以下内容较长，请洗脸保持清醒 #\_#

工作中时常遇到以下的错误

Oracle 10g：

WARNING: clssnmPollingThread: node <node\_name> at 50% **heartbeat fatal**, eviction in 14.494 seconds

…

[    CSSD][1111296320](:CSSNM00008:)clssnmCheckDskInfo: Aborting local node **to avoid** **splitbrain**. **Cohort of 1** **nodes** with leader 1, sprora01, **is smaller than** **cohort of 3 nodes** led by node 2, sprora02, based on map type 2

Oracle 11g

[cssd(11011) ]CRS-1612:Network communication with node node1 (1) missing for 50% of timeout interval.  **Removal** of this node from cluster in 14.510 seconds

ClssnmPollingThread： CSSD进程的NM模块提示节点X网络心跳延迟超过50%，将在14s内被驱逐，

clssnmCheckDskInfo ：CSSD进程的NM模块检查voting DISK信息后，为避免脑裂关闭本节点，因为 leader 1 为节点sprora01的群组只有一个节点，leader2为节点sprora02的群组有2个节点，所以驱逐smaller的群组

节点驱逐是RAC环境中经常遇见的一个现象，那如何解释节点驱逐呢，Oracle为什么要这么做，由哪个后台进程完成，有哪些触发条件（阈值）？

这该从何说起呢？

…

**1）** **背景**

往上追溯到RAC架构，RAC在集群层将多个实例聚合成一个，能够实现多个节点的实例协同工作，高可用和负载 美丽的背后，并不是那么一番风顺，不可避免的会遇到一些问题

1 并发控制：多个instance同时请求更改database中同一条数据怎么办？

2 健忘：节点1关闭，节点2修改了集群配置，节点1启动后仍然读取了旧信息，配置丢失怎么办？

3 脑裂：正常情况下，各个节点通过心跳了解彼此的工作情况，心跳网络出现故障，那每个节点都认为自己是集群中的“幸存者“，想获得存储数据的控制权，都去更改数据，造成数据灾难怎么办？

4 IO隔离：上一个问题的延伸，脑裂后 虽然确定了谁不是集群的控制者，但这个节点仍然在运行，通过此节点去更改共享数据怎么办？

Oracle RAC最终解决方案

1 并发控制    --利用DLM(distribute lock management)分布式锁控制并发

2 健忘    --OCR

3 脑裂    --voting disk

4 IO隔离 --重启故障节点

参考：

大话Oracle RAC 集群 高可用 备份与恢复  --张晓明编著

本文仅对节点驱逐相关的脑裂和IO隔离做讨论，先了解几个概念

》脑裂：

脑裂代表什么意思？

脑裂意味着有两个或多个单独的节点组（群组），分裂的群组之间没有通信

例如：

假设下列场景中有4个节点分别被命名为A,B,C,D

\*节点A,B可以相互通信；节点C,D可以相互通信

\*但是A和B，不能和C或者D通信，反之亦然

然后会分裂为2个群组{A,B}和{C,D}

上文中说过，脑裂会造成数据灾难，必须被解决，Oracle集群通过终止节点数少的那个群组（SMALLER cohort）的所有节点来控制脑裂，如果两个群组有相同的节点数，包含最低节点号的群组会幸存。就是说集群cluster确认节点数最多的群组(LARGEST cohort)，然后终止不属于这个群组cohort的所有节点。

》理解群组（cohort）信息

在脑裂场景中，每个节点的ocssd都会将其能够通信的一组节点信息记录到仲裁盘（voting disk）每一组节点称之为群组cohort

参考mos

Oracle Grid Infrastructure: Understanding Split-Brain Node Eviction (Doc ID 1546004.1)

》Voteing disk

脑裂的解决方式已经知道了，那他是怎么实现的呢，这里就要提到投票盘voting disk，Oracle进程通过读取voting disk信息，确认哪个群组是LARGEST cohort

》IO隔离

IO fencing实现有硬件和软件方式

硬件方式

       锁住 节点数多的群组（LARGEST cohort）存储，不共享 --有些存储能够实现这种方式

关闭 节点数少的群组（SMALLER cohort）机器电源

软件方式

重启 节点数少的群组（SMALLER cohort）操作系统  —oracle采用这种方式

了解了节点驱逐的背景后，下面看一下节点驱逐是如何实现的

**2）** **节点驱逐的实现**

首先看一下Oracle集群提供的服务

Oracle cluster是一个部署在操作系统之上的软件，提供了很多功能，其中有一个叫CSS服务

**Cluster Synchronization Services (CSS)**: Manages the cluster configuration by controlling which nodes are members of the cluster and by notifying members when a node joins or leaves the cluster. If you are using certified third-party clusterware, then CSS processes interface with your clusterware to manage node membership information.

