系统进程与计划任务管理

一、系统进程管理

1.程序和进程的关系

当你的软件运行的时候，会打开一个或多个进程

程序>软件

- 程序是静态数据集合
- 进程是动态运行周期
- **进程和程序的区别**
- 父进程--init/systemd
- 守护进程：开机启动，与终端无关，只有关机才停止，无法干预
- 用户进程：通过执行用户程序、应用程序或内核之外的系统程序而产生的进程，可以在用户的控制下运行或关闭

2.ps命令——查看静态的进程统计信息

（report a snapshot of the current processes）

格式： ps aux

显示现行终端机下以用户为主的所有程序

ps a 显示现行终端机下的所有程序，包括其他用户的程序。

ps u 以用户为主的格式来显示程序状况。

ps x 显示所有程序，不以终端机来区分。

ps aux | head

stat命令：（记忆）

- S:可中断的睡眠
- R:就绪或运行状态
- Z:僵死状态
- <:高优先级进程
- N:低优先级进程
- +:前台进程组中的进程

ps aux | grep sshd

ps -C sshd --no-headers | wc -l 筛选，不要表头，不算进程（面试题）

3.top命令——查看进程动态信息（w/uptime）

```
top - 12:58:33 up  3:13,  2 users,  load average: 0.00, 0.01, 0.05
Tasks: 168 total,   1 running, 167 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.3 us,  0.3 sy,  0.0 ni, 99.3 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :   997956 total,    82100 free,   411628 used,   504228 buff/cache
KiB Swap:  2097148 total,  2096884 free,      264 used.   346256 avail Mem
```

第一行：当前时间，系统已运行的时间，当前在线用户，系统平均负载 （面试）

第二行：进程总数量，正在运行（来回切换），167个睡眠进程，0个停止的，0个僵死的进程

第三行：（CPU占用百分比），us(用户空间占用)，sy(内核系统空间占用)，ni(系统优先级切换)

id(idle空闲CPU百分比),wa(wait等待)，hi(硬件占用)，si(软件占用)，st(虚拟化占用)

第四行：总内存空间；已用内存；空闲内存；缓冲区域

第五行：总交换空间；已用交换空间；空闲交换空间；缓存空间

top -b -n 指定循环显示次数

注意：当CPU占用率过高时，不应再直接执行top命令查看，可以将信息存入一个文件内查看，以免CPU占用率过高导致崩溃，操作如下

```
top -b -n1 > /top.txt
cat /top.txt
```

top命令环境中的常用交互命令

P：根据CPU占用百分比排序

M:根据内存排序

T:根据时间排序

k:终止进程

q:退出程序

r:重新安排一个进程的优先级别（-20~19）

进程优先级分类：

ni：nice值，普通优先级，值越低优先级越高

pri: priority 优先级，进程优先级，数值越大优先级越高

4.pgrep命令——查询特定进程信息

-l :显示进程名和PID

-U :指定特定用户

-t :指定终端

[root@localhost ~]# pgrep -l -U test -t tty2

5.pstree命令——用ASCII字符显示树状进程结构（display a tree of processes）

- 显示进程结构，父进程-子进程
- tree /boot/ :查看目录树形结构

6.终止进程的运行

dd if=/dev/zeo of=/file bs=1M count=1024

1.Ctrl+c #终止正在执行的进程

2.kill、killall命令

- 9 强制删除，强行停止进程
- 15 和谐删除（默认）

ps aux #命令
kill PID号（15 默认信号）
kill -9 PID号
进程号要看清楚，别删错了

3.指定名字
ping 127.0.0.1
ps
killall ping（后面跟名字杀进程）

4.指定条件
指定用户/终端
pgrep -9 -U test -t tty1
pgrep -l -U test -t tty1
pkill -9 -U test -t tty1

5.top命令里面的 k 命令
指定PID号 -9强制杀死

二、任务管理

前台进程：占用命令窗口吗
后台进程：ping 127.0.0.1 >> ping.txt &
只要命令后加&就属于后台运行
注意：坑坑坑！！！！终端关闭，保留任务，后台还在运行
不管是前台还是后台的任务，都是和窗口共存的，窗口没 了，任务也没了

1. nohup命令

---永久运行进程
nohup ping 127.0.0.1 >> /ping.txt &
防止任务还没做完，终端退出造成的影响

dd if /dev/zero of=/file bs=1M count=8192

ctrl+z 放后台，停止状态

jobs： 查看后台任务 -l 列出ID 及信息

fg 1 ： 调回前台运行　　1是编号

bg 1 ： 调回后台运行

2.计划任务

1.at一次性任务设置

atd 服务

2.at命令的使用

- Ctrl+D 保存退出
- atq :查看任务 = at -l
- atrm： 删除 at -d
- at -c 1　　#查看任务1的详细信息

```
[root@localhost ~]# atrm 2
[root@localhost ~]# atq
1       Thu Mar  5 05:30:00 2020 a root
[root@localhost ~]# at 22:00 +7 days            # 时间
at> systenctl restart network                           #任务
at> <EOT>                                                       #Ctrl+D
job 3 at Thu Mar 12 22:00:00 2020
[root@localhost ~]# atq
1       Thu Mar  5 05:30:00 2020 a root
3       Thu Mar 12 22:00:00 2020 a root
[root@localhost ~]# ls /tmp/ps.txt
```

ps:
网络时间同步 ： ntpdate time.windows.com
手动设置 ： date -s "YYYY-mm-dd HH:MM:SS"

