

ipcs、ipcrm、sysresv、 kernel.shmmax

姓名：小麦苗

时间：2017.11.14

QQ 群：230161599

微信公众号：xiaomaimiaolhr

博客地址：<http://blog.itpub.net/26736162/abstract/1/>



小麦苗

ipcs、ipcrm、sysresv、kernel.shmmax

1.1 BLOG 文档结构图



1.2 前言部分

1.2.1 导读和注意事项

各位技术爱好者，看完本文后，你可以掌握如下的技能，也可以学到一些其它你所不知道的知识，~o(n_n)o~:

- ① ipcs 的使用
- ② ipcrm 释放 oracle 内存段
- ③ sysresv 的使用
- ④ 内核参数 kernel.shmmax
- ⑤ 如何快速的清理 Oracle 的进程
- ⑥ 其它维护操作

Tips:

- ① 本文在 itpub (<http://blog.itpub.net/26736162>)、博客园 (<http://www.cnblogs.com/lhrbest>) 和微信公众号 (xiaomaimiaolhr) 上有同步更新。
- ② 文章中用到的所有代码、相关软件、相关资料及本文的 pdf 版本都请前往小麦苗的云盘下载, 小麦苗的云盘地址见: <http://blog.itpub.net/26736162/viewspace-1624453/>。
- ③ 若网页文章代码格式有错乱, 请下载 pdf 格式的文档来阅读。
- ④ 在本篇 BLOG 中, 代码输出部分一般放在一行一列的表格中。
- ⑤ 本文适合于初中级人员阅读, 数据库大师请略过本文。
- ⑥ 不喜勿喷。

本文若有错误或不完善的地方请大家多多指正, 您的批评指正是我写作的最大动力。

1.3 本文简介

最近有朋友因为 kernel.shmmax 内核参数的问题导致数据库不能启动。小麦苗之前碰到过一次, 只是没有记录下来, 而且以前安装数据库的时候也没有详细介绍这几个参数的含义, 趁这次机会就把这个参数在详细介绍一下吧。

1.4 相关文章链接

- ① 【故障解决】IPCS 和 IPCRM 使用: <http://blog.itpub.net/26736162/viewspace-2112518>
- ② ORACLE 内核参数: <http://blog.itpub.net/26736162/viewspace-2112447/>
- ③ sysresv: <http://blog.itpub.net/26736162/viewspace-2112443/>
- ④ 视频讲解 IPCS 和 IPCRM 使用: http://www.iqiyi.com/w_19rs33qqsp.html
- ⑤ 有关“TNS-12518: TNS:listener could not hand off client connection”的更多内容请参考: 【故障|监听】TNS-12518 、 TNS-00517 和 Linux Error : 32 : Broken pipe : <http://blog.itpub.net/26736162/viewspace-2135468/>

第 2 章 ipcs/ipcrm 命令

更多内容请参考: <http://blog.itpub.net/26736162/viewspace-2112518>

unix/linux 下的共享内存、信号量、队列信息管理

在 Unix 或 Linux 下, 经常有因为共享内存、信号量, 队列等共享信息没有干净地清除而引起一些问题。

查看共享内存的命令是: `ipcs [-m|-s|-q]`。若 `ipcs` 命令不带参数, 则默认会列出共享内存、信号量, 队列信息, 而 `-m` 列出共享内存, `-s` 列出共享信号量, `-q` 列出共享队列。

清除命令是: `ipcrm [-m|-s|-q] id`, 其中, `-m` 删除共享内存, `-s` 删除共享信号量, `-q` 删除共享队列。

```
[oracle@rhel6lhr ~]$ ipcs -h
ipcs provides information on ipc facilities for which you have read access.
Resource Specification:
    -m : shared_mem
    -q : messages
```

```

-s : semaphores
-a : all (default)
Output Format:
-t : time
-p : pid
-c : creator
-l : limits
-u : summary
-i id [-s -q -m] : details on resource identified by id
usage : ipcs -asmq -tclup
       ipcs [-s -m -q] -i id
       ipcs -h for help.

```

2.1 ipcs

1. 命令格式

```

ipcs [resource-option] [output-format]
ipcs [resource-option] -i id

```

2. 命令功能

提供 IPC 设备的信息

3. 使用方法

resource 选项:

```

ipcs -m    查看系统共享内存信息
ipcs -q    查看系统消息队列信息
ipcs -s    查看系统信号量信息
ipcs [-a]  系统默认输出信息，显示系统内所有的 IPC 信息

```

```

[martin@localhost data]$ ipcs -a

----- Message Queues -----
key          msqid      owner      perms      used-bytes   messages

----- Shared Memory Segments -----
key          shmid       owner      perms      bytes       nattch     status
0x00000000  229376     martin    600        4194304     2         dest
0x00000000  196609     martin    600        524288      2         dest
0x00000000  327682     martin    600        393216      2         dest
0x00000000  491525     martin    600        2097152     2         dest

----- Semaphore Arrays -----
key          semid       owner      perms      nsems

```

输出格式控制:

ipcs -c 查看 IPC 的创建者和所有者
ipcs -l 查看 IPC 资源的限制信息
ipcs -p 查看 IPC 资源的创建者和使用的进程 ID
ipcs -t 查看最新调用 IPC 资源的详细时间
ipcs -u 查看 IPC 资源状态汇总信息

```
[martin@localhost data]$ ipcs -u --human

----- Messages Status -----
allocated queues = 0
used headers = 0
used space = 0B

----- Shared Memory Status -----
segments allocated 4
pages allocated 1760
pages resident 339
pages swapped 0
Swap performance: 0 attempts 0 successes

----- Semaphore Status -----
used arrays = 0
allocated semaphores = 0
```

额外格式控制:

ipcs -l --human

以人类可以阅读的方式显示 size

```
[martin@localhost data]$ ipcs -l --human

----- Messages Limits -----
max queues system wide = 3644
max size of message = 8K
default max size of queue = 16K

----- Shared Memory Limits -----
max number of segments = 4096
max seg size = 16E
max total shared memory = 16E
min seg size = 1B

----- Semaphore Limits -----
max number of arrays = 128
max semaphores per array = 250
max semaphores system wide = 32000
max ops per semop call = 32
semaphore max value = 3276
```

```
[oracle@rhel6lhr ~]$ ipcs -l

----- Shared Memory Limits -----
max number of segments = 4096
max seg size (kbytes) = 98442
max total shared memory (kbytes) = 3221512
min seg size (bytes) = 1

----- Semaphore Limits -----
max number of arrays = 2048
max semaphores per array = 250
max semaphores system wide = 256000
max ops per semop call = 100
semaphore max value = 32767

----- Messages: Limits -----
max queues system wide = 7643
max size of message (bytes) = 65536
default max size of queue (bytes) = 65536
```

2.2 ipcrm

1. 命令功能

通过指定 ID 删除删除 IPC 资源，同时将与 IPC 对象关联的数据一并删除，只有超级用户或 IPC 资源创建者能够删除

2. 使用方法

`ipcrm -M shmkey`

移除用 shmkey 创建的共享内存段

`ipcrm -m shmid`

移除用 shmid 标识的共享内存段

`ipcrm -S semkey`

移除用 semkey 创建的信号量

`ipcrm -s semid`

移除用 semid 标识的信号量

`ipcrm -Q msgkey`

移除用 msgkey 创建的消息队列

`ipcrm -q msgid`

移除用 msgid 标识的消息队列

2.3 如何快速的清理 Oracle 的进程？

真题 1、如何快速的清理 Oracle 的进程？

答案：若想要快速清理掉 Oracle 的进程，则最直接的办法是杀 pmon 进程。有如下 3 条命令可供选择，其中加粗的 orcl 替换成 ORACLE_SID 的值即可。

```
kill -9 `ps -ef|grep orcl| grep -v grep | awk '{print $2}`
ps -ef |grep orcl|grep -v grep|awk '{print $2}' | xargs kill -9
ipcs -m | grep oracle | awk '{print $2}' | xargs ipcrm shm
```