CSS：控制哪些节点是群集的成员, 以及在节点加入或离开群集时通知成员, 管理群集配置

软件功能必定也是通过进程完成。上图可知，CSS服务由ocssd.bin,cssdmonitor,cssdagent 3个后台进程组成，最重要的是ocssd.bin进程

参考docs：

[https://docs.oracle.com/cd/E11882\_01/rac.112/e41959/intro.htm#CWADD90956](https://docs.oracle.com/cd/E11882_01/rac.112/e41959/intro.htm%20/l%20CWADD90956)

ocssd : 它的主要功能是节点监控（Node Monitoring）和组管理(Group Management),它是CRS的核心进程之一。节点监控是指监控集群中节点的健康，监控的方法是通过网络心跳(network heartbeat)和磁盘心跳（disk heartbeat）实现的，如果集群中的节点连续丢失磁盘心跳或网络心跳，该节点就会被从集群中驱逐，也就是节点重启。

Node Monitoring

网络心跳（network heartbeat）:ocssd.bin进程每秒钟向集群中的各个节点通过私网发送网络心跳信息，以确认各个节点是否正常。如果某个节点连续丢失网络心跳达到阀值，misscount（默认为30秒，如果存在其他集群管理软件则为600秒），集群会通过表决盘进行投票（只有网络心跳是不够的，因为，一旦出现了网络问题，当节点间互相无法发现对方的网络心跳时，无法判断那一个（些）节点的状态正常，哪一些不正常，所以，我们还需要磁盘心跳。），使丢失网络心跳的节点被主节点驱逐出集群，即节点重启。如果集群只包含2个节点，则会出现脑裂，结果是节点号小的节点存活下来，即使是节点号小的节点 存在网络问题。

磁盘心跳（disk heartbeat）：ocssd.bin进程每秒钟都会向所有表决盘（Voting File）注册本节点的状态信息，这个过程叫做磁盘心跳。如果某个节点连续丢失磁盘心跳达到阀值，disk timeout(一般为200秒)，则该节点会自动重启以保证集群的一致性。另外，CRS只要求[N/2]+1个表决盘可用即可，其中N为表决盘数量，一般为奇数。

网络心跳的延迟称之为misscount--MC

磁盘心跳的延迟称之为disk timeout—DIO/IOT

集群节点重新配置：ocssd.bin会每隔一段时间都对集群中的每个节点的最后一次网络心跳和磁盘心跳进行判断，以决定是否需要对集群进行重新配置（也就是说，是否有某一个/些节点需要被evict掉）。当然，当一个节点加入集群时，集群的其他节点需要收到集群重新配置的信息，得到最新的节点成员列表。同样的道理，当集群的某一个节点离开集群时，其他节点需要知道最新的成员列表。当然，无论是节点加入或者离开集群，都会有一个RMN(reconfig)节点负责集群的重新配置，并通知知集群中的其他节点。

由上可知，最基本的节点驱逐是由ocssd进程的NM监控network heartbeat和disk heartbeat，当这两种心跳丢失超过misscount时触发驱逐协议(eviction protocol)

对于ocssd进程的组管理（Group Management）功能这里简单描述一下

在一个集群中，除了节点之外，还有很多资源是需要管理的，（当然，主要的资源管理工作是由crsd.bin来完成的）而这些资源会作为一个一个的组（在css的世界里，我们把组称为grock）注册到css上。

