



马哥教育
最专业的Linux培训机构

进程和计划任务

- ❖ 进程相关概念
- ❖ 进程及系统相关工具
- ❖ 计划任务

马哥教育

www.magedu.com

- ❖ 内核的功用：进程管理、文件系统、网络功能、内存管理、驱动程序、安全功能等
- ❖ **Process**：运行中的程序的一个副本，是被载入内存的一个指令集合
进程**ID**（**Process ID**, **PID**）号码被用来标记各个进程
UID、**GID**、和**SELinux**语境决定对文件系统的存取和访问权限，通常从执行进程的用户来继承
存在生命周期
- ❖ **task struct**：Linux内核存储进程信息的数据结构格式
- ❖ **task list**：多个任务的**task struct**组成的链表
- ❖ 进程创建：
 init：第一个进程
 父子关系
 进程：都由其父进程创建，**COW**
 fork(), **clone()**

❖ 进程优先级:

系统优先级: 数字越小, 优先级越高

0-139 (CentOS4,5)

各有**140**个运行队列和过期队列

0-98, 99 (CentOS6)

实时优先级: **99-0**: 值最大优先级最高

Nice值: **-20, 19**对应系统优先级**100-139**或**99**

❖ Big O: 时间复杂度, 用时和规模的关系

$O(1)$, $O(\log n)$, $O(n)$ 线性, $O(n^2)$ 抛物线, $O(2^n)$

❖ 进程内存:

Page Frame: 页框, 用存储页面数据, 存储Page 4k

LRU: Least Recently Used 近期最少使用算法, 释放内存
物理地址空间和线性地址空间

MMU: Memory Management Unit 负责转换线性和物理地址

❖ IPC: Inter Process Communication

同一主机上

signal

shm: shared memory

semaphore 信号量, 一种计数器

不同主机上:

rpc: remote procedure call

socket: IP和端口号

❖ **Linux内核**：抢占式多任务

❖ **进程类型**：

守护进程：**daemon**，在系统引导过程中启动的进程，和终端无关进程

前台进程：跟终端相关，通过终端启动的进程

注意：两者可相互转化

❖ **进程状态**：

运行态：**running**

就绪态：**ready**

睡眠态：

可中断：**interruptable**

不可中断：**uninterruptable**

停止态：**stopped**，暂停于内存中，但不会被调度，除非手动启动

僵死态：**zombie**，结束进程，父进程结束前，子进程不关闭

❖ 进程的分类:

CPU-Bound: CPU密集型, 非交互

IO-Bound: IO密集型, 交互

❖ Linux系统状态的查看及管理工具: `pstree`, `ps`, `pidof`, `pgrep`, `top`, `htop`, `glance`, `pmap`, `vmstat`, `dstat`, `kill`, `pkill`, `job`, `bg`, `fg`, `nohup`

❖ `pstree`命令:

`pstree` - display a tree of processes

❖ `ps`: process state

`ps` - report a snapshot of the current processes

Linux系统各进程的相关信息均保存在`/proc/PID`目录下的各文件中

列举进程ps

❖ 适用ps来查看进程信息

❖ ps [OPTION]...

❖ 支持三种选项：

UNIX选项 如-A -e

BSD选项 如a

GUN选项 如--help

• 默认显示当前终端中的进程

• a 选项包括所有终端中的进程

• x 选项包括不链接终端的进程

• u 选项显示进程所有者的信息

• f 选项显示进程的父进程

• o 属性... 选项显示定制的信息：

➡ pid、comm、%cpu、%mem、state、tty、euser、ruser

- ❖ **VSZ: Virtual memory SiZe**, 虚拟内存集, 线性内存
- ❖ **RSS: ReSident Size**, 常驻内存集
- ❖ **STAT: 进程状态**
 - R: running**
 - S: interruptable sleeping**
 - D: uninterruptable sleeping**
 - T: stopped**
 - Z: zombie**
 - +: 前台进程**
 - l: 多线程进程**
 - N: 低优先级进程**
 - <: 高优先级进程**
 - s: session leader**, 会话（子进程）发起者

