

02

OPEN ORIENTED

凹凸实验室

Crontab 快速上车

@2017.09.04

路面畅通!
可以随时开车!
over!



大纲

- 介绍
- 例子
- 调试

jiè shào

介绍

crontab

cron 源于希腊语 chronos(χρνο), 原意是时间, tab 即 table

Linux 下的任务调度分为两类

- **系统任务调度** 系统周期性所要执行的工作，比如写缓存数据到硬盘、日志清理等。 [/etc/crontab](#)
- **用户任务调度** 用户定期要执行的工作，比如用户数据备份、定时邮件提醒等。 [/var/spool/cron/<username>](#)

crond

yum install crontabs # 安装 crontab

service crond start # 启动服务

service crond stop # 关闭服务

service crond restart # 重启服务

service crond reload # 重新载入配置

service crond status # 查看 crontab 服务状态

ntsysv # 查看 crontab 服务是否已设置为开机启动

chkconfig --level 35 crond on # 加入开机自动启动

`crontab [-e [UserName] | -l [UserName] | -r [UserName] | -v [UserName] | file]`

`-e` 编辑某个用户的 cron 服务

`-l` 列出某个用户 cron 服务的详细内容

`-r` 删除某个用户的 cron 服务

`-v` 列出用户 cron 作业的状态

`-i` 会和 `-r` 配合使用，在删除当前的 crontab 时询问，输入 y 则删除

`-u` 可选，省略则表示操作当前用户的 crontab

`file` 命令文件的名字,表示将 file 做为 crontab 的任务列表文件并载入 crontab

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



```
$ cat /etc/crontab
```

```
# 省略其他内容
```

```
59 23 * * * root cd /export/servers/nginx/sbin;./logrotate.sh
```




minute 表示分钟，可以是 0-59 之间的任何整数

hour 表示小时，可以是 0-23 之间的任何整数

day 表示日期，可以是 1-31 之间的任何整数

month 表示月份，可以是 1-12 之间的任何整数

week 表示星期几，可以是 0-6 之间的任何整数，这里的 0 代表星期日

command 要执行的命令，可以是系统命令，也可以是自己编写的脚本文件

* */2 2-6 1,2,5 mon-wed

星号 (***) 代表所有的取值范围内的数字

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。

逗号 (*,*) 可以用逗号隔开的值指定一个列表范围，例如："1,2,5"

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"

run-parts

每小时执行 /etc/cron.hourly 内的脚本

01 * * * * root run-parts /etc/cron.hourly

每天执行 /etc/cron.daily 内的脚本

02 4 * * * root run-parts /etc/cron.daily

每星期执行 /etc/cron.weekly 内的脚本

22 4 * * 0 root run-parts /etc/cron.weekly

每月去执行 /etc/cron.monthly 内的脚本

42 4 1 * * root run-parts /etc/cron.monthly

lì zǐ

例子

一些典型的例子

每天早上6点10分

10 6 * * *

每两个小时

0 */2 * * *

晚上11点到早上8点之间每两个小时，早上8点

0 23-7/2, 8 * * *

每个月的4号和每个礼拜一到礼拜三的早上11点

0 11 4 * 1-3

1月1日早上4点

0 4 1 1 *

一些典型的例子

30 21 * * *

45 4 1,10,22 * *

10 1 * * 6,0

0,30 18-23 * * *

0 23 * * 6

* */1 * * *

* 23-7/1 * * *

* 8,13 * * 1-5

0 11 4 * mon-wed

0 4 1 jan *

一些典型的例子

30 21 * * * # 表示每晚的 21:30

45 4 1,10,22 * * # 表示每月1、10、22日的 4:45

10 1 * * 6,0 # 表示每周六、周日的1:10

0,30 18-23 * * * # 表示每天18:00至23:00之间每隔 30 分钟

0 23 * * 6 # 表示每星期六的 11:00 pm

* */1 * * * # 每一小时

* 23-7/1 * * * # 晚上 11 点到早上 7 点之间，每隔一小时

* 8,13 * * 1-5 # 从周一到周五的上午 8 点和下午 1 点

0 11 4 * mon-wed # 每月的 4 号与每周一到周三的 11 点

0 4 1 jan * # 一月一号的 4 点

tiáo shì

调试

- 先手动执行脚本确保正常
- 脚本中涉及文件路径时写全局路径
- 确保环境变量引入没问题（脚本的开头加载 `source / etc/profile`）
- 新创建的 cron job，不会马上执行，至少要过 2 分钟才执行，如果重启 cron 则马上执行

确保 crontab 已启动

\$ service crond status

查看 log 确保计划是否执行

\$ tail -f /var/log/cron

查看 mail 确保是否正确执行

\$ tail -f /var/spool/mail/<username>

错误输出到日志

0 6 * * * /sh.sh 2 >> log.log



都上车了吗？

THANKS
FOR YOUR WATCHING

02

OPEN ORIENTED

凹凸实验室