Oracle Gateway for msql showing high %system CPU usage for the "grid" user

On one of our 3-node RAC Exadata systems, after configuring monitoring using Netdata, Prometheus and Grafana, I noticed that one of the nodes had a much higher CPU %System time than the other two nodes. This %system time was attributed to the Grid user.

Running a *top -u grid* gave the top CPU-using processes owned by the *grid* user.

This identified a few *dg4msql** processes with the highest CPU usage.

```
49362 grid      20   0  143m  11m 8820 R 98.9  0.0  19605:27 dg4msqlSO
(LOCAL=NO)
11804 grid      20   0  177m  13m 5644 R 96.2  0.0  50823:58 dg4msqlobdr
(LOCAL=NO)
21091 grid      20   0  178m  15m 5648 R 95.9  0.0  43935:17 dg4msqlobdr
(LOCAL=NO)
23286 grid      20   0  394m  81m  10m S  6.6  0.0   0:00.20 dg4odbctelkomqa
(LOCAL=NO)
```

```
[root@node3]# locate msqlSO
/u01/app/oracle/product/gateways/11.2.0.4/gt_1/dg4msql/admin/initSO.ora

[root@neiwdb03 ~] strace -f -p 49362
[pid 49362] sched_yield()               = 0
[pid 49362] sched_yield()               = 0
[pid 49362] sched_yield()               = 0
[pid 49362] sched_yield()               = 0
[pid 49362] sched_yield()               = 0
[pid 49362] sched_yield()               = 0
[pid 49362] sched_yield()               = 0
[pid 49362] sched_yield()               = 0
[pid 49362] sched_yield()               = 0
[pid 49362] sched_yield()               = 0
^CProcess 49362 detached
Process 49363 detached

//Clearly waiting for something

[root@neiwdb03 ~]# pstack 49362
Thread 2 (Thread 0x7fe5c5110700 (LWP 49363)):
#0  0x00007fe5c8f44334 in __lll_lock_wait () from /lib64/libpthread.so.0
#1  0x00007fe5c8f3f5d8 in _L_lock_854 () from /lib64/libpthread.so.0
#2  0x00007fe5c8f3f4a7 in pthread_mutex_lock () from /lib64/libpthread.so.0
#3  0x00007fe5c85c4b83 in sltsmna () from
/u01/app/oracle/product/gateways/11.2.0.4/gt_1/lib/libclntsh.so.11.1
#4  0x00007fe5c923586b in holtmxa () from
/u01/app/oracle/product/gateways/11.2.0.4/gt_1/lib/libagtsh.so
#5  0x00007fe5c924606c in hotkpep_PFEntryPoint () from
/u01/app/oracle/product/gateways/11.2.0.4/gt_1/lib/libagtsh.so
#6  0x00007fe5c8f3daa1 in start_thread () from /lib64/libpthread.so.0
#7  0x00007fe5c5b7fbcd in clone () from /lib64/libc.so.6
```

```
Thread 1 (Thread 0x7fe5c9718700 (LWP 49362)):
#0  0x00007fe5c5b664a7 in sched_yield () from /lib64/libc.so.6
#1  0x00007fe5c7c051bd in sltstyield () from
/u01/app/oracle/product/gateways/11.2.0.4/gt_1/lib/libclntsh.so.11.1
#2  0x00007fe5c923963f in homtwfr_WaitForRPCs () from
/u01/app/oracle/product/gateways/11.2.0.4/gt_1/lib/libagtsh.so
#3  0x0000000000403fc6 in shorm ()
#4  0x0000000000403e60 in main ()

//Both threads in a wait state, but spinning on the CPU
```

So the gateway processes is seemingly waiting, albeit very enthusiastically.

Searching this stack output pinged in My Oracle Support Knowledge Base with the following information:

Gateway parallel prefetching may be causing the problem.

To fix, edit gateway *init<sid>.ora* and set the following parameter:

```
_HS_RPC_PREFETCH=off
```

as per Doc ID 1277997.1

To modify logging level for the gateway, add the following:

```
HS_FDS_TRACE_LEVEL=OFF
 OR
HS_FDS_TRACE_LEVEL=255 or DEBUG
HS_FDS_TRACE_FILE_NAME=/tmp/hsodbc.log
```

Apparently the issue can also present when using Oracle 12.1.0.2 database to SQL Server 2014 through Gateway 12.1.0.2: (as per Doc ID 2232968.1)

This is the *init<sid>.ora* fix for that:

```
_HS_RPC_PREFETCH = off
HS_FDS_WORKAROUNDS=16
HS_FDS_DELAYED_OPEN=FALSE
```

From:
https://wiki.dewberry.co.za/ - **Shaun's Wiki**

Permanent link:
https://wiki.dewberry.co.za/doku.php?id=rac_node_grid_system_cpu&rev=1574343072

Last update: **2019/11/21 13:31**