❖ 常用组合: **-ef**

-e: 显示所有进程

-f: 显示完整格式程序信息

❖ 常用组合: **-eFH**

-F: 显示更完整格式的进程信息

-H: 以进程层级格式显示进程相关信息

❖ 常用组合: 自定义

-eo pid,tid,class,rtprio,ni,pri,psr,

pcpu,stat,comm

axo stat,euid,ruid,ttty,tpgid,sess,pgrp,ppid,

pid,pcpu,comm

❖ **ni: nice值**

pri: priority, 优先级

psr: processor, CPU编号

rtprio: 实时优先级

马哥教育

www.magedu.com

❖ 最灵活: **ps** 选项 | 其它命令

❖ 按预定义的模式: **pgrep**

pgrep [options] pattern

-u uid: effective user, 生效者

-U uid: real user, 真正发起运行命令者

-t terminal: 与指定终端相关的进程

-l: 显示进程名

-a: 显示完整格式的进程名

-P pid: 显示指定进程的子进程

❖ 按确切的程序名称: **/sbin/pidof**

\$pidof bash

❖ uptime

显示当前时间，系统已启动的时间、当前上线人数，系统平均负载（1、5、10分钟的平均负载，一般不会超过1）

❖ 系统平均负载：

指在特定时间间隔内运行队列中的平均进程数。

- ❖ 如果每个CPU内核的当前活动进程数不大于3的话，那么系统的性能良好。如果每个CPU内核的任务数大于5，那么这台机器的性能有严重问题。
- ❖ 如果linux主机是1个双核CPU的话，当Load Average 为6的时候说明机器已经被充分使用了。

❖ **top**: 有许多内置命令:

排序:

P: 以占据的**CPU**百分比, **%CPU**

M: 占据内存百分比, **%MEM**

T: 累积占据**CPU**时长, **TIME+**

首部信息显示:

uptime信息: **l**命令

tasks及**cpu**信息: **t**命令

cpu分别显示: **1** (数字)

memory信息: **m**命令

退出命令: **q**

修改刷新时间间隔: **s**

终止指定进程: **k**

保存文件: **W**

❖ 栏位信息简介

us: 用户空间

sy: 内核空间

ni: 调整**nice**时间

id: 空闲

wa: 等待**IO**时间

hi: 硬中断

si: 软中断（模式切换）

st: 虚拟机偷走的时间

❖ 栏位信息简介

PID (Process Id):任务的进程ID

PPID (Parent Process Pid):父任务的进程ID

RUSER (Real User Name):任务的所有者真实名称

UID (User Id):任务所有者ID

USER (User Name):任务所有者名称

GROUP (Group Name):任务所有者群组名

TTY (Controlling Tty):终端

PR (Priority):优先级

%CPU (CPU usage):CPU使用率

%MEM (Memory usage (RES)):内存使用率

S (Process Status):进程状态

TIME+ (CPU Time, hundredths):CPU时间, 精确到秒

❖ 选项:

- d #: 指定刷新时间间隔, 默认为3秒
- b: 以批次方式
- n #: 显示多少批次

❖ htop命令: 需从Fedora-EPEL源安装

http://172.16.0.1/fedora-epel/7/x86_64

选项:

- d #: 指定延迟时间;
- u UserName: 仅显示指定用户的进程;
- s COLUMN: 以指定字段进行排序;

子命令:

- s: 跟踪选定进程的系统调用;
- l: 显示选定进程打开的文件列表;
- a: 将选定的进程绑定至某指定CPU核心;
- t: 显示进程树

❖ **vmstat**命令：虚拟内存信息

vmstat [options] [delay [count]]

vmstat 2 5

❖ **procs**:

r: 等待运行的进程的个数，和核心数有关

b: 处于不可中断睡眠态的进程个数(被阻塞的队列的长度)

