults under the `Chassis/Server/Faults` tab. See the SEL logs for the DIMM errors under the `Chassis/Server/SEL Logs` tab.
A Memory test failure needs further analysis.	See the diagnostics operation logs in following log file archive on the primary FI in the /workspace partiti `diagnostics/diag_log_<system-name>_<timestamp>_<chassis-id>_<blade-id>.t` See the analysis file: `tmp/ServerDiags/MemoryPmem2.<id>/MemoryPmem2.analysis` in t previously mentioned log file archive. Use the following command to find the diagnostics logs with the analysis files: `# for file in `ls /workspace/diagnostics/*diag*`; do tar -tzvf $file | grep analysis && echo "IN " $file; done`