3.crontab周期性任务设置

1.简述 crontab

crontab 是用来让使用者在固定时间或固定间隔执行程序之用，换句话说，也就是类似使用者的时程表。

-u user 是指设定指定 user 的时程表，这个前提是你必须要有其权限(比如说是 root)才能够指定他人的时程表。如果不使用 -u user 的话，就是表示设定自己的时程表。

[root@localhost ~]# ls /etc/cron.　　#两次TAB键

[root@localhost ~]# cat /etc/crontab
SHELL=/bin/bash　　#设置执行计划任务的shell环境
PATH=/sbin:/bin:/usr/sbin:/usr/bin　　#定义可执行命令及程序的绝对路径　　当它查找命令的时候只从这四个目录下查找命令，所以你的路径如果在别的路径当中的时候，可能无法识别；如果是这四个路径下的命令可以不用写绝对路径
MAILTO=root　　#将任务输出信息发送到指定用户的邮箱
HOME=/　　#执行计划任务时使用的主目录

2. crontab 文件的含义
 # 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
 # | | | | |
 # * * * * * user-name command to be executed　命令写绝对路径

星号（*）：代表所有可能的值，例如month字段如果是星号，则表示在满足其它字段的制约条件后每月都执行该命令操作。
逗号（,）：可以用逗号隔开的值指定一个列表范围，例如，“1,2,5,7,8,9”
中杠（-）：可以用整数之间的中杠表示一个整数范围，例如“2-6”表示“2,3,4,5,6”
正斜线（/）：可以用正斜线指定时间的间隔频率，例如“0-23/2”表示每两小时执行一次。同时正斜线可以和星号一起使用，例如*/10，如果用在minute字段，表示每十分钟执行一次。

3./var/spool/cron/文件

由用户自行设置的cron计划任务将被保存到目录/var/spool/cron/中，文件名与用户名相同。

[root@localhost ~]# ls -l /var/spool/cron/

总用量 4

-rw-------. 1 root root 61 3月 3 13:05 root

计划任务很重要，tar打包，备份很重要！！！！！

4.参数

crontab

-e ： 执行文字编辑器来设定时程表

-r ： 删除目前的时程表

-l :查看目前的时程表

-u<用户名称>：指定要设定计时器的用户名称。

```
[root@localhost ~]# crontab -e
crontab: installing new crontab
"/tmp/crontab.WsqVwT":2: bad minute
errors in crontab file, can't install.
Do you want to retry the same edit? n
crontab: edits left in /tmp/crontab.WsqVwT
[root@localhost ~]# cat /var/spool/cron/root
* */5 * * * /usr/sbin/ntpdate
pool.ntp.org > /dev/null 2>&1
[root@localhost ~]# vim /var/spool/cron/root
```

crontab -r 命令 ： 可以删除用户的计划任务列表 #最好别直接用-r,清空就啥都没了

PS:

1.配置完计划任务后建议重启计划任务服务

vim /var/spool/cron/root

systemctl restart crond

2.计划任务信息要定时备份

给用户test编辑计划任务
方法 一：[root@localhost ~]# crontab -e -u test
方法二：[root@localhost ~]# vim /var/spool/cron/test

5.练习题
00 8 /2 * * 1-5 df -Th
00 8,5 * * 6,7 df -Th

三、系统日志

系统日志是记录系统中硬件、软件和系统问题的信息，同时还可以监视系统中发生的事件。用户可以通过它来检查错误发生的原因，或者寻找受到攻击时攻击者留下的痕迹。

系统日志包括系统日志、应用程序日志和安全日志,用户日志

1.分类

1、应用程序日志：记录应用程序或一般程序的事件（软件、网站、DNS）。

2、记录linux用户登录及退出系统的相关信息，包括用户名、登录的终端、登录时间、来源主机、正在使用的进程操作等。

3、安全性日志：可以记录例如有效和无效的登录尝试等安全事件，以及与资源使用有关的事件。例如创建、打开或删除文件以及有关设置的修改。

4、系统日志：包含由linux系统组件记录的事件，例如，在系统日志中记录启动期间要加载的驱动程序或其他系统组件的故障

5、系统邮件信息服务相关的日志

6、与定时任务相关的日志文件

2.rsyslog服务（采集和存放日志）

rsyslog是一个开源工具，被广泛用于linux系统以通TCP/UDP协议转发或接受日志文进阿消息。

rsyslog服务的主配置文件为 **/etc/rsyslog.conf**，指定日志保存位置修改配置文件，修改后重启rsyslog服务生效。

- ".":比后面等级要高（包含该等级）的都记录。例如："*.info"
- ".=":只记录该等级。例如：".=debug"
- "!":除了该等级都记录--- !info
- "-":当有记录信息需要记录时，先存到缓存中，到一定大小时一次性写入，以减少对磁盘读写性能的占用。例 如"-/var/log/maillog"