❖ **memory**:

swpd: 交换内存的使用总量

free: 空闲物理内存总量

buffer: 用于buffer的内存总量

cache: 用于cache的内存总量

❖ **swap**:

si: 从磁盘交换进内存的数据速率(kb/s)

so: 从内存交换至磁盘的数据速率(kb/s)

❖ io:

bi: 从块设备读入数据到系统的速率(kb/s)

bo: 保存数据至块设备的速率

❖ system:

in: interrupts, 中断速率, 包括时钟

cs: context switch, 进程切换速率

❖ cpu:

us: Time spent running non-kernel code

sy: Time spent running kernel code

id: Time spent idle. Linux 2.5.41前, 包括IO-wait time.

wa: Time spent waiting for IO. 2.5.41前, 包括in idle.

st: Time stolen from a virtual machine. 2.6.11前, unknown.

❖ 选项:

-s: 显示内存的统计数据

❖ **pmap**命令：进程对应的内存映射

❖ **pmap [options] pid [...]**

-x: 显示详细格式的信息;

#pmap 1

另外一种实现:

cat /proc/PID/maps

马哥教育

www.magedu.com

- ❖ `glances`命令: EPEL源
- ❖ `glances [-bdehmnrsvyz1] [-B bind] [-c server] [-C conffile] [-p port] [-P password] [--password] [-t refresh] [-f file] [-o output]`
- ❖ 内建命令:
 - a Sort processes automatically
 - c Sort processes by CPU%
 - m Sort processes by MEM%
 - p Sort processes by name
 - i Sort processes by I/O rate
 - d Show/hide disk I/O stats
 - f Show/hide file system stats
 - n Show/hide network stats
 - s Show/hide sensors stats
 - y Show/hide hddtemp stats
 - l Show/hide logs
 - b Bytes or bits for network I/O
 - w Delete warning logs
 - x Delete warning and critical logs
 - 1 Global CPU or per-CPU stats
 - h Show/hide this help screen
 - t View network I/O as combination
 - u View cumulative network I/O
 - q Quit (Esc and Ctrl-C also work)

❖ 常用选项:

- b: 以Byte为单位显示网卡数据速率
- d: 关闭磁盘I/O模块
- f /path/to/somefile: 设定输入文件位置
- o {HTML|CSV}: 输出格式
- m: 禁用mount模块
- n: 禁用网络模块
- t #: 延迟时间间隔
- 1: 每个CPU的相关数据单独显示

www.magedu.com

❖ C/S模式下运行glances命令

❖ 服务模式：

glances -s -B IPADDR

IPADDR: 指明监听的本机哪个地址

❖ 客户端模式：

glances -c IPADDR

IPADDR: 要连入的服务器端地址

www.magedu.com

- ❖ **dstat**命令：系统资源统计
- ❖ **dstat [-afv] [options..] [delay [count]]**
 - c: 显示cpu相关信息
 - C #,#,...,total
 - d: 显示disk相关信息
 - D total,sda,sdb,...
 - g: 显示page相关统计数据
 - m: 显示memory相关统计数据
 - n: 显示network相关统计数据
 - p: 显示process相关统计数据
 - r: 显示io请求相关的统计数据
 - s: 显示swapped相关的统计数据

- ❖ `--tcp`
- ❖ `--udp`
- ❖ `--unix`
- ❖ `--raw`
- ❖ `--socket`
- ❖ `--ipc`
- ❖ `--top-cpu`: 显示最占用**CPU**的进程
- ❖ `--top-io`: 显示最占用**io**的进程
- ❖ `--top-mem`: 显示最占用内存的进程
- ❖ `--top-latency`: 显示延迟最大的进程

❖ kill命令:

