

ulimit

控制shell程序的资源

补充说明

ulimit命令 用来限制系统用户对shell资源的访问。如果不懂什么意思，下面一段内容可以帮助你理解：

假设有这样一种情况，当一台 Linux 主机上同时登陆了 10 个人，在系统资源无限制的情况下，这 10 个用户同时打开了 500 个文档，而假设每个文档的大小有 10M，这时系统的内存资源就会受到巨大的挑战。

而实际应用的环境要比这种假设复杂的多，例如在一个嵌入式开发环境中，各方面的资源都是非常紧缺的，对于开启文件描述符的数量，分配堆栈的大小、CPU 时间，虚拟内存大小，等等，都有非常严格的要求。资源的合理限制和分配，不仅仅是保证系统可用性的必要条件，也与系统上软件运行的性能有着密不可分的联系。这时，ulimit 可以起到很大的作用，它是一种简单并且有效的实现资源限制的方式。

ulimit 用于限制 shell 启动进程所占用的资源，支持以下各种类型的限制：所创建的内核文件的大小、进程数据块的大小、Shell 进程创建文件的大小、内存锁住的大小、常驻内存集的大小、打开文件描述符的数量、分配堆栈的最大大小、CPU 时间、单个用户的最大线程数、Shell 进程所能使用的最大虚拟内存。同时，它支持硬资源和软资源的限制。

作为临时限制，ulimit 可以作用于通过使用其命令登录的 shell 会话，在会话终止时便结束限制，并不影响于其他 shell 会话。而对于长期的固定限制，ulimit 命令语句又可以被添加到由登录 shell 读取的文件中，作用于特定的 shell 用户。

语法

ulimit(选项)

选项

- a 显示目前资源限制的设定；
- c <core文件上限>：设定core文件的最大值，单位为区块；
- d <数据节区大小>：程序数据节区的最大值，单位为KB
- e 默认进程优先级, 值越小优先级越高
- f <文件大小>：shell所能建立的最大文件，单位为区块；
- H 设定资源的硬性限制，也就是管理员所设下的限制；
- m <内存大小>：指定可使用内存的上限，单位为KB
- n <文件数目>：指定同一时间最多可开启的文件数；
- p <缓冲区大小>：指定管道缓冲区的大小，单位512字节；
- s <堆叠大小>：指定堆叠的上限，单位为KB
- S 设定资源的弹性限制；
- t <CPU时间>：指定CPU使用时间的上限，单位为秒；
- u <程序数目>：用户最多可开启的程序数目；
- v <虚拟内存大小>：指定可使用的虚拟内存上限，单位为KB

实例

```
[root@localhost ~]# ulimit -a
core file size          (blocks, -c) 0           #core文件的最大值为100 blocks
data seg size           (kbytes, -d) unlimited  #进程的数据段可以任意大。
```

scheduling priority	(-e) 0	
file size	(blocks, -f) unlimited	#文件可以任意大。
pending signals	(-i) 98304	#最多有98304个待处理的信号。
max locked memory	(kbytes, -l) 32	#一个任务锁住的物理内存的最大值为32KB
max memory size	(kbytes, -m) unlimited	#一个任务的常驻物理内存的最大值。
open files	(-n) 1024	#一个任务最多可以同时打开1024的文件。
pipe size	(512 bytes, -p) 8	#管道的最大空间为4096字节。
POSIX message queues	(bytes, -q) 819200	#POSIX的消息队列的最大值为819200字节。
real-time priority	(-r) 0	
stack size	(kbytes, -s) 10240	#进程的栈的最大值为10240字节。
cpu time	(seconds, -t) unlimited	#进程使用的CPU时间。
max user processes	(-u) 98304	#当前用户同时打开的进程（包括线程）的最大个数为98304。
virtual memory	(kbytes, -v) unlimited	#没有限制进程的最大地址空间。
file locks	(-x) unlimited	#所能锁住的文件的最大个数没有限制。

From:

<https://rd.irust.top/> - 学习笔记

Permanent link:

<https://rd.irust.top/doku.php?id=command:ulimit>Last update: **2021/10/15 14:58**