若想要快速杀掉集群的进程，则可以执行如下命令：

```
kill -9 `ps -ef|grep d.bin| grep -v grep | awk '{print $2}'`
```

注意，生产库上严禁使用，否则可能导致集群不能正常启动。

第 3 章 sysresv 命令

3.1 若是一个主机上有多个 oracle 实例的话该如何确定哪个共享内存段属于我们该清掉的 oracle 实例的内存段？

答案：使用 sysresv 命令。sysresv 是 Oracle 在 Linux/Unix 平台提供的工具，用来查看 Oracle 实例使用的共享内存和信号量等信息。sysresv 存放的路径：\$ORACLE_HOME/bin/sysresv。使用时需要设置 LD_LIBRARY_PATH 环境变量，用来告诉 Oracle 共享库文件的位置。sysresv 用法如下：

```
[oracle@rhel6lhr ~]$ sysresv -h
sysresv: invalid option -- 'h'
usage : sysresv [-if] [-d <on/off>] [-l sid1 <sid2> ...]
        -i : Prompt before removing ipc resources for each sid
        -f : Remove ipc resources silently, overrides -i option
        -d <on/off> : List ipc resources for each sid if on
        -l sid1 <sid2> .. : apply sysresv to each sid
Default : sysresv -d on -l $ORACLE_SID
Note    : ipc resources will be attempted to be deleted for a
          sid only if there is no currently running instance
          with that sid.
[oracle@rhel6lhr ~]$ which sysresv
/u01/app/oracle/product/11.2.0/dbhome_1/bin/sysresv
```

来看一下简单使用：

```
oracle@sunvs-b@/oracle/oracle $ uname -a
SunOS sunvs-b 5.10 Generic_139555-08 sun4u sparc SUNW,Sun-Fire-480R
oracle@sunvs-b@/oracle/oracle $ ps -ef|grep pmon
oracle 26257      1   0   5月 24 ?           140:42 ora_pmon_H2
oracle 15479 14078    0 14:01:36 pts/4       0:00 grep pmon
oracle 12449      1   0   8月 17 ?           17:44 ora_pmon_U2

oracle@sunvs-b@/oracle/oracle $ sysresv -l H2

IPC Resources for ORACLE_SID "H2" :
Shared Memory:
ID              KEY
1979711594      0x00000000
1979711595      0x00000000
1979711596      0x00000000
1979711597      0xce653c24
Semaphores:
ID              KEY
16777316        0x25393874
Oracle Instance alive for sid "H2"
```

```
oracle@sunvs-b@/oracle/oracle $ ipcs -ms
IPC status from <running system> as of 2011年08月29日 星期一 14时11分51秒 CST
T          ID          KEY          MODE          OWNER          GROUP
Shared Memory:
m 1577058426 0xf5649758 --rw-r----- oracle oinstall
m 1577058425 0          --rw-r----- oracle oinstall
m 1577058424 0          --rw-r----- oracle oinstall
m 1577058423 0          --rw-r----- oracle oinstall
m 1979711605 0x4e65af  --rw-r--r--  oracle oinstall
m 1979711604 0x3e65af  --rw-r--r--  oracle oinstall
m 1979711603 0x1e65af  --rw-r--r--  oracle oinstall
m 1979711602 0xe65af   --rw-r--r--  oracle oinstall
m 1979711597 0xce653c24 --rw-r----- oracle oinstall
m 1979711596 0          --rw-r----- oracle oinstall
m 1979711595 0          --rw-r----- oracle oinstall
m 1979711594 0          --rw-r----- oracle oinstall
m 1979711511 0x31f4002 --rw-rw-rw-  cupsz cupucuse
m 754974788 0xc93f    --rw-rw-rw-  hsm1 cupucuse
m 754974787 0xc93e    --rw-rw-rw-  hsm1 cupucuse
m 754974786 0xc93d    --rw-rw-rw-  hsm1 cupucuse
m 754974785 0xc93c    --rw-rw-rw-  hsm1 cupucuse
m 754974784 0xc93b    --rw-rw-rw-  hsm1 cupucuse
m 754974783 0xc93a    --rw-rw-rw-  hsm1 cupucuse
m 754974782 0xc939    --rw-rw-rw-  hsm1 cupucuse
m 754974781 0xc938    --rw-rw-rw-  hsm1 cupucuse
m 754974780 0xc937    --rw-rw-rw-  hsm1 cupucuse
m 754974779 0xc936    --rw-rw-rw-  hsm1 cupucuse
m 754974778 0xc935    --rw-rw-rw-  hsm1 cupucuse
m 754974777 0xc934    --rw-rw-rw-  hsm1 cupucuse
m 754974776 0xc933    --rw-rw-rw-  hsm1 cupucuse
m 754974775 0xc932    --rw-rw-rw-  hsm1 cupucuse
m 754974774 0xc930    --rw-rw-rw-  hsm1 cupucuse
m 754974773 0xc92f    --rw-rw-rw-  hsm1 cupucuse
m 754974772 0xc92e    --rw-rw-rw-  hsm1 cupucuse
m 754974771 0xc92d    --rw-rw-rw-  hsm1 cupucuse
m 754974770 0xc931    --rw-rw-rw-  hsm1 cupucuse
m          45 0x741cc1a6 --rw-rw-rw-  root   root
m          44 0x741cc1a5 --rw-rw-rw-  root   root
m          43 0x741cc1a4 --rw-rw-rw-  root   root
m          42 0x741cc1a3 --rw-rw-rw-  root   root
m          41 0x741cc1a2 --rw-rw-rw-  root   root
m          40 0x741cc1a1 --rw-rw-rw-  root   root
m          39 0x741cc1a0 --rw-rw-rw-  root   root
m          37 0x435dce60 --rw-rw-rw-  root   root
m          0 0x22bb     --rw-rw----  root   dba
Semaphores:
s 16777324 0x25393ad4 --ra-r----- oracle oinstall
s 16777320 0x1e65af   --ra-ra-ra-  oracle oinstall
s 16777319 0xe65af    --ra-ra-ra-  oracle oinstall
s 16777316 0x25393874 --ra-r----- oracle oinstall
s 16777296 0          --ra-ra-ra-  cupst cupucuse
s 16777294 0          --ra-ra-ra-  cupst cupucuse
s 16777289 0          --ra-ra-ra-  cuput cupucuse
s 16777287 0          --ra-ra-ra-  cuput cupucuse
s 16777282 0          --ra-ra-ra-  cupvip cupucuse
s 16777280 0          --ra-ra-ra-  cupvip cupucuse
s 16777279 0          --ra-ra-ra-  cupfb cupucuse
s 16777277 0          --ra-ra-ra-  cupfb cupucuse
s 16777268 0          --ra-ra-ra-  cupuc cupucuse
s 16777266 0          --ra-ra-ra-  cupuc cupucuse
s 16777261 0          --ra-ra-ra-  cuphx cupucuse
s 16777259 0          --ra-ra-ra-  cuphx cupucuse
s 16777258 0          --ra-ra-ra-  cupsz cupucuse
s 16777256 0          --ra-ra-ra-  cupsz cupucuse
```



```
s      1    0x55064bec --ra-r--r--    root    root
s      0    0x710644ac --ra-ra-ra-    root    root
```