向进程发送控制信号，以实现对进程管理

显示当前系统可用信号: **kill -l**

常用信号: **man 7 signal**

1) **SIGHUP**: 无须关闭进程而让其重读配置文件

2) **SIGINT**: 中止正在运行的进程; 相当于**Ctrl+c**

9) **SIGKILL**: 杀死正在运行的进程

15) **SIGTERM**: 终止正在运行的进程

18) **SIGCONT**: 继续运行

19) **SIGSTOP**: 后台休眠

指定信号的方法:

(1) 信号的数字标识; **1, 2, 9**

(2) 信号完整名称; **SIGHUP**

(3) 信号的简写名称; **HUP**

- ❖ 按PID: `kill [-SIGNAL] pid ...`
- ❖ 按名称: `killall [-SIGNAL] comm...`
- ❖ 按模式: `pkill [options] pattern`
 - `-SIGNAL`
 - `-u uid: effective user`, 生效者
 - `-U uid: real user`, 真正发起运行命令者
 - `-t terminal`: 与指定终端相关的进程
 - `-l`: 显示进程名 马哥教育
 - `-a`: 显示完整格式的进程名
 - `-P pid`: 显示父进程为此处指定的进程的进程列表

❖ Linux的作业控制

前台作业：通过终端启动，且启动后一直占据终端；

后台作业：可通过终端启动，但启动后即转入后台运行（释放终端）

❖ 如何让作业运行于后台？

(1) 运行中的作业： **Ctrl+z**

(2) 尚未启动的作业： **# COMMAND &**

❖ 后台作业虽然被送往后台运行，但其依然与终端相关；退出终端，将关闭后台作业。如果希望送往后台后，剥离与终端的关系

nohup COMMAND & #screen;COMMAND

❖ 查看所有作业：

jobs

❖ 作业控制：

fg [%]JOB_NUM: 把指定的后台作业调回前台；

bg [%]JOB_NUM: 让送往后台的作业在后台继续运行；

kill [%JOB_NUM]: 终止指定的作业；

❖ 同时运行多个进程，提高效率

❖ vi all.sh

f1.sh&

f2.sh&

f3.sh&

❖ (f1.sh&):(f2.sh&):(f3.sh&)

❖ { f1.sh& f2.sh& f3.sh& }

马哥教育
www.magedu.com

进程优先级

❖ 进程优先级调整:

静态优先级: 100-139

进程默认启动时的**nice**值为0, 优先级为120

只有根用户才能降低**nice**值 (提高优先性)

❖ **nice**命令:

nice [OPTION] [COMMAND [ARG]...]

❖ **renice**命令:

renice [-n] priority pid...

❖ 查看:

ps axo pid,comm,ni

❖ Linux任务计划、周期性任务执行

- 未来的某时间点执行一次任务：

at

batch: 系统自行选择空闲时间去执行此处指定的任务

- 周期性运行某任务：

cron

马哥教育

www.magedu.com

❖ **at命令:** **at [option] TIME**

❖ **常用选项:**

-V 显示版本信息:

-l: 列出指定队列中等待运行的作业; 相当于**atq**

-d: 删除指定的作业; 相当于**atrm**

-c: 查看具体作业任务

-f /path/from/somefile: 从指定的文件中读取任务

-m: 当任务被完成之后, 将给用户发送邮件, 即使没有标准输出

❖ **注意:** 作业执行命令的结果中的标准输出和错误以邮件通知给相关用户

❖ **TIME:** 定义出什么时候进行 **at** 这项任务的时间

HH:MM [YYYY-mm-dd]

noon, midnight, teatime (4pm)

tomorrow

now+#{minutes, hours, days, OR weeks}

❖ HH:MM 02:00

在今日的 HH:MM 进行，若该时刻已过，则明天此时执行任务

❖ HH:MM YYYY-MM-DD 04:00 2016-09-20

规定在某年某月的某一天的特殊时刻进行该项任务

❖ HH:MM[am|pm] [Month] [Date]

