# 天坑系列bash之命令执行上下文是否为子进程

[mnstory.net](http://mnstory.net/)

写了一段脚本，如果系统有flock命令，就用flock来加锁，否则用ps来查看是否已经有本进程在运行，以保障其单例性。

lockEntry()

{

if ! which flock >/dev/null 2>&1; then

#if no which or no flock, use ps style

echo "[ps] flock is not found"

# 1, 0(ps not found) or empty(wc not found) or anything other invalid, can run

# 2 is not running, can run

# > 2 has already running, can't run

local count=$(ps aux 2>/dev/null | grep $YOUREPROCESSNAME | grep -v grep | wc -l 2>/dev/null)

if [ "" != "$count" -a $count -gt 2 ]; then

echo "[ps] has same process already running"

return 1

fi

echo "[ps] found count $count, can run"

main

else

echo "[flock] flock found"

local lock=""

if [ -d "/tmp" ]; then

lock="/tmp"

fi

lock="$lock/heartalive.lock"

touch ${lock}

(

if ! flock -n -x 200; then

echo "[flock] flock $lock failed $?"

return 1

fi

echo "[flock] flock $lock success"

main

) 200<${lock}

fi

return 0

}

lockEntry

这里用了：

local count=$(ps aux 2>/dev/null | grep $YOUREPROCESSNAME | grep -v grep | wc -l 2>/dev/null)

if [ "" != "$count" -a $count -gt 2 ]; then

echo "[ps] has same process already running"

return 1

fi

来获取进程个数，判断的时候，却用的是$count -gt 2而不是$count -gt 1？

这个问题，剥离出来，是wc -l命令执行的上下文是当前脚本进程还是其子进程。

你在脚本里，这么调用，显示的是 1 ：

ps aux | grep $YOUREPROCESSNAME | grep -v grep 2>/dev/null | wc -l

你想获取变量值的时候，如下，count却是 2：

local count=`ps aux | grep $YOUREPROCESSNAME | grep -v grep 2>/dev/null | wc -l `

因为``，$()等执行上述命令时，又会fork一个子进程来执行，在``，$()等内部显示类似这样：

root 20553 0.0 0.0 12428 1640 pts/0 S+ 14:39 0:00 | \\_ /bin/bash ./YOUREPROCESSNAME.sh

root 20568 0.0 0.0 12428 688 pts/0 S+ 14:39 0:00 | \\_ /bin/bash ./YOUREPROCESSNAME.sh

所以，上面的$count，如果在脚本你们要判断脚本名称的进程是否存在，条件为大于2而不是大于1。

2017/4/16