说明一下：在安装 ORACLE 产品前，需要设置系统的共享内存段的最大值和个数限制，实例在启动后，应尽量保证 SGA 在一个共享内存段上，这里由于我是在 RAC 的一个节点上进行的测试，所以实例内存被分配到 4 个共享内存段上。

IPC 的清理可以使用 `sysresv -if`，如果实例正在运行，清理操作会被终止：

```
oracle@sunvs-b@/oracle/oracle $ sysresv -fi -l H2

IPC Resources for ORACLE_SID "H2" :
Shared Memory:
ID          KEY
1979711594   0x00000000
1979711595   0x00000000
1979711596   0x00000000
1979711597   0xce653c24
Semaphores:
ID          KEY
16777316    0x25393874
Oracle Instance alive for sid "H2"
SYSRESV-005: Warning
      Instance maybe alive - aborting remove for sid "H2"
```

另外如果需要清理内存段和信号量，而 `sysresv` 发现实例是 alive 的，可以使用 `ipcrm` 命令：

```
ipcrm -m <memid>
ipcrm -s <semid>
```

3.1.1 实验

```
[ZFXDESKDB2:oracle]:/oracle>ps -ef|grep ora_pmon_
oracle 12255344 21626964  0 17:43:01 pts/0  0:00 grep ora_pmon_
oracle 17629238      1  0 18:57:42      -  0:09 ora_pmon_rac1hr2
oracle 20250806      1  0 18:57:42      -  0:10 ora_pmon_oraESKDB2

[ZFXDESKDB2:oracle]:/oracle>which sysresv
/oracle/app/oracle/product/11.2.0/db/bin/sysresv

[ZFXDESKDB2:oracle]:/oracle>ORACLE_SID=rac1hr2
[ZFXDESKDB2:oracle]:/oracle>sysresv
```

```
IPC Resources for ORACLE_SID "rac1hr2" :
Shared Memory:
ID          KEY
5242886     0xffffffff
```

```
5242883 0xffffffff
```

```
1048583 0xd92489e0
```

```
Oracle Instance alive for sid "rac1hr2"
```

```
[ZFXDESKDB2:oracle]:/oracle>ipcs
```

```
IPC status from /dev/mem as of Wed Jun 1 17:43:47 BEIST 2016
```

```
T      ID      KEY      MODE      OWNER      GROUP
```

```
Message Queues:
```

```
q      0 0x9283a0d2 -Rrw----- root    system
q      1 0xffffffff ----- root    system
```

```
Shared Memory:
```

```
m 1048576 00000000 --rw-r----- grid    dba
m 1048577 00000000 --rw-r----- grid    dba
m 1048578 0x210000aa --rw-rw---- root    system
m 5242883 00000000 --rw-r----- oracle  asmadmin
m 1048580 00000000 --rw-r----- oracle  asmadmin
m 1048581 00000000 --rw-r----- oracle  asmadmin
m 5242886 00000000 --rw-r----- oracle  asmadmin
m 1048583 0xd92489e0 --rw-r----- oracle  asmadmin
m 1048584 0xd1a4a5d8 --rw-r----- grid    dba
m 8388617 0x3f516768 --rw-r----- oracle  asmadmin
m 759169034 0x21000148 --rw-rw---- oracle  dba
```

```
Semaphores:
```

```
s 3145728 0x0100324a --ra-ra-r-- root    system
s      1 0x620025b4 --ra-r--r-- root    system
s      2 0x02001958 --ra-ra-ra- root    system
s      3 0x01001958 --ra-ra-ra- root    system
s      9 0x010024be --ra----- root    system
s 1048590 0x410000a8 --ra-ra---- root    system
s 11534361 0x41000147 --ra-ra---- oracle  dba
```

```
[ZFXDESKDB2:oracle]:/oracle>ipcs -m
```

```
IPC status from /dev/mem as of Wed Jun 1 17:43:56 BEIST 2016
```

```
T      ID      KEY      MODE      OWNER      GROUP
```

```
Shared Memory:
```

```
m 1048576 00000000 --rw-r----- grid    dba
m 1048577 00000000 --rw-r----- grid    dba
m 1048578 0x210000aa --rw-rw---- root    system
m 5242883 00000000 --rw-r----- oracle  asmadmin
m 1048580 00000000 --rw-r----- oracle  asmadmin
m 1048581 00000000 --rw-r----- oracle  asmadmin
m 5242886 00000000 --rw-r----- oracle  asmadmin
m 1048583 0xd92489e0 --rw-r----- oracle  asmadmin
m 1048584 0xd1a4a5d8 --rw-r----- grid    dba
m 8388617 0x3f516768 --rw-r----- oracle  asmadmin
m 759169034 0x21000148 --rw-rw---- oracle  dba
```

```
[ZFXDESKDB2:oracle]:/oracle>ipcrm -m 5242886
```

```
[ZFXDESKDB2:oracle]:/oracle>ipcrm -m 5242883
[ZFXDESKDB2:oracle]:/oracle>ipcrm -m 1048583
[ZFXDESKDB2:oracle]:/oracle>sysresv

IPC Resources for ORACLE_SID "rac1hr2" :
Shared Memory
ID          KEY
No shared memory segments used
Oracle Instance not alive for sid "rac1hr2"
Oracle Instance not alive for sid "rac1hr2"
[ZFXDESKDB2:oracle]:/oracle>ps -ef|grep ora_pmon_
oracle 17629238      1   0 18:57:42      -   0:09 ora_pmon_rac1hr2
oracle 20250806      1   0 18:57:42      -   0:10 ora_pmon_oraESKDB2
oracle 23330844 21626964    0 17:44:46 pts/0    0:00 grep ora_pmon_
[ZFXDESKDB2:oracle]:/oracle>sqlplus / as sysdba

SQL*Plus: Release 11.2.0.4.0 Production on Wed Jun 1 17:44:52 2016

Copyright (c) 1982, 2013, Oracle. All rights reserved.

Connected to an idle instance.

SYS@rac1hr2> shutdown abort
ORACLE instance shut down.
SYS@rac1hr2> exit
Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, Real Application Clusters, Automatic Storage Management, OLAP,
Data Mining and Real Application Testing options
```

第 4 章 Oracle 内核参数

查看: `more /proc/sys/kernel/shmmax`

临时生效: `echo 3145728 > /proc/sys/kernel/shmmax`

永久生效, 修改文件: `/etc/sysctl.conf`, 并使修改参数立即生效: `/sbin/sysctl -p`

重要的几个参数如下所示:

```
kernel.shmall = 2097152
kernel.shmmax = 1054472192
kernel.shmmni = 4096
kernel.sem = 250 32000 100 128
```

其含义分别如下所示:

(一) `kernel.shmall = 2097152` # `kernel.shmall` 参数是控制共享内存页数。Linux 共享内存页大小为 4KB, 共享内存段的大小都是共享内存页大小的整数倍。如果一个共享内存段的最大大小是 16G, 那么需要共享内

存页数是 $16\text{GB}/4\text{KB} = 16777216\text{KB}/4\text{KB} = 4194304$ (页)，也就是 64Bit 系统下 16GB 物理内存，设置 `kernel.shmall = 4194304` 才符合要求（几乎是原来设置 2097152 的两倍）。简言之，该参数的值始终应该至少为：`ceil(SHMMAX/PAGE_SIZE)`。这个值太小有可能导致数据库启动报错 (ORA-27102: out of memory)。

