

shahongzhou的专栏

≡ 目录视图

≡ 摘要视图

RSS 订阅

个人资料



shahongzhou

访问：143983次

积分：2023

等级：5

排名：第11605名

原创：46篇

转载：74篇

译文：3篇

评论：18条

文章搜索

文章分类

读书笔记 (27)

面试宝典 (4)

实习机会 (3)

编程小记 (25)

论文 (5)

笑笑 (2)

专利 (1)

文章存档

2015年10月 (3)

2015年09月 (2)

2015年07月 (2)

2015年06月 (6)

2015年05月 (2)

展开

阅读排行

linux查看文件有多少行 (9762)

北邮校园网常见知识问答 (6633)

基于位置的服务（简介） (5585)

36辆车，6条跑道，无计 (4727)

使程序在Linux下后台运行 (4508)

2016软考项目经理实战班 学院周年礼-顶尖课程钜惠呈现 Hadoop英雄会—暨Hadoop 10周年生日大趴 【博客专家】有奖试读—Windows PowerShell实战指南

使程序在Linux下后台运行（关掉终端继续让程序运行的方法）

标签：dataset 服务器 程序运行

2014-05-13 14:02 4513人阅读 评论(0) 收藏 举报

分类：编程小记 (24)

一、为什么要使程序在后台运行

我们计算的程序都是周期很长的，通常要几个小时甚至一个星期。我们用的环境是用putty远程连接到日本Linux服务器。所以使程序在后台跑有以下三个好处：

1：我们这边是否关机不影响日本那边的程序运行。（不会像以前那样，我们这网络一断开，或一关机，程序就断掉或找不到数据，跑了几天的程序只能重头再来，很是烦恼）

2：不影响计算效率

2：让程序在后台跑后，不会占据终端，我们可以用终端做别的事情。

二、怎么样使程序在后台执行

方法有很多，这里主要列举两种。假如我们有程序pso.cpp,通过编译后产生可执行文件pso，我们要使pso在linux服务器后台执行。当客户端关机后重新登入服务器后继续查看本来在终端输出的运行结果。（假设操作都在当前目录下）

方法1在终端输入命令：

```
# ./pso > pso.file 2>&1 &
```

解释：将pso直接放在后台运行，并把终端输出存放在当前目录下的pso.file文件中。

当客户端关机后重新登陆服务器后，直接查看pso.file文件就可看执行结果（命令：#cat pso.file）。

方法2在终端输入命令：

```
# nohup ./pso > pso.file 2>&1 &
```

解释：nohup就是不挂起的意思，将pso直接放在后台运行，并把终端输出存放在当前目录下的pso.file文件中。当客户端关机后重新登陆服务器后，直接查看pso.file文件就可看执行结果（命令：#cat pso.file）。

三、常用任务管理命令

```
# jobs //查看任务，返回任务编号n和进程号
```

```
# bg %n //将编号为n的任务转后台运行
```

```
# fg %n //将编号为n的任务转前台运行
```

```
# ctrl+z //挂起当前任务
```

http://blog.csdn.net/shahongzhou/article/details/25706315

1/4

linux popen用法 (4391)

linux下md5加密（使用openssl） (3986)

vim,ctag和Taglist (3385)

linux下调试python程序 (3196)

Transfer-Encoding:chunked (3193)

评论排行

北邮校园网常见知识问答 (4)

五人分鱼问题（附答案） (2)

linux下md5加密（使用openssl） (2)

DevC++ 工程没有调试信息 (2)

36辆车，6条跑道，无计时器 (2)

记录一个bug (2)

strcpy、snprintf、memcpy (1)

uTP协议的前世今生（from wikipedia） (1)

adaboostM2初探（记录一个例子） (1)

后台程序结果重定向到文件 (1)

推荐文章

\*一个炫字都不够？！！！！手把手带你打造3D自定义view

\*21行python代码实现拼写检查器

\*数据库性能优化之SQL语句优化

\*拉开大变革序幕（下）：分布式计算框架与大数据

\*Chromium网页URL加载过程分析

\*Hadoop中止下线操作后大量剩余复制块的解决方案

最新评论

adaboostM2初探（记录一个例子）  
xinrumingjing1: 你好，可否将工具包发我一份，谢谢，邮箱xinrumingjing86@163.com。

五人分鱼问题（附答案）  
qq\_31475461: sleep 是什么鬼

五人分鱼问题（附答案）  
hzy147611: def can\_divided(the\_sum, divided\_times=5): if divi...

36辆车，6条跑道，无计时器，真难！  
shahongzhou: @u014330035: 举个例子，第一组第一名跑9秒，第二名11秒，第三名12秒。第二组第一名，9....

36辆车，6条跑道，无计时器，真难！  
u014330035: 第一问是8辆车肯定没有问题。但是第二问感觉有点问题啊。再筛的时候，第一组取前三肯定没问题。第二组和第...