于NM类似，当组中的某一个成员离开或者加入组的时候，也会有一个master节点（一般称这个节点为GM master）来完成组成员的重新配置。

**3）阈值** **(MC/DIO/IOT)**

那么这个阈值到底是多大呢，如果因为某些特殊需求，是否可以更改呢？

如：心跳线插拔，希望在可延迟时间内完成，而不造成节点驱逐

答案是可以更改

网络心跳（network heartbeat）阈值

不同平台与使用oracle cluster与在不同oracle版本下的默认最大网络延时

|  |  |  |
| --- | --- | --- |
| **OS** | **10g (R1 &R2)** | **11g** |
| Linux | 60 | 30 |
| Unix | 30 | 30 |
| VMS | 30 | 30 |
| Windows | 30 | 30 |

使用第三方集群软件的最大网络延迟时间

      使用第三方集群软件，默认为600秒的延迟时间，并且，决定脑裂的工作由第三方集群软件所决定

    对于在AIX平台下，数据库版本为10.2.0.3的环境下，网络超时时间却只有30秒

磁盘心跳（disk heartbeat）阈值

|  |  |
| --- | --- |
| 9.x.x.x | NOTE, MISSCOUNT WAS A  DIFFERENT ENTITY IN THIS RELEASE |
| 10.1.0.2 | No one should be on this version |
| 10.1.0.3 | DTO = MC - 15 seconds |
| 10.1.0.4 | DTO = MC - 15 seconds |
| 10.1.0.4+Unpublished Bug 3306964 | DTO = MC - 3 seconds |
| 10.1.0.4 with CRS II Merge patch | DTO =Disktimeout (Defaults to 200 seconds) Normally OR Misscount seconds only during initial Cluster formation or Slightly before reconfiguration |
| 10.1.0.5 | IOT = MC - 3 seconds |
| 10.2.0.1 +Fix for unpublished Bug 4896338 | IOT=Disktimeout (Defaults to 200 seconds) Normally OR Misscount seconds only during initial Cluster formation or Slightly before reconfiguration |
| 10.2.0.2 | Same as above (10.2.0.1 with Patch [Bug:4896338](https://support.oracle.com/epmos/faces/BugDisplay?parent=DOCUMENT&sourceId=294430.1&id=4896338) |
| 10.1 - 11.2 | During node join and leave (reconfiguration) in a cluster we need to reconfigure, in that particular case we use Short Disk TimeOut (SDTO) which is in all versions SDTO = MC – reboottime (usually 3 seconds) |

查看当前阈值

[grid@rac1 ~]$ crsctl get css misscount

CRS-4678: Successful get misscount 30 for Cluster Synchronization Services.

[grid@rac1 ~]$ crsctl get css disktimeout

CRS-4678: Successful get disktimeout 200 for Cluster Synchronization Services.

[grid@rac1 ~]$ crsctl get css reboottime

CRS-4678: Successful get reboottime 3 for Cluster Synchronization Services.

修改当前阈值

$CRS\_HOME/bin/crsctl set css misscount <n>    #### where <n> is the maximum private network latency in seconds

$CRS\_HOME/bin/crsctl set css reboottime <r> [-force]  #### (<r> is seconds)

$CRS\_HOME/bin/crsctl set css disktimeout <d> [-force] #### (<d> is seconds)

**4）节点驱逐场景**

节点驱逐除了上面最基本的情况外还有一些场景，下面统计一下ai

1 丢失网络心跳超过阈值

Cssd 日志

[ CSSD]2012-03-02 23:56:18.749 [3086] >WARNING: clssnmPollingThread: node <node\_name> at 50% heartbeat fatal, eviction in 14.494 seconds

2 丢失磁盘心跳超过阈值  --磁盘心跳保证解决脑裂的最基本组件，磁盘心跳自身出问题会导致节点驱逐

Cssd日志

[    CSSD][4110736288](:CSSNM00058: )clssnmvDiskCheck: No I/O completions for 200880 ms for voting file /dev/sdb8)