(二) `kernel.shmmax = 1054472192` #定义一个内存段最大可以分配的内存空间，单位为字节。如果定义太小，那么会导致启动实例失败，或者 SGA 就会被分配到多个共享内存段。那么内存中的指针连接会给系统带来一定的开销，从而降低系统性能。这个值的设置应该大于 `SGA_MAX_TARGET` 或 `MEMORY_MAX_TARGET` 的值，最大值可以设置成大于或等于实际的物理内存。如果 `kernel.shmmax` 为 100M，`sga_max_size` 为 500M，那么启动 Oracle 实例至少会分配 5 个共享内存段；如果设置 `kernel.shmmax` 为 2G，`sga_max_size` 为 500M，那么启动 Oracle 实例只需要分配 1 个共享内存段。

(三) `kernel.shmmni = 4096` #设置系统级最大共享内存段数量，该参数的默认值是 4096。这一数值已经足够，通常不需要更改。。

(四) `kernel.sem = 250 32000 100 128` #信号灯的相关配置，信号灯 `semaphores` 是进程或线程间访问共享内存时提供同步的计数器。可以通过命令 “`cat /proc/sys/kernel/sem`” 来查看当前信号灯的参数配置，如下所示：

```
[root@edsir4p1 ~]# cat /proc/sys/kernel/sem
250      32000    100      128
```

其 4 个值的含义分别如下：

① 250 表示 `SEMMSL`，设置每个信号灯组中信号灯最大数量，推荐的最小值是 250。对于系统中存在大量并发连接的系统，推荐将这个值设置为 `PROCESSES` 初始化参数加 10。

② 32000 表示 `SEMMNS`，设置系统中信号灯的最大数量。操作系统在分配信号灯时不会超过 `LEAST(SEMMNS, SEMMSL*SEMMNI)`。事实上，如果 `SEMMNS` 的值超过了 `SEMMSL*SEMMNI` 是非法的，因此推荐 `SEMMNS` 的值就设置为 `SEMMSL*SEMMNI`。Oracle 推荐 `SEMMNS` 的设置不小于 32000。

③ 100 表示 `SEMOPM`，设置每次系统调用可以同时执行的最大信号灯操作的数量。由于一个信号灯组最多拥有 `SEMMSL` 个信号灯，因此有推荐将 `SEMOPM` 设置为 `SEMMSL` 的值。Oracle 验证的 10.2 和 11.1 的 `SEMOPM` 的配置为 100。

④ 128 表示 `SEMMNI`，设置系统中信号灯组的最大数量。Oracle10g 和 11g 的推荐值为 142。

4.1 kernel.shmmax 参数

4.1.1 实验 1

下面临时设置 `kernel.shmmax` 为 3M，会导致 Oracle 不能启动，设置 `sqlplus` 不能进入：

```
[root@edsir4p1 ~]# echo 3145728 > /proc/sys/kernel/shmmax <<<==== 临时设置 3M
[oracle@edsir4p1- ~]$ more /proc/sys/kernel/shmmax <<<==== 查看是否生效
3145728
[root@edsir4p1 ~]# /sbin/sysctl -a | grep shm
vm.hugetlb shm group = 0
kernel.shmmni = 4096
kernel.shmall = 2097152
kernel.shmmax = 3145728
[root@edsir4p1 ~]# more /etc/sysctl.conf | grep kernel.shm
kernel.shmall = 2097152
kernel.shmmax = 2147483648
kernel.shmmni = 4096
[root@edsir4p1 ~]# su - oracle
[oracle@edsir4p1- ~]$ . PROD1 env
[oracle@edsir4p1-PROD1 ~]$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.1.0 Production on Tue Nov 14 10:09:08 2017

Copyright (c) 1982, 2009, Oracle. All rights reserved.

ERROR:
ORA-12547: TNS:lost contact

Enter user-name:

[oracle@edsir4p1-PROD1 ~]$ oerr ora 12547
12547, 00000, "TNS:lost contact"
// *Cause: Partner has unexpectedly gone away, usually during process
// startup.
// *Action: Investigate partner application for abnormal termination. On an
// Interchange, this can happen if the machine is overloaded.
```

告警日志:

```
Linux Error: 32: Broken pipe
Tue Nov 14 10:00:38 2017
14-NOV-2017 10:00:38 *
(CONNECT_DATA=(SID=PROD1)(CID=(PROGRAM=emagent)(HOST=edsir4p1.us.oracle.com)(USER=oracle))) *
(ADDRESS=(PROTOCOL=tcp)(HOST=10.190.104.111)(PORT=26305)) * establish * PROD1 * 12518
TNS-12518: TNS:listener could not hand off client connection
TNS-12547: TNS:lost contact
TNS-12560: TNS:protocol adapter error
TNS-00517: Lost contact
Linux Error: 32: Broken pipe
```

或启动报错:

```
SYS@PROD1> startup
ORA-00443: background process "PMON" did not start
SYS@PROD1> startup
ORA-12547: TNS:lost contact
SYS@PROD1>
```

有关“TNS-12518: TNS:listener could not hand off client connection”的更多内容请参考:

【故障|监听】TNS-12518、TNS-00517 和 Linux Error : 32 : Broken pipe :

<http://blog.itpub.net/26736162/viewspace-2135468/>

4.1.2 实验 2

下面临时设置 kernel.shmmax 为 100M, sga_max_size 为 500M, 则至少需要 5 个共享内存段, 查看临时段的个数:

```
[root@edsir4p1 ~]# echo 104857600 > /proc/sys/kernel/shmmax
[root@edsir4p1 ~]# more /proc/sys/kernel/shmmax
104857600
[root@edsir4p1 ~]# su - oracle
[oracle@edsir4p1- ~]$ . PROD1_env
[oracle@edsir4p1-PROD1 ~]$ sysresv

IPC Resources for ORACLE_SID "PROD1" :
Shared Memory
ID                KEY
No shared memory segments used<<<==== 无实例的共享内存段
Semaphores:
ID                KEY
98304             0xa3dda878
Oracle Instance not alive for sid "PROD1"
[oracle@edsir4p1-PROD1 ~]$ ipcs

----- Shared Memory Segments -----
key          shmid      owner      perms      bytes      nattch     status
0x00000000   32768      vncuser    644        790528     2          dest
0x00000000   65537      vncuser    644        790528     2          dest
0x00000000   98306      vncuser    644        790528     2          dest

----- Semaphore Arrays -----
key          semid      owner      perms      nsems
0xa3dda878   98304      oracle     660        154

----- Message Queues -----
key          msqid      owner      perms      used-bytes   messages
[oracle@edsir4p1-PROD1 ~]$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.1.0 Production on Tue Nov 14 10:29:07 2017

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Connected to an idle instance.
SYS@PROD1> startup
ORACLE instance started.

Total System Global Area  313860096 bytes
Fixed Size                  1336232 bytes
Variable Size             251661400 bytes
Database Buffers           54525952 bytes
Redo Buffers                6336512 bytes
Database mounted.
Database opened.
SYS@PROD1> show parameter sga
```

```

NAME                                TYPE                                VALUE
-----
lock sga                            boolean                             FALSE
pre page sga                        boolean                             FALSE
sga max size                        big integer                         500M
sga target                          big integer                         300M
SYS@PROD1> exit
Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
[oracle@edsir4p1-PROD1 dbs]$
[oracle@edsir4p1-PROD1 ~]$ sysresv

IPC Resources for ORACLE SID "PROD1" :
Shared Memory:
ID                                KEY
1245194                          0x00000000
1277963                          0x00000000
1310732                          0x00000000
1343501                          0x00000000
1376270                          0x00000000
1409039                          0x90c3be20
Semaphores:
ID                                KEY
917504                           0xa3dda878
Oracle Instance alive for sid "PROD1"
[oracle@edsir4p1-PROD1 ~]$ ipcs

----- Shared Memory Segments -----
key          shmid      owner      perms      bytes      nattch     status
0x00000000  32768      vncuser    644        790528     2          dest
0x00000000  65537      vncuser    644        790528     2          dest
0x00000000  98306      vncuser    644        790528     2          dest
0x00000000  1245194    oracle     660        8388608    30          <<<==== 该共享内存段为 8M
0x00000000  1277963    oracle     660        104857600  30
0x00000000  1310732    oracle     660        104857600  30
0x00000000  1343501    oracle     660        104857600  30
0x00000000  1376270    oracle     660        104857600  30
0x90c3be20  1409039    oracle     660        100663296  30          <<<==== 每个共享内存段为 100M

----- Semaphore Arrays -----
key          semid      owner      perms      nsems
0xa3dda878  917504     oracle     660        154

----- Message Queues -----
key          msqid      owner      perms      used-bytes   messages