04pm March 17

17:20 tomorrow

❖ HH:MM[am|pm] + number [minutes|hours|days|weeks]

在某个时间点再加几个时间后才进行该项任务

now + 5 minutes

04pm + 3 days

❖ 执行方式:

1) 交互式 2) 输入重定向 3) **at -f** 文件

❖ 依赖与**atd**服务,需要启动才能实现**at**任务

❖ **at**队列存放在**/var/spool/at**目录中

❖ 通过**/etc/at.{allow,deny}**控制用户是否能执行**at**任务

先寻找**/etc/at.allow** 文件,文件中的使用者才能使用 **at** ,
没有在这个文件中的使用者则不能使用 **at** (即使没有写在
at.deny 当中)

如果**/etc/at.allow** 不存在,就寻找 **/etc/at.deny** 这个文件,
若写在这个 **at.deny** 的使用者则不能使用 **at** , 而没有在这个
at.deny 文件中的使用者则可使用 **at** 命令。

如果两个文件都不存在,只有 **root** 可以使用 **at** 这个命令。

周期性任务计划cron

❖ 周期性任务计划：cron

相关的程序包：

cronie：主程序包，提供**crond**守护进程及相关辅助工具

cronie-anacron：**cronie**的补充程序；用于监控**cronie**任务执行状况；如**cronie**中的任务在过去该运行的时间点未能正常运行，则**anacron**会随后启动一次此任务

crontabs：包含CentOS提供系统维护任务

马哥教育

www.magedu.com

- ❖ 确保**crond**守护处于运行状态:

CentOS 7:

`systemctl status crond`

CentOS 6:

`service crond status`

- ❖ 计划周期性执行的任务提交给**crond**, 到指定时间会自动运行

系统**cron**任务: 系统维护作业

`/etc/crontab`

用户**cron**任务:

`crontab`命令

- ❖ 日志: `/var/log/cron`

❖ 系统cron任务:/etc/crontab

❖ 注释行以 # 开头

❖ 详情参见 man 5 crontab

Example of job definition:

.----- minute (0 - 59)

| .----- hour (0 - 23)

| | .----- day of month (1 - 31)

| | | .----- month (1 - 12) OR jan,feb,mar,apr ...

| | | | .----- day of week (0 - 6) (Sunday=0 or 7) OR
sun,mon,tue,wed,thu,fri,sat

| | | | | www.magedu.com

* * * * * user-name command to be executed

❖ 例如：晚上9点10分运行echo命令；

10 21 * * * centos /bin/echo "Howdy!"

- ❖ 时间表示法:
- ❖ (1) 特定值:
 给定时间点有效取值范围内的值
- ❖ (2) *
 给定时间点上有效取值范围内的所有值
 表示“每...”
- ❖ (3) 离散取值:
 #,#,#
- ❖ (4) 连续取值: -
 #-#
- ❖ (5) 在指定时间范围上, 定义步长:
 /#: #即为步长
- ❖ 例如: 每3小时echo命令;
 0 */3 * * * centos /bin/echo "howdy!"

❖ @reboot	Run once after reboot.
❖ @yearly	0 0 1 1 *
❖ @annually	0 0 1 1 *
❖ @monthly	0 0 1 * *
❖ @weekly	0 0 * * 0
❖ @daily	0 0 * * *
❖ @hourly	0 * * * *