以上两种是最基本的驱逐情况，都是由ocssd进程发起，如果ocssd进程自身出问题呢？

3 oclsomon：这个进程负责监控ocssd是否挂起，如果发现ocssd.bin存在性能问题，则重启该节点

<crs主目录>/log/<节点名称>/cssd/oclsomon/oclsomon.log

如：操作系统资源争用（如CPU）

4 oprocd：这个进程只在Linux和Unix系统，并且第三方集群管理软件未安装的情况下才会出现。如果它发现节点挂起,则重启该节点。

oprocd日志

oprocd.log(/etc/oracle/oprocd/\*.log.\* 或 /var/opt/oracle/oprocd/\*.log.\*)

Dec 21 16:15:30.369857 | LASTGASP | AlarmHandler:  timeout(2312 msec) exceeds interval(1000 msec)+margin(500 msec).   Rebooting NOW

**5）11gR2的重要的新特性—rebootless restart**

这个新特性是在版本11.2.0.2被介绍的。从这个版本开始,当集群中的某个节点被驱逐（例如丢失网络心跳）或者该节点的 ocssd.bin出现问题时，集群将不会直接重新启动该节点，而是首先尝试重新启动GI stack来解决问题，如果GI stack不能够在指定的时间内（short disk I/O timeout）完成graceful shutdown，才会重新启动节点，之后，存活的节点对集群进行重新配置。

SDTO = MC – reboottime (usually 3 seconds)

Cssdagent.bin/Cssdmonitor.bin： 这2个进程是11gR2新增加的。他们的工作主要是同ocssd.bin进行本地心跳（Local HeartBeat），默认情况下每1秒钟一次。本地心跳的作用是监控本地的ocssd.bin进程是否正常运行。同时，他们也监控自己到 ocssd.bin之间的连接。所以，我们可以认为，这两个进程的出现代替了10g中的oclsomon/oclsvmon（如果第三方集群管理软件存 在）和oprocd。

这就是为什么11g，节点驱逐不会立刻重启服务器，而是等待一段时间，如果不能完成GI重启，才会重启服务器！

参考oracle blog

<https://blogs.oracle.com/database4cn/>如何诊断节点重启问题

参考mos

Pre-11.2: Using Diagwait as a diagnostic to get more information for diagnosing Oracle Clusterware Node evictions (Doc ID 559365.1)

参考blog

<https://www.cnblogs.com/macleanoracle/archive/2013/03/19/2967737.html>

**6）日志**

进程日志所在路径

Cluster Synchronization Services (CSS)      Grid\_home/log/host\_name/cssd

Cluster Time Synchronization Service (CTSS)    Grid\_home/log/host\_name/ctssd

集群日志结构参考docs

[https://docs.oracle.com/cd/E11882\_01/rac.112/e41959/troubleshoot.htm#CWADD91091](https://docs.oracle.com/cd/E11882_01/rac.112/e41959/troubleshoot.htm%20/l%20CWADD91091)

**7） 思考：**

当单个或小于(N/2+1)个的voting disk故障时，这种故障可能是由于短期内节点访问voting disk发生I/O error错误而引起的，此时css会立刻将这些失败的voting disk标记为OFFLINE。虽然有一定数量的voting disk OFFLINE了，但是我们仍有至少(N/2+1)个投票磁盘可用，这保证了eviction protocol不会被调用，所以没有节点会被reboot重启

那么这个 (N/2+1) 怎么计算呢，会有哪些情况呢？

我们可以使用2或4个vote disk吗？

Answer:可以的。

但是2、4这样的数目在“至少(N/2+1)个投票磁盘要被节点正常访问”这一disk heartbeat的硬性算法下是不利的：

当我们使用2个vote disk 时，不能发生任意个vote disk的心跳失败

当我们使用3个vote disk 时，不能发生大于1个的vote disk心跳失败

当我们使用4个vote disk 时，不能发生大于1个的vote disk心跳失败 ，这和3个时的容错率是一样，但是因为我们有更多的vote disk，这会导致管理成本和引入的风险增长

当我们使用5个vote disk 时，不能发生大于2个的vote disk心跳失败

当我们使用6个vote disk 时，仍然不能发生大于2个的vote disk心跳失败， 同样的因为比5时多出一个 ，也会引入不合理的管理成本和风险

Reference

Oracle Grid Infrastructure: Understanding Split-Brain Node Eviction (Doc ID 1546004.1)

Oracle blog：css 功能介绍

<https://blogs.oracle.com/database4cn/css>