```

下面临时设置 kernel.shmmax 为 2G, sga_max_size 为 500M, 则只需要 1 个共享内存段, 查看临时段的个数:

```

[oracle@edsir4p1-PROD1 ~]$ ss

SQL*Plus: Release 11.2.0.1.0 Production on Tue Nov 14 10:49:21 2017

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SYS@PROD1> select 2*1024*1024*1024 from dual;

```

```
2*1024*1024*1024
```

```
-----
2147483648
```

```
SYS@PROD1> exit
```

```
Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
```

```
[oracle@edsir4p1-PROD1 ~]$ sudo echo 2147483648 > /proc/sys/kernel/shmmax
```

```
-bash: /proc/sys/kernel/shmmax: Permission denied
```

```
[oracle@edsir4p1-PROD1 ~]$ su - root
```

```
Password:
```

```
[root@edsir4p1 ~]# echo 2147483648 > /proc/sys/kernel/shmmax
```

```
[root@edsir4p1 ~]# exit
```

```
logout
```

```
[oracle@edsir4p1-PROD1 ~]$ ipcs -m
```

```
----- Shared Memory Segments -----
```

key	shmid	owner	perms	bytes	nattch	status
0x00000000	32768	vncuser	644	790528	2	dest
0x00000000	65537	vncuser	644	790528	2	dest
0x00000000	98306	vncuser	644	790528	2	dest
0x00000000	1245194	oracle	660	8388608	30	
0x00000000	1277963	oracle	660	104857600	30	
0x00000000	1310732	oracle	660	104857600	30	
0x00000000	1343501	oracle	660	104857600	30	
0x00000000	1376270	oracle	660	104857600	30	
0x90c3be20	1409039	oracle	660	100663296	30	

<<<==== 需要重启数据库，重新分配共享内存段

```
[oracle@edsir4p1-PROD1 ~]$ ss
```

```
SQL*Plus: Release 11.2.0.1.0 Production on Tue Nov 14 10:50:23 2017
```

```
Copyright (c) 1982, 2009, Oracle. All rights reserved.
```

```
Connected to:
```

```
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
```

```
With the Partitioning, OLAP, Data Mining and Real Application Testing options
```

```
SYS@PROD1> startup force
```

```
ORACLE instance started.
```

```
Total System Global Area 523108352 bytes
```

```
Fixed Size 1337632 bytes
```

```
Variable Size 343934688 bytes
```

```
Database Buffers 171966464 bytes
```

```
Redo Buffers 5869568 bytes
```

```
Database mounted.
```

```
Database opened.
```

```
SYS@PROD1> exit
```

```
Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
```

```
[oracle@edsir4p1-PROD1 ~]$ sysresv
```

```
IPC Resources for ORACLE_SID "PROD1" :
```

```
Shared Memory:
```

```
ID KEY
```

```
1474570 0x90c3be20
```

```
Semaphores:
```

```
ID KEY
```

```
1081344 0xa3dda878
```

```
Oracle Instance alive for sid "PROD1"
```

```
[oracle@edsir4p1-PROD1 ~]$ ipcs
```


----- Shared Memory Segments -----

key	shmid	owner	perms	bytes	nattch	status
0x00000000	32768	vncuser	644	790528	2	dest
0x00000000	65537	vncuser	644	790528	2	dest
0x00000000	98306	vncuser	644	790528	2	dest
0x90c3be20	1474570	oracle	660	528482304	31	

<<<====共享内存段为 500M

----- Semaphore Arrays -----

key	semid	owner	perms	nsems
0xa3dda878	1081344	oracle	660	154

----- Message Queues -----

key	msqid	owner	perms	used-bytes	messages
-----	-------	-------	-------	------------	----------

4.2 kernel.shmall

该参数设置过小，有可能导致数据库启动报错。很多人调整系统内核参数的时候只关注 SHMMAX 参数，而忽略了 SHMALL 参数的设置。

```
[root@edsir4p1 ~]# echo 10 > /proc/sys/kernel/shmall
[root@edsir4p1 ~]#
[root@edsir4p1 ~]#
[root@edsir4p1 ~]#
[root@edsir4p1 ~]#
[root@edsir4p1 ~]#
[root@edsir4p1 ~]# more /proc/sys/kernel/shmall
10
[oracle@edsir4p1-PROD1 ~]$ ss

SQL*Plus: Release 11.2.0.1.0 Production on Tue Nov 14 11:13:53 2017

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Connected to an idle instance.

SYS@PROD1> startup
ORA-27102: out of memory
Linux Error: 28: No space left on device
SYS@PROD1>
```

4.3 其它博客内容

4.3.1 原文地址：ORACLE 内核参数 作者：it_newbalance

服务器内存为 4G 的情况下

修改 [/etc/sysctl.conf](#) 文件 (ROOT 账户)

```
kernel.shmmax = 2147483648
```

//公式:2G*1024*1024*1024=2147483648(字节)

//表示最大共享内存, 如果小的话可以按实际情况而定, 一般为物理内存的一半(单位:字节)

```
kernel.shmmni=4096
```

//表示最小共享内存固定 4096KB(由于 32 位操作系统默认一页为 4K)

```
kernel.shmall=1048576
```

//公式:4G*1024*1024/4K = 1048576(页)

//表示所有内存大小(单位: 页)

```
kernel.sem=250 32000 100 128
```

//4 个参数依次是 SEMMSL: 每个用户拥有信号量最大数, SEMMNS: 系统信号量最大数, SEMOPM: 每次 semopm 系统调用操作数, SEMMNI: 系统信号量集数最大数。这 4 个参数为固定内容大小

```
fs.file-max=65536
```

//file-max 固定大小 65536

```
net.ipv4.ip_local_port_range=1024 65000
```

//ip_local_port_range 表示端口的范围, 为指定的内容

以上步骤做完执行 `/sbin/sysctl -p` 使内核生效

验证参数 (root 账户执行):

```
#/sbin/sysctl -a | grep shm
```

```
#/sbin/sysctl -a | grep sem
```

```
#/sbin/sysctl -a | grep file-max
```

```
#/sbin/sysctl -a | grep ip_local_port_range
```

最近解决了一些这方面的问题, 并在网络上查询了一些相关资料终于发现一个比较全面解释这类问题的官方文档。本来打算当一次活雷锋全文翻译的, 后来考虑自己英文一般, 并且对于其中一些 OS 相关的知识也没有深入了解。就保留英文大家自己去领会其中的要领, 自己简单总结了一下解决这类问题的关键点并整理一下英文原文。这个文档是 **oracle** 官方技术支持网站 **Metalink** 的资料, 里面引用了一些其它的文档例如 **NOTE:115235.1** 。

