

How_to_debug_the_root_cause_for_high_CPU_usage

Unified CCX is not a CPU intensive application. Apart from a few background threads, most of the processing in the CCX processes happen only during the execution of an application workflow (call). Then why is the CPU usage of a CCX service high?

Probable Reasons

- High call load.
- Application thread looping continuously.
- Application thread busy in processing some spurious events.

Where to Check?

- Login into RTMT and browse 'Server -> Process -> Sort by %CPU'.
- Find the culprit process.

How to find the root cause?

- **Find the process/processes that has high %CPU.**
 - ◆ Through RTMT
 - ◇ Server -> Process -> Sort by %CPU.
 - ◇ Find the processId (PID) of the top most entries.
 - ◆ Through CLI
 - ◇ Run 'show process using-most cpu'.
 - ◇ Find the processId (PID) of the processes listed.

```
PCPU PID CPU NICE STATE CPUTIME  ARGS
4.1 24087 - 0 S 05:47:41 LRMServer -s CadSplkStd -l CADLRMServer
74.8 23884 - 0 S 4-08:02:12 /opt/cisco/uccx/bin/UCCX_Engine /opt/cisco/uccx/conf/UCCX_Engin
```

- **Find the thread/threads in the process that has high %CPU.**
 - ◆ Through CLI
 - ◇ Run 'show process pid <pid extracted above>'.
 - Find the entry (thread) that has high %CPU.
 - Entries are not sorted based upon %CPU. Need to figure out manually.
 - Extract the thread id (TID) of the thread.
 - ◆ Through root account (remote account)
 - ◇ Run 'top -b -n1 -H -p <pid extracted above>'.
 - Find the entry (thread) that has high %CPU - listed at the top.
 - Extract the thread id (PID) of the thread.

```
PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM  TIME+  COMMAND
32010 uccxuser  16   0   680m 438m 16m  S  15.4  5.4  814:41.05 UCCX_Engine
1305  uccxuser  16   0   680m 438m 16m  S  13.5  5.4  733:38.05 UCCX_Engine
1341  uccxuser  16   0   680m 438m 16m  S  13.5  5.4  623:30.13 UCCX_Engine
942   uccxuser  16   0   680m 438m 16m  S   3.9  5.4  326:40.52 UCCX_Engine
```

- - ◆ Convert the thread id (decimal) to the corresponding hexadecimal value.
 - ◇ Use decimal to hexadecimal converters.
 - ◇ Thread dumps have threads listed with hexadecimal thread ids. So this mapping is required.

How_to_debug_the_root_cause_for_high_CPU_usage

- **Generate a thread dump.**

- ◆ If the process is a Java process (most of the UCCX processes are Java processes - UCCX_Engine, UCCX_Cvd, tomcat, etc)
 - ◇ Login into CCX Serviceability Admin. Browse 'Tools -> Performance Configuration and Logging -> Select Server and Service -> Dump Thread trace'.
 - ◇ The thread dump gets generated in JVM.log of the respective service log directory.
 - For tomcat, it's logged in catalina.out.

- **Open the thread dump, and search for the TID (hexadecimal value) extracted above.**

```
MIVR_ICD_CTI_CLIENTPOOL-72-451-client_thread_452; daemon prio=10 tid=0xa01a4000 nid=0x7d0a runn
java.lang.Thread.State: RUNNABLE
```

- Debug the application logs and the code to find out why this thread is taking more CPU cycles.