后台程序结果重定向到文件，结果不对！  
itjinks: 和你遇到同样的问题了，谢谢！

linux下md5加密（使用openssl）  
zsy19881226: printf("%s\n",buf); 应为: printf("%s\n",buf);

DevC++ 工程没有调试信息的解决方案  
sinat\_28883099: 谢谢了

uTP协议的前世今生（from wikipedia）  
u010850027: 谢谢分享

linux下md5加密（使用openssl）  
hxbt: good

```
# ctrl+c //结束当前任务
```

注：如果要使在前天执行任务放到后台运行，则先要用ctrl+z挂起该任务，然后用bg使之后台执行。

附：

在Linux中，如果要让进程在后台运行，一般情况下，我们在命令后面加上&即可，实际上，这样是将命令放入到一个作业队列中了：

```
$ ./test.sh &
[1] 17208

$ jobs -l
[1]+ 17208 Running                  ./test.sh &
```

对于已经在前台执行的命令，也可以重新放到后台执行，首先按ctrl+z暂停已经运行的进程，然后使用bg命令将停止的作业放到后台运行：

```
$ ./test.sh
[1]+  Stopped                  ./test.sh

$ bg %1 [1]+ ./test.sh &

$ jobs -l
[1]+ 22794 Running                ./test.sh &
```

但是如上方到后台执行的进程，其父进程还是当前终端shell的进程，而一旦父进程退出，则会发送hangup信号给所有子进程，子进程收到hangup以后也会退出。如果我们要在退出shell的时候继续运行进程，则需要使用nohup忽略hangup信号，或者setsid将父进程设为init进程(进程号为1)

```
$ echo $$
21734

$ nohup ./test.sh &
[1] 29016

$ ps -ef | grep test
515      29710 21734  0 11:47 pts/12    00:00:00 /bin/sh ./test.sh
515      29713 21734  0 11:47 pts/12    00:00:00 grep test

$ setsid ./test.sh &
[1] 409

$ ps -ef | grep test
515      410      1  0 11:49 ?          00:00:00 /bin/sh ./test.sh
515      413 21734  0 11:49 pts/12    00:00:00 grep test
```

上面的试验演示了使用nohup/setsid加上&使进程在后台运行，同时不受当前shell退出的影响。那么对于已经在后台运行的进程，该怎么办呢？可以使用disown命令：

```
$ ./test.sh &
[1] 2539

$ jobs -l
[1]+  2539 Running                ./test.sh &

$ disown -h %1

$ ps -ef | grep test
515      410      1  0 11:49 ?          00:00:00 /bin/sh ./test.sh
515      2542 21734  0 11:52 pts/12    00:00:00 grep test
```

另外还有一种方法，即使将进程在一个subshell中执行，其实这和setsid异曲同工。方法很简单，将命令用括号()括起来即可：

```
$ (./test.sh &)

$ ps -ef | grep test
515      410      1  0 11:49 ?          00:00:00 /bin/sh ./test.sh
```

```
515      12483 21734  0 11:59 pts/12   00:00:00 grep test
```

注：本文试验环境为Red Hat Enterprise Linux AS release 4 (Nahant Update 5),shell为/bin/bash，不同的OS和shell可能命令有些不一样。例如AIX的ksh，没有disown，但是可以使用nohup -p *PID*来获得disown同样的效果。

还有一种更加强大的方式是使用**screen**，首先创建一个断开模式的虚拟终端，然后用-r选项重新连接这个虚拟终端，在其中执行的任何命令，都能达到nohup的效果，这在有多个命令需要在后台连续执行的时候比较方便：

```
$ screen -dmS screen_test
$ screen -list There is a screen on:
                27963.screen_test      (Detached) 1 Socket in /tmp/uscreens/S-jiangfeng.
$ screen -r screen_test
```

顶

0

踩

0

上一篇

去除Python readline()函数读入的\n

下一篇

urllib2.urlopen超时问题

我的同类文章

编程小记（24）

• 欢迎使用CSDN-markdown编辑器

• matplotlib使用指南

• Python SocketAPI

• tcpdump实验总结

• 后台程序结果重定向到文件，结果看不到文件？缓...

• 序列S的所有可能情况

• Python使用heapq实现小顶堆（TopK大）、大顶堆...

• 武林中的爱恨情仇<杂谈机器学习的拟合，过拟合，...

• [LeetCode]Reverse Integer

• 从有序链表中去掉重复的

更多

主题推荐

终端

linux

远程连接

服务器

日本

网络

数据

putty

猜你在找

- Linux环境C语言编程基础

linux下oracle11g 高可用dataguard从0开始一步一步3

高并发linux网络服务器核心代码实现

《C语言/C++学习指南》Linux开发篇

嵌入式Linux C编程基础
- Qt程序运行自动弹出qtcreator\_process\_stub终端关闭

Linux下计算程序运行时间的两种方法

Linux下计算程序运行时间的两种方法

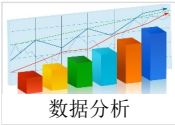
Linux下计算程序运行时间的一个简单方法

Linux下评估程序运行时间及内存占用情况的简便方法



查看评论

暂无评论



您还没有登录,请[登录](#)或[注册](#)

\* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题

Hadoop

AWS

移动游戏

Java

Android

iOS

Swift

智能硬件

Docker

OpenStack

VPN

Spark

ERP

IE10

Eclipse

CRM

JavaScript

数据库

Ubuntu

NFC

WAP

jQuery

BI

HTML5

Spring

Apache

.NET

API

HTML

SDK

IIS

Fedora

XML

LBS

Unity