对于 **unix** 操作系统中 **Semaphores** 问题只是针对和 **oracle** 相关问题作一些解释。对于信号量和共享内存段参数在不同的系统中可能有不同的参数对应, 具体你去查询对应的 OS 文档。

在解决这类问题的时候我发现大部分问题都是因为在安装 **oracle** 时没有仔细阅读针对指定 OS 的安装说明造成安装实例失败, 一般 **oracle** 的官方文档都详细说明在对应操作系统上如何设置这些内核参数。还有就是因为其他原因 OS 管理人员调整了参数, 但是没有通知 DBA, 一旦 **oracle** 崩溃再次重新启动的时候就可能因为新的内核参数不合适而无法启动。如果是 **oracle** 意外停机之后重新启动不成功, 并出现类似 **ora-27123** 的错误那么一定要询问是否有其他人修改过内核参数, 有时候你没有修改并不代表其他人没有修改哟, 我遇到过不少这样的情况!

1、与 oracle 相关的信号量和共享内存段参数

一般 unix 系统中和信号量相关的是三个参数 **SEMMNI SEMMSL SEMMNS**。他们相互关联决定系统可以分配的信号量。Oracle 使用信号量完成内部进程之间的通信。

关于共享内存段使用 **shmmx** 参数进行总体控制。它指定了系统可以分配的共享内存段最大大小，实际并没有分配那么多只是给出一个可以使用的最大限制。

对于类核参数的修改必须要重新启动系统之后才会生效。

2、出现信号量和共享内存段相关问题的情况

oracle 只有在 **startup nomount** 的时候才会请求 os 的这些资源，用于建立 **SGA** 和启动后台进程。

有些情况下因为 oracle 崩溃之后 os 没有清除 oracle 分配的 **SGA**，也可能造成共享内存段不足，需要人工清除。

3、如何解决相关的问题

你可以简单的修改 **init** 参数减少 oracle 对共享内存段和信号量的需求。

对于控制信号量的三个参数 **SEMMNI SEMMSL SEMMNS**。最终可以使用的信号量由下面公式 提取 (**semmsl * semmni**) 或者 **semmns** 中最小的值。

例如在 **linux** 下。进入目录 **/proc/sys/kernel**；用 **cat** 命令或 **more** 命令查看 semaphore 当前参数的值：

```
cat sem
```

命令运行后将会出现如下的结果：

```
250 32000 32 128
```

其中，250 是参数 **SEMMSL** 的值，32000 是参数 **SEMMNS** 的值，32 是参数 **SEMOPM** 的值，而 128 则是参数 **SEMMNI** 的值。250*128=32000

对于 **oracle7** 需要信号量的设置等于 **init** 中 **processes** 的设置。对于 **8i 9i** 需要等于 **processes*2**。

对于信号量参数的设定一定要小心，因为不正确的设置可能会让系统使用默认值。这个值一般比 **oracle** 系统要求的低。在 **HP unix** 上遇到过这样的问题，当时在参数配置的时候指定两个不同的 **sem-mni** 造成系统使用默认的设置。

对于共享内存段，系统的设置至少要等于 **SGA** 的大小。

Semaphores and Shared Memory

BULLETIN Status: PUBLISHED Content Type: TEXT/PLAIN Creation Date: 05-AUG-2001

Last Revision Date: 05-AUG-2002

PURPOSE-----

To provide an overview of shared memory and semaphores, answer common questions related to these OS resources and provide links to more detailed information.

SCOPE & APPLICATION

This document is intended for anyone who is responsible for creating or administering an Oracle Database. It is intended to compliment the semaphore and shared memory information already provided in the Oracle Installation Guides.

关于信号量和共享内存段的背景知识

Semaphores and shared memory are two very distinct sets of Operating System resources. Semaphores are a system resource that Oracle utilizes for interprocess communication and they occupy a relatively small memory space, while shared memory is utilized to contain the SGA and can garner a large portion of physical memory.

How many of these resources are available and how they are allocated is controlled by the configuration of the operating system kernel('kernel' referring to the centralized core components of the underlying operating system).

There are three OS kernel parameters that work together to limit semaphore allocation and one OS kernel parameter that dictates the maximum size of a shared memory segment.

Operating System kernel parameters generally cannot be tuned on the fly. If they are modified, the changes will not take place until the system is rebooted.

Remember also that the kernel parameters related to semaphores and shared memory represent 'high-water' marks. Meaning that the OS will not automatically allocate a given amount, but will allow up to that given amount to be available upon request.

什么时候信号量和共享内存段问题最有可能发生

Both semaphore or shared memory errors appear primarily at instance startup (The 'startup nomount' stage specifically). This is the only time that Oracle tries to acquire semaphores and shared memory for the instance. Errors related to semaphores or shared memory rarely appear during normal database operations.

The most common circumstance in which these errors occur is during the creation of a new database.

Sometimes when an Oracle instance crashes, however, it's shared memory segments may not be released by the OS. This limits the overall amount of shared memory available for the instance to start up again. In this case, you will need to remove those segments manually.

如何解决信号量和共享内存段问题:

How to resolve semaphore and shared memory errors:

In addressing both semaphore and shared memory errors at instance startup, there are two separate areas that should be considered for reconfiguration.

The first and most simple fix is to modify the init.ora to reduce the number of semaphores or the amount of shared memory Oracle will try to grab at instance startup.

If your situation requires that you not reduce the appropriate init.ora parameters, you will have to modify the operating system kernel to allow the OS to provide more semaphores or allow larger shared memory segments.

SEMAPHORES

=====

=====

IMPORTANT NOTE: ORACLE DOES NOT UTILIZE SEMAPHORES ON AIX OR DIGITAL/TRU64.

与信号量相关的的 **ORA** 错误

What kind of ORA errors are related to semaphores?

'Out of memory' type errors are seldom related to semaphores. Error messages which reference a 'SEMM*****' function are related to semaphores.

IMPORTANT NOTE: THESE ERRORS ONLY OCCUR AT INSTANCE STARTUP.

ORA-7250 "spcre: semget error, unable to get first semaphore set."

ORA-7279 "spcre: semget error, unable to get first semaphore set."

ORA-7251 "spcre:semget error, could not allocate any semaphores."

ORA-7252 "spcre: semget error, could not allocate any semaphores."

ORA-7339 "spcre: maximum number of semaphore sets exceeded."

[NOTE:115235.1] Resolving ORA-7279 or ORA-27146 errors when starting instance

VERY COMMON On Oracle8i and Oracle9i:

ORA-3113 "end-of-file on communication channel" at instance startup.

ORA-27146 "post/wait initialization failed"

[NOTE:115235.1] Resolving ORA-7279 or ORA-27146 errors when starting instance

If you want a very specific explanation of causes for the above errors, refer to:

[NOTE:15566.1] TECH Unix Semaphores and Shared Memory Explained

However, while their exact cause varies, all these error messages indicate that your init.ora is configured to grab more semaphores than the OS has available.

If you configure your OS as indicated in the following sections, you will not get any of the errors indicated above.

成功配置信号量的步骤

The Basic Steps to Semaphore Success:

1. Understand The Basic Concept Behind Semaphores
2. Understand How Many Semaphores Your Oracle Instance(s) Will Attempt to Grab From The Operating System.
3. Configure Your OS Kernel To Accomodate all Your Oracle Instance(s) And also Allow For Future Growth.