马哥教育

www.magedu.com

- 系统的计划任务：

`/etc/crontab`

`/etc/cron.d/` 配置文件

`/etc/cron.hourly/` 脚本

`/etc/cron.daily/` 脚本

`/etc/cron.weekly/` 脚本

`/etc/cron.monthly/` 脚本

www.magedu.com

- ❖ 运行计算机关机时**cron**不运行的任务，**CentOS6.0**以后版本取消**anacron**服务，由**crond**服务管理。
- ❖ 假设计算机没有一直开机
- ❖ 对笔记本电脑、台式机、工作站及其它不一直开机的系统很重要
- ❖ 对偶尔要关机的服务器很有用
- ❖ 配置文件：**/etc/anacrontab**，负责执行**/etc/cron.daily** **/etc/cron.weekly** **/etc/cron.monthly**中系统任务。**/etc/cron.hourly**由**/cron.d/0hourly**执行
 - 字段**1**：如果在这些日子里没有运行这些任务.....
 - 字段**2**：在重新引导后等待这么多分钟后运行它
 - 字段**3**：任务识别器，在日志文件中标识
 - 字段**4**：要执行的任务

- ❖ rhel6.0使用/etc/cron.daily/tmpwatch定时清除临时文件。
- ❖ rhel7.0使用systemd-tmpfiles-setup服务实现
- ❖ 配置文件：
 - /etc/tmpfiles.d/*.conf
 - /run/tmpfiles.d/*.conf
 - /usr/lib/tmpfiles.d/*.conf
- ❖ /usr/lib/tmpfiles.d/tmp.conf
 - d /tmp 1777 root root 10d
 - d /var/tmp 1777 root root 30d
- ❖ 命令：
 - systemd-tmpfiles -clean|remove|create configfile

❖ 用户cron:

❖ crontab命令定义，每个用户都有专用的cron任务文件：
`/var/spool/cron/USERNAME`

❖ crontab命令:

`crontab [-u user] [-l | -r | -e] [-i]`

-l: 列出所有任务;

-e: 编辑任务;

-r: 移除所有任务;

-i: 同-r一同使用，以交互式模式移除指定任务

-u user: 仅root可运行，指定用户管理cron任务

❖ 控制用户执行计划任务:

`/etc/cron.{allow,deny}`

- ❖ 一次性作业使用 **at**
- ❖ 重复性作业使用**crontab**

Create	at time	crontab -e
List	at -l	crontab -l
Details	at -c jobnum	N/A
Remove	at -d jobnum	crontab -r
Edit	N/A	crontab -e

- ❖ 没有被重定向的输出会被邮寄给用户
- ❖ 根用户能够修改其它用户的作业

❖ 注意：运行结果的标准输出和错误以邮件通知给相关用户

(1) `COMMAND > /dev/null`

(2) `COMMAND &> /dev/null`

❖ 对于**cron**任务来讲，**%**有特殊用途；如果在命令中要使用**%**，则需要转义；不过，如果把**%**放置于单引号中，也可以不用转义

马哥教育

www.magedu.com

❖ 思考:

❖ (1) 如何在秒级别运行任务?

```
* * * * * for min in 0 1 2; do echo "hi"; sleep 20;
done
```

❖ (2) 如何实现每7分钟运行一次任务?

❖ sleep命令:

sleep NUMBER[SUFFIX]...

SUFFIX:

s: 秒, 默认

m: 分

h: 小时

d: 天

- ❖ 1、每4小时备份一次/etc目录至/backup目录中，保存的文件名称格式为“etc-yyyy-mm-dd-HH.tar.xz”
- ❖ 2、每周2, 4, 7备份/var/log/messages文件至/logs目录中，文件名形如“messages-yyyymmdd”
- ❖ 3、每两小时取出当前系统/proc/meminfo文件中以S或M开头的信息追加至/tmp/meminfo.txt文件中
- ❖ 4、工作日时间，每10分钟执行一次磁盘空间检查，一旦发现任何分区利用率高于80%，就执行wall警报

马哥教育

www.magedu.com

- ❖ 博客: <http://magedu.blog.51cto.com>
- ❖ 主页: <http://www.magedu.com>
- ❖ QQ: 1661815153, 113228115
- ❖ QQ群: 203585050, 279599283

马哥教育
www.magedu.com



马哥教育
最专业的Linux培训机构

Thank You!