3.日志文件位置

- **占用空间大**
- **打开日志----/var/log/**
- **/var/log/的权限最好不要改，记录的密码**
- **软件/服务并不是只生成一个日志文件，会有多个文件，会在/var/log/下创建一个子目录，拿子目录来存放多个文件**

4.常见日志文件及查看方式

日志文件	存放内容
/var/log/messages	内核消息及各种应用程序的公共日志信息，包括启动、I/O错误、网络错误、程序故
/var/log/cron	crond周期性计划任务产生的时间信息
/var/log/dmesg	引导过程中的各种时间信息
/var/log/maillog	进入或发出系统的电子邮件活动
/var/log/lastlog	每个用户最近的登录事件
/var/log/secure	用户认证相关的安全事件信息
/var/log/wtmp	每个用户登录、注销及系统启动和停机事件
/var/log/btmp	失败的、错误的登录尝试及验证事件

/var/log/secure ----安全防护！！！防止被破解密码

5.日志消息的级别

数字等级越小，优先级越高，消息越重要

级别	英文表示	意义
0	EMERG(紧急)	会导致主机系统不可用的情况
1	ALERT(警告)	必须马上采取措施解决的问题
2	CRITICAL(严重)	比较严重的情况
3	ERROR(错误)	运行出现错误

err crit 简写

6.日志文件分析

1.内核及大多数系统消息

最近日志消息过滤一下

grep -E -iw "emerg | alert | critical | error" /var/log/messages

2.用户日志

存放位置:

/var/log/wtmp、/var/log/btmp、/var/log/lastlog

不是普通文本文件，不能用cat查看，只能用命令

查询命令:

users、who、w、last、lastlog、lastb等

last：成功登陆的信息

lastlog：最近一次登录的信息

lastb：登陆失败的信息

3.程序日志

在Linux操作系统中，还有一部分应用程序没有使用rsyslog服务来管理日志，而是由程序自己维护日志记录。例如：httpd网站服务程序使用两个日志文件access_log和error_log分别记录客户访问事件和错误事件，不同应用程序的日志记录格式差别大，没有严格使用统一的记录格式。

7.日志文件分析注意事项

总的来说，作为一名合格的系统管理人员，应该提高警惕，随时注意各种可疑的状况，定期并随机的检查各种系统日志文件， 包括一般信息日志、网络连接日志、文件传输日志及用户登录日志记录等。在检查这些日志时，要注意是否有不合常理的时间或操作记录。

注意！！！

- 用户在非常规的时间登录，或者用户登录系统的IP地址和以往不一样
- 用户登陆失败的日志记录，尤其是那些一再连续尝试进入失败的日志记录
- 非法使用或不正当使用超级用户权限
- 无故或者非法重启各项网络服务的记录
- 不正常的日志记录，如日志残缺不全，或者像wtmp这样的日志文件无辜缺少了中间的记录文件
- 日志并不是完全可靠的，高明的黑客在入侵系统后经常会打扫现场，管理人员需要运用综上的所有知识，全面、综合的进行审查和检测。

日志管理

1. 针对日志定期备份、异地备份（保留1~3个月的日志记录）
2. 针对日志定期切割（0 0 * * * mv /var/log/messages /var/log/messages-$(date -d "-1 days"+%F)）
3. 针对日志的权限要严格（为了防止敏感信息泄露）
4. 针对日志做集中管理

8.对日志文件的保护

格式 ： chatter+a 日志文件

a选项：append(追加) only ,给日志文件加上a的权限后，将只可以追加，不可以删除和修改之前的内容。

查看

```
[root@localhost ~]# lsattr /var/log/secure
---------------- /var/log/secure
[root@localhost ~]# chattr +a /var/log/secure                    #防止别人删除文件，注意别让人拿到管理员权限
[root@localhost ~]# lsattr /var/log/secure
-----a---------- /var/log/secure
[root@localhost ~]# rm -rf /var/log/secure
rm: 无法删除"/var/log/secure": 不允许的操作
[root@localhost ~]# > /var/log/secure
-bash: /var/log/secure: 不允许的操作
```

root用户在vi编辑器的命令模式按dd删除了两行内容后进行wq! 强制保存退出，也会报错

递归式增加a权限 chattr +a -R

```
[root@localhost ~]# chattr +a -R /var/log/
[root@localhost ~]# lsattr -R /var/log/
```