[STEP 1] How are semaphores released by the OS for use by an application?

There are 3 OS kernel parameters that work together to limit semaphore allocation. When an application requests semaphores, the OS releases them in 'sets'.

Illustrated here as 2 sets: +---+ +---+

| | | |

| | | |

+---+ +---+

Controlled by SEMMNI -->OS limit on the Number of Identifiers or sets.

Each set contains a tunable number of individual semaphores.

Illustrated here as 2 semaphores per semaphore set: +---+ +---+

| S | | S | S | | S |

+---+ +---+

Controlled by SEMMSL -->The number of semaphores in an identifier or

SEMMNS --> The total Number of Semaphores allowed system wide.

Even though SEMMNI is 100000000 and SEMMSL is 100000000, the max # of semaphores available on your system will only be 10, because SEMMNS is set to 10.

Inversely: Let's say SEMMNI = 10 and SEMMSL = 10 while SEMMNS= 1000000000000000000000000000 Because SEMMNI is 10 and SEMMSL is 10, the max # of semaphores avail on your system will only be 100 or (10 X 10), despite what SEMMNS is set too.

THIS NOTION CAN BE SUMMARIZED BY THE FOLLOWING STATEMENT:

The max # of semaphores that can be allocated on a system will be the lesser of:
(semmsl * semmni) or semmns.

SEMMNI, SEMMSL & SEMMNS are the basic names for OS semaphore kernel parameters, the full name may vary depending on your OS. Consult your OS specific Oracle Install guide.

[NOTE:116638.1] Understanding and Obtaining Oracle Documentation)

[STEP 2] How many semaphores will my Oracle instance(s) require?

With Oracle7: The number of semaphores required by an instance is equal to the setting the 'processes' parameter in the init.ora for the instance.

With Oracle8, Oracle8i and Oracle9i: The number of semaphores required by an instance is equal to 2 times the setting of the 'processes' parameter in the init.ora for the instance. Keep in mind, however, that Oracle only momentarily **grabs 2 X 'processes' then releases half at instance startup**. This measure was apparently introduced to ensure Oracle could not exhaust a system of semaphores.

Oracle may also grab a couple of additional semaphores per instance for internal use.

[STEP 3] Configure your OS kernel to accomodate all your Oracle instances.

There seems to be some confusion of how to deal with lack of semaphore errors. The popular theory being that if Oracle cannot find enough semaphores on a system, increase semmns. This is not always the case, as illustrated in [STEP 1].

Once you have determined your semaphore requirements for Oracle and compensated for future growth, contact your System Administrator or OS vendor for assistance in modifying the OS kernel.

What should I set 'semmni', 'semmsl' & 'semmns' to?

Oracle Support typically does not recommend specific values for semaphore kernel parameters. Instead, use the information provided in this document to set the parameters to values that are appropriate for your operating environment.

For more info please look at the following note : [NOTE:15654.1] TECH: Calculating Oracle's SEMAPHORE Requirements

快速解决信号量问题

Quick fix for resolving lack of semaphore errors:

Reduce the number of semaphores Oracle requires from the OS.

The first and most simple fix is to modify the init.ora to reduce the number of semaphores or the amount of shared memory Oracle will try to grab at instance startup.

Keep in mind, with Oracle8, we grab 2 X 'processes' then release half. This measure was apparently introduced to ensure Oracle could not exhaust a system of semaphores.

如何查找 **OS** 配置的信号量

How can I find out how my OS kernel is configured for semaphores?

The files that are used to tune kernel parameters varies depending on your Operating System. Consult your system administrator or OS vendor, because **viewing the system file may not show accurate information about the runtime values.**

However, an important point to remember is that if a typographical error is made while editing these files, the OS will defer to a default value which is usually too low to accommodate Oracle. So it's a good idea to check runtime values with utilities like '/etc/sysdef'.

I've tuned my OS kernel parameters, but I am still having semaphore problems....

常见问题！！

This may mean that you made a typographical error or did not rebuild your Operating System kernel correctly(if a typographical error is made while editing these files, the OS will defer to a default value which is usually too low to accommodate Oracle).

On Solaris, check current OS kernel values with this command:

```
> /etc/sysdef|grep -i semm
```

If these values do not reflect what you put in your 'system' file, you likely made a typographical error.

On HP, be sure the OS kernel was rebuilt correctly and that the OS was booted off the correct file. Contact your System Administrator or HP for more information.

在 Linux 系统上

进入目录/proc/sys/kernel; 用 cat 命令或 more 命令查看 semaphore 当前参数的值:

```
cat sem
```

命令运行后将会出现如下的结果:

```
250 32000 32 128
```

其中, 250 是参数 SEMMSL 的值, 32000 是参数 SEMMNS 的值, 32 是参数 SEMOPM 的值, 而 128 则是参数 SEMMNI 的值。250*128=32000

如何获得当前正在使用的信号量

How can I determine how many semaphores are currently being utilized?

On most Unix systems, current semaphore allocation can be displayed with the OS command 'ipcs -s'.

```
% ipcs -s
```

While good to know, this command is seldom used as part of troubleshooting semaphore errors.

SHARED MEMORY

OS 如何分配共享内存段

How is shared memory allocated by the OS?

This process varies slightly depending on Unix platform, but the basic premise is this:

An application requests a given amount of contiguous shared memory from the OS. The OS dictates how large of a shared memory segment it will allow with the kernel

parameter SHMMAX(Shared Memory Maximum). If the amount of shared memory requested by the application is greater than SHMMAX, the OS may be granted the shared memory in multiple segments. Ideally, however, you want the amount requested by the application to be less than SHMMAX so that the application's request can be fulfilled with one shared memory segment.

SHMMAX 和 SGA 的关系

How does SHMMAX relate to my SGA?

Since the SGA is comprised of shared memory, SHMMAX can potentially limit how large your SGA can be and/or prevent your instance from starting.

What limits the size of my SGA?

In no particular order.

5. The amount of Physical Memory and Swap space available on your system.
6. The kernel parameter SHMMAX.
7. Other OS specific limitations on shared memory.

Memory SHMMAX OS Limits +-----+ +-----+ +-----+

| | | | | +-----+

| | | | | S |

| | | | | > | G |

| | | | | A |

| | | | | +-----+

+-----+ +-----+ +-----+

Some OS specific limitations are discussed in the following documents:

"Oracle Administrator's Reference" available on the Oracle Install CD

Additionally:

HP-UX: [NOTE:77310.1] HP-UX Large SGA support for HP, Memory Windows

[NOTE:69119.1] HP-UX SGA Sizing Issues on HP-UX

Solaris: [NOTE:61896.1] SOLARIS: SGA size, sgabeg attach address and Sun

与共享内存当相关的错误

What kind of ORA errors are related to shared memory?

Error Messages referencing a 'SHMM****' function are related to shared memory.

ORA-7306, ORA-7336, ORA-7329, ORA-7307, ORA-7337, ORA-7320, ORA-7329, ORA-7334

VERY COMMON IN 8i: **ORA-27100 "shared memory realm already exists"** ORA-27102 "out of memory"

ORA-27125 "unable to create shared memory segment" and/or "linux 43 identifier removed"

ORA-27123 "unable to attach to shared memory segment"

[NOTE:115753.1] UNIX Resolving the ORA-27123 error

[NOTE:1028623.6] SUN SOLARIS: HOW TO RELOCATE THE SGA

如何设置 **SHMMAX**

What should I set 'shmmax' to?

