**控制Cache Memory/Page Cache Size**

# 怎么样观察当前的cache memory的使用情况？

[root@gurkullinux ~]# free -m
total used free shared buffers cached
Mem: 15976 15195 781 0  167     **9153**
-/+ buffers/cache: 5874 10102
Swap: 2000 0 1999
[root@gurkullinux ~]#

Note：单位是M

# How to limit the Page Cache or Cache Memory size in Red Hat Enterprise Linux ( RHEL) 6?

在RHEL 6中，页面缓存是动态控制的，并且可以占用机器中可用的内存。 这里重要的一点是，在RHEL中没有内核参数可以直接控制页面缓存大小，但我们在这里可以做的只是通过调整一些可配置的内核参数来限制页面缓存的增长。

# What are configurable kernel parameters that controls the size of the page cache ( or cache memory ) size?

vm.vfs\_cache\_pressure (default = 100)
  它控制着内核回收内存的趋势。 减少vfs\_cache\_pressure会导致内核倾向于保留dentry和inode缓存。 将vfs\_cache\_pressure增加到100以上会导致内核更喜欢回收dentries和inode

  增加这个值（例如500）会使你更频繁地回收高速缓冲存储器并限制页面高速缓存的大小。
vm.dirty\_background\_ratio ( default = 20)
  指示系统内存的百分比，pdflush后台写回守护进程将开始写出脏数据的页数。 减少这个数字会导致pdflush开始更快地输出污垢数据，并且会限制页面缓存的大小。

Note: Ubuntu 16.04 的系统default=10
 vm.dirty\_ratio (default=40)

表示进程写出自己的脏数据的页数的百分比。 减少此数字将导致进程更快地写出脏数据，这将限制页面缓存大小。
vm.dirty\_expire\_centisecs ( default = 3000 , mentioned in milliseconds)
  指示脏页的到期时间，以便它们有资格被pdflush刷新。 减小此值将使更脏的数据页符合刷新条件，并且会限制页面缓存大小。
 vm.swappiness ( default=60)

表示，您想要换出数据的时间越短，值越有可能交换。 减小此值将使机器交换的可能性降低，因此更愿意将数据写入磁盘。 这将限制页面缓存大小。

注意：使此值为零，不会停止交换系统。

# How to Change these kernel Parameters ?

如果要在运行的机器上动态更改参数，可以使用类似于下面的命令

echo “500” > /proc/sys/vm/vfs\_cache\_pressure

如果你想持久化配置（下面的命令将修改/etc/sysctl.conf）

sysctl -w vm.vfs\_cache\_pressure=”500″

如果遇到像整个页面缓存占用物理内存那样的问题，最初应考虑逐个调整参数，将值改为15％至20％。 但是不要一下子改变它们，因为对这些值的修改可能以显着的方式以正面或负面的方式影响系统的性能。

除非您确认之前的更改不会对您的系统造成负面影响，否则不要一次进行多项更改。

# 一个例子

默认情况下，在所有RHEL 6系统中，将这些值设置如下，并发现它非常有效。

vm.swappiness=0
vm.dirty\_background\_ratio=3
vm.dirty\_ratio=15
vm.dirty\_expire\_centisecs=500
vm.dirty\_writeback\_centisecs=100