一、问题

网上有很多关于修改`memory.swappiness`的文章，但大多没有经过验证。实际上，在目前流行的Linux发行版中，因为`cgroup v1`是内核默认的资源管理方式，而该版本中`memory.swappiness`存在缺陷，导致修改`memory.swappiness`值不生效。

通常都是要求这样配置`memory.swappiness`：

```
[root@localhost ~]# echo 'vm.swappiness = 10' >> /etc/sysctl.conf
[root@localhost ~]# sysctl -p
[root@localhost ~]# reboot
```

以上配置存在一个问题，修改的新值只会被应用到`cgroup memory`子系统的顶层，顶层之下的子层不会生效，而用户进程一般默认受子层`cgroup`控制（不同`systemd`版本可能会有差异），可以这样验证（假设`memory`子系统挂载位置为`/sys/fs/cgroup/memory/`）：

```
# 顶层值同sysctl配置值：
[root@bogon ~]# cat /sys/fs/cgroup/memory/memory.swappiness
10

# 子层还是默认值60：
[root@bogon ~]# cat /sys/fs/cgroup/memory/user.slice/memory.swappiness
60
[root@bogon ~]# cat /sys/fs/cgroup/memory/system.slice/memory.swappiness
60

# 示例系统上运行了GPDB数据库，我们看一下它的进程情况：
[yz@bogon ~]$ ps ux | grep postgres
yz   12010  0.0  0.9 559700 78932 ?   Ss  20:16  0:00
/opt/gpdb/gp_bin/bin/postgres -D /opt/gpdb/db/primary/gpseg0 -c gp_role=execute
...

# 可以看到该进程的memory受/user.slice资源组控制：
[yz@bogon ~]$ cat /proc/12010/cgroup
11:devices:/user.slice
10:memory:/user.slice
...
```

二、解决方法

2.1 原理

先通过`sysctl`修改系统配置文件，然后启动一个系统服务，在系统启动时，强制重新设置`cgroup`子层的`memory.swappiness`值。

2.2 方法

本方法摘自参考文献[1]。在`CentOS 7.6`上验证可行。

方法是简单地强行改变`memory.swappiness`。最好是在引导时，在所有现有的`cgroups`上切换，特别是`After=systemd-sysctl.service`。我草拟了以下方法。

第1步：还是需要先修改`sysctl`配置。

```
# 假设需要修改为10
[root@localhost ~]# echo 'vm.swappiness = 10' >> /etc/sysctl.conf
[root@localhost ~]# sysctl -p
# 查看配置是否生效
[root@localhost ~]# sysctl vm.swappiness
```

第2步：编写服务脚本。

```
[root@localhost ~]# cat /usr/bin/cgroup_swappiness_set.sh
#!/bin/sh

CGROUP_V1_MEMORY_DIR=$(mount | grep "^cgroup .*memory" | cut -d ' ' -f 3)

if [ -z $CGROUP_V1_MEMORY_DIR ]; then
    exit -22  # EINVAL
fi

GLOBAL_SWAPPINESS=$(cat /proc/sys/vm/swappiness)
for cg in $(find $CGROUP_V1_MEMORY_DIR -name memory.swappiness); do
    echo $GLOBAL_SWAPPINESS > $cg
done

[root@localhost ~]# chmod +x /usr/bin/cgroup_swappiness_set.sh
```

第3步：编写服务配置文件。

```
[root@localhost ~]# cat /usr/lib/systemd/system/swappiness_fix.service
[Unit]
Description=Set all existing -v1 memory cgroups to global vm.swappiness
After=systemd-sysctl.service

[Service]
Type=oneshot
ExecStart=/usr/bin/cgroup_swappiness_set.sh

[Install]
WantedBy=multi-user.target
```

第4步：启用脚本，并重启服务器。

```
[root@localhost ~]# systemctl enable swappiness_fix.service
[root@localhost ~]# systemctl start swappiness_fix.service
[root@localhost ~]# reboot
```

2.3 脚本的其他用法

上一小节的脚本，也可以在不重启服务器的情况下生效：先完成第1步，第2步时，直接运行`/usr/bin/cgroup_swappiness_set.sh`脚本即可。但是需要注意的是，修改`swappiness`不会影响正在运行的进程。

三、更多

3.1 与cgroup关系

`swappiness`是`cgroup v1`的特性，在`cgroup v2`已经被替代。

> swapiness is a cgroupvs1 feature, and it has no counterpart on cgroupsv2. (the new latency stuff is maybe a better replacement though).

`swappiness`实现很糟糕，systemd不打算再维护它了。

3.2 关于swappiness=0

`swappiness=0`就相当于关闭`swap`么?

与内核版本有关，kernel 3.5以上表示关闭`swap`，kernel 3.5以前的版本表示“尽量避免使用`swap`”。所以，想要彻底关闭`swap`，那就使用`swapoff`命令。

四、参考

1. system.slice swappiness is inconsistent with vm.swappiness sysctl
2. Turning off swapping for only one process with cgroups?
3. How do I configure swappiness?