On some Unix platforms, the Install Guide recommends specific values. Previous

versions of the Install Guide recommended setting SHMMAX to .5 *(physical memory present in machine). Most recently it's been suggested SHMMAX be set to 4294967295 (4GB). This may not seem appropriate, particularly if the system has considerably less physical memory available, but it does prevent you from having to modify your system kernel everytime a new instance is created or additional physical memory is added to the system. Remember that SHMMAX is a high water mark, meaning that the OS will attempt to allow up to that amount for an application.

解决缺少共享内存段的问题

Quick fix for resolving lack of shared memory errors:

NOTE: If you have never configured your OS kernel for shared memory, you cannot employ this 'Quick Fix'. You will have to first configure the OS kernel. **The amount of shared memory Oracle requests is roughly equal to the size of the SGA.** The first and most simple fix is to modify the init.ora to reduce the amount of shared memory Oracle will try to grab at instance startup.

This document lists the init.ora parameters that contribute to the size of the SGA:

[NOTE:1008866.6] HOW TO DETERMINE SGA SIZE (8.0, 8i, 7.x)

oracle 崩溃之后重新启动失败的问题

My instance crashed. When I try to restart it, I receive errors related to shared memory. What should I do?

This may indicate that the shared memory segment associated with the SGA of the crashed instance is still in memory. In this case it may be appropriate to manually remove the segment using OS commands.

THIS PROCESS SHOULD NOT BE ATTEMPTED UNLESS YOU FULLY UNDERSTAND THE CONCEPTS BEHIND IT!!!

The basic steps are:

1. Identify the shared memory segment that is 'stuck' in memory.
2. Remove the 'stuck' shared memory segment using the OS command 'ipcrm'.

[NOTE:68281.1] DETERMINING WHICH INSTANCE OWNS WHICH SHARED MEMORY & SEMAPHORE SEGMENTS

[NOTE:69642.1] also describes this process - Step 9.

[NOTE:123322.1] SYSRESV UTILITY: This note describes the new 8i 'sysresv' utility that can be used on Solaris to associate a given ORACLE_SID with it's shared memory segment(s). .

4.3.2 Oracle 性能优化之内核的 shmall 和 shmmax 参数

1. 内核的 shmall 和 shmmax 参数

SHMMAX= 配置了最大的内存 segment 的大小 —>这个设置的比 SGA_MAX_SIZE 大比较好。

SHMMAX 参数: Linux 进程可以分配的单独共享内存段的最大值。一般设置为内存总大小的一半。这个值的设置应该大于 SGA_MAX_TARGET 或 MEMORY_MAX_TARGET 的值, 因此对于安装 Oracle 数据库的系统, shmmax 的值应该比内存的二分之一大一些。

SHMMIN= 最小的内存 segment 的大小。

SHMMNI= 整个系统的内存 segment 的总个数。设置系统级最大共享内存段数量。Oracle10g 推荐最小值为 4096, 可以适当比 4096 增加一些。

SHMSEG= 每个进程可以使用的内存 segment 的最大个数

shmall=是全部允许使用的共享内存大小, shmmax 是单个段允许使用的大小。这两个可以设置为内存的 90%。例如 16G 内存, $16 \times 1024 \times 1024 \times 1024 \times 90\% = 15461882265$, shmall 的大小为 $15461882265 / 4k (\text{getconf PAGESIZE 可得到}) = 3774873$ 。

shmall 设置共享内存总页数。这个值太小有可能导致数据库启动报错。很多人调整系统内核参数的时候只关注 SHMMAX 参数, 而忽略了 SHMALL 参数的设置。

—

2. 配置信号灯 (semaphore) 的参数

信号灯 semaphores 是进程或线程间访问共享内存时提供同步的计数器。

SEMMSL= 设置每个信号灯组中信号灯最大数量, 推荐的最小值是 250。对于系统中存在大量并发连接的系统, 推荐将这个值设置为 PROCESSES 初始化参数加 10。

SEMMNI= 设置系统中信号灯组的最大数量。Oracle10g 和 11g 的推荐值为 142。

SEMMNS=设置系统中信号灯的最大数量。操作系统在分配信号灯时不会超过 $\text{LEAST}(\text{SEMMNS}, \text{SEMMSL} \times \text{SEMMNI})$ 。事实上, 如果 SEMMNS 的值超过了 SEMMSL*SEMMNI 是非法的, 因此推荐 SEMMNS 的值就设置为 SEMMSL*SEMMNI。Oracle 推荐 SEMMNS 的设置不小于 32000, 假如数据库的 PROCESSES 参数设置为 600, 则 SEMMNS 的设置应为:

```
SQL> select (600+10)*142 from dual;
```

```
(600+10)*142
```

```
-----
```

```
86620
```

```
1
```

```
2
```

```
3
```

```
4
```

```
5
```

SEMOPM 参数：设置每次系统调用可以同时执行的最大信号灯操作的数量。由于一个信号灯组最多拥有 SEMMSL 个信号灯，因此有推荐将 SEMOPM 设置为 SEMMSL 的值。Oracle 验证的 10.2 和 11.1 的 SEMOPM 的配置为 100。

通过下面的命令可以检查信号灯相关配置：

```
# cat /proc/sys/kernel/sem
```

```
250 32000 100 128
```

```
1
```

```
2
```

对应的 4 个值从左到右分别为 SEMMSL、SEMNS、SEMOPM 和 SEMMNI

```
—
```

3. 修改 /etc/sysctl.conf

```
kernel.shmmax=15461882265
```

```
kernel.shmall=3774873
```

```
kernel.msgmax=65535
```

```
kernel.msgmnb=65535
```

执行 `sudo sysctl -p`

可以使用 `ipcs -l` 看结果, `ipcs -u` 可以看到实际使用的情况

About Me

- 本文作者：小麦苗，只专注于数据库的技术，更注重技术的运用
- 本文在 itpub (<http://blog.itpub.net/26736162/abstract/1/>)、博客园 (<http://www.cnblogs.com/lhrbest>) 和个人微博 (<http://weibo.com/lhrbest>)
- 本文 itpub 地址: <http://blog.itpub.net/26736162/viewspace-2147273/>
- 本文博客园地址: <http://www.cnblogs.com/lhrbest/p/7838547.html>
- 本文 pdf 版、个人简介及小麦苗云盘地址: <http://blog.itpub.net/26736162/viewspace-1624453/>

- 数据库笔试面试题库及解答: <http://blog.itpub.net/26736162/viewspace-2134706/>
- DBA 宝典今日头条号地址: <http://www.toutiao.com/c/user/6401772890/#mid=1564638659405826>
-
- QQ 群号: 230161599 (满)、618766405
- 微信群: 可加我微信, 我拉大家进群, 非诚勿扰
- 联系我请加 QQ 好友 (646634621), 注明添加缘由
- 于 2017-11-06 09:00 ~ 2017-11-06 22:00 在魔都完成
- 文章内容来源于小麦苗的学习笔记, 部分整理自网络, 若有侵权或不当之处还请谅解
- 版权所有, 欢迎分享本文, 转载请保留出处
-
- 小麦苗的微店: <https://weidian.com/s/793741433?wfr=c&iffr=shopdetail>
- 小麦苗出版的数据库类丛书: <http://blog.itpub.net/26736162/viewspace-2142121/>
-

使用**微信客户端**扫描下面的二维码来关注小麦苗的微信公众号 ([xiaomaimiaolhr](#)) 及 QQ 群 (DBA 宝典), 学习最实用的数据库技术。



小麦苗的微信公众号



DBA 宝典 QQ 群 2



《DBA 笔试面宝典》



扫二维码进店
小麦苗的微店