# Linux 系统学习以及问题定位

## Linux的进程解析

**父子进程 孤儿进程以及僵尸进程：**

每一个进程都有一个进程号（PID），进程号是一个正数，用以唯一标识系统中的某个进程。

每个进程都有一个创建自己的父进程。使用系统调用getppid()可以获取父进程的进程号。

实际上，每个进程的父进程号属性反映了系统上所有进程间的树状关系。

每个父进程的父进程又有自己的父进程，以此类推，回溯到1号进程——init进程，即所有进程的始祖。使用pstree命令可以查看这一树状关系。

孤儿进程：一个父进程退出，而它的一个或多个子进程还在运行，那么那些子进程将成为孤儿进程。孤儿进程将被init进程(进程号为1)所收养，并由init进程对它们完成状态收集工作。

僵尸进程：一个进程使用fork创建子进程，如果子进程退出，而父进程并没有调用wait或waitpid获取子进程的状态信息，那么子进程的进程描述符仍然保存在系统中。一般会包含defunct字段！

**进程组 会话**

进程必定属于一个进程组，也只能属于一个进程组。 一个进程组中可以包含多个进程。

进程组的生命周期从被创建开始，到其内所有进程终止或离开该组。

当一个用户登录一次系统就形成一次会话 。

一个会话可包含多个进程组，但只能有一个前台进程组。

后台进程的话可以使用a.out & 让进程后台运行！会话结束的时候，进程组也就被终止了！

**进程的详细信息 (/proc/pid)**

**在/proc文件系统中，每一个进程都有一个相应的文件**

**/proc/pid/cmdline 包含用于开始进程的命令**

**/proc/pid/cwd 包含当前进程工作目录的一个链接**

**/proc/pid/environ 包含可用进程环境变量的列表**

**/proc/pid/exe 包含正在进程中运行的程序链接**

**/proc/pid/fd/ 包含进程打开的每一个文件的链接**

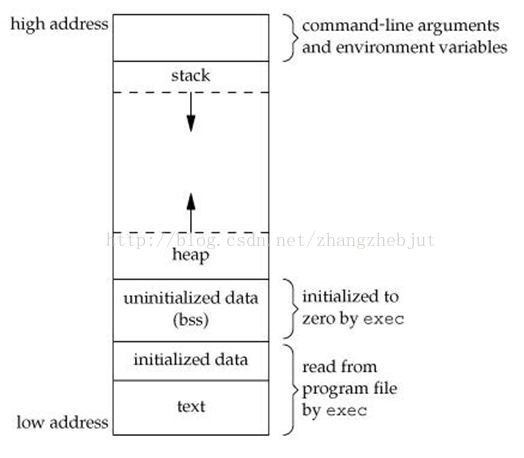
**/proc/pid/mem 包含进程在内存中的内容**

**/proc/pid/stat 包含进程的状态信息**

**/proc/pid/statm 包含进程的内存使用信息**

**/proc/pid/limit 包含进程的软限制，硬限制等信息**

**进程的地址空间&内存布局**



**代码段 : 程序代码在内存中的映射，存放函数体的二进制代码。**

**初始化过的数据 :在程序运行初已经对变量进行初始化的数据。**

**BSS段 :在程序运行初未对变量进行初始化的数据。**

**栈 :存储局部、临时变量，函数调用时，存储函数的返回指针，用于控制函数的调用和返回。在程序块开始时自动分配内存,结束时自动释放内存，先进后出。**

**堆 :存储动态内存分配,需要程序员手工分配,手工释放.注意它与数据结构中的堆是两回事，分配方式类似于链表。**

**查看进程的相关信息**

**查看相应的进程号：ps –ef|grep a.out**

**查看进程打开的文件描述符：lsof –p pid**

**查看进程相关的socket：netstat –anp|grep pid**

**查看进程占用的内存和CPU信息：top –p pid**

**同样的我们也可以通过/proc/pid/\* 获取该进程的详细信息**

**top使用详解:**