[**crontab 详细用法 定时任务**](https://www.cnblogs.com/aminxu/p/5993769.html)

转自：http://blog.chinaunix.net/uid-25785357-id-3434344.html

使用crontab你可以在指定的时间执行一个shell脚本或者一系列Linux命令。例如系统管理员安排一个备份任务使其每天都运行

如何往 cron 中添加一个作业?

# crontab –e  
0 5 \* \* \* /root/bin/backup.sh

这将会在每天早上5点运行 /root/bin/backup.sh

Cron 各项的描述

以下是 crontab 文件的格式：

{minute} {hour} {day-of-month} {month} {day-of-week} {full-path-to-shell-script}   
o minute: 区间为 0 – 59   
o hour: 区间为0 – 23   
o day-of-month: 区间为0 – 31   
o month: 区间为1 – 12. 1 是1月. 12是12月.   
o Day-of-week: 区间为0 – 7. 周日可以是0或7.

Crontab 示例

1. 在 12:01 a.m 运行，即每天凌晨过一分钟。这是一个恰当的进行备份的时间，因为此时系统负载不大。

1 0 \* \* \* /root/bin/backup.sh

2. 每个工作日(Mon – Fri) 11:59 p.m 都进行备份作业。

59 11 \* \* 1,2,3,4,5 /root/bin/backup.sh

下面例子与上面的例子效果一样：

59 11 \* \* 1-5 /root/bin/backup.sh

3. 每5分钟运行一次命令

\*/5 \* \* \* \* /root/bin/check-status.sh

4. 每个月的第一天 1:10 p.m 运行

10 13 1 \* \* /root/bin/full-backup.sh

5. 每个工作日 11 p.m 运行。

0 23 \* \* 1-5 /root/bin/incremental-backup.sh

Crontab 选项

以下是 crontab 的有效选项:

o crontab –e : 修改 crontab 文件. 如果文件不存在会自动创建。   
o crontab –l : 显示 crontab 文件。   
o crontab -r : 删除 crontab 文件。  
o crontab -ir : 删除 crontab 文件前提醒用户。

以上就是crontab命令的具体使用方法了。

在linux平台上如果需要实现任务调度功能可以编写cron脚本来实现。  
以某一频率执行任务  
linux缺省会启动crond进程，crond进程不需要用户启动、关闭。  
crond进程负责读取调度任务并执行，用户只需要将相应的调度脚本写入cron的调度配置文件中。  
cron的调度文件有以下几个：  
  
   1. crontab  
   2. cron.d  
   3. cron.daily  
   4. cron.hourly  
   5. cron.monthly  
   6. cron.weekly   
  
如果用的任务不是以hourly monthly weekly方式执行，则可以将相应的crontab写入到crontab 或cron.d目录中。  
示例：  
每隔一分钟执行一次脚本 /opt/bin/test-cron.sh  
可以在cron.d新建脚本 echo-date.sh  
内容为  
\*/1 \* \* \* \* root  /opt/bin/test-cron.sh  
在指定的时间运行任务  
也可以通过at命令来控制在指定的时间运行任务  
如：  
at -f test-cron.sh -v 10:25  
其中-f 指定脚本文件， -v 指定运行时间  
quote:ea946d690b="lophyxp"]首先用  
contab -l >contabs.tmp  
导出contab的配置。  
然后编辑contabs.tmp文件。以一下格式添加一行：  
分钟 小时 天 月 星期 命令  
比如  
10 3 \* \* 0,6 hello  
就是每周六、周日的3点10分执行hello程序。  
15 4 \* \* 4-6 hello  
就是从周四到周六的4点15点执行hello程序。  
然后用  
contab contabs.tmp  
命令导入新的配置。  
一般不建议直接修改/etc/下的相关配置文件。  
启动cron进程的方法：/etc/init.d/crond start  
开机就启动cron进程的设置命令：chkconfig --add crond  
方法二：  
把cron加入到启动脚本中：  
# rc-update add vixie-cron default  
crontab -l #查看你的任务  
crontab-e#编辑你的任务  
crontab-r#删除用户的crontab的内容  
实例讲解二：  
系统cron设定：/etc/crontab  
    通过 /etc/crontab 文件，可以设定系统定期执行的任务，当然，要想编辑这个文件，得有root权限  
0 7   \*    \*   \*    root    mpg123 ~/wakeup.mp3  
分 时 日 月 周  
示例：  
0 4  \* \* 0     root emerge --sync && emerge -uD world              #每周日凌晨4点，更新系统  
0 2 1 \* \*     root   rm -f /tmp/\*                                                    #每月1号凌晨2点，清理/tmp下的文件  
0 8 6 5 \*   root     mail  robin < /home/galeki/happy.txt             #每年5月6日给robin发信祝他生日快乐  
假如，我想每隔2分钟就要执行某个命令，或者我想在每天的6点、12点、18点执行命令，诸如此类的周期，可以通过 “ / ” 和 “ , ” 来设置：  
\*/2   \*   \*   \*   \*           root      ...............      #每两分钟就执行........  
0 6,12,18   \*   \*   \*    root      ...............      #每天6点、12点、18点执行........  
每两个小时  
0 \*/2 \* \* \* echo "have a break now." >> /tmp/test.txt  
晚上11点到早上8点之间每两个小时，早上八点  
0 23-7/2，8 \* \* \* echo "have a good dream：）" >> /tmp/test.txt  
每个月的4号与每个礼拜的礼拜一到礼拜三的早上11点  
0 11 4 \* 1-3 command line  
1月1日早上4点  
0 4 1 1 \* command line  
收获：可以把经常要做的一些事放到其中，简化工作量，如每周一检查服务器的运行状态，查看报告，杀掉一些进程等等……  
  
  
\*　　\*　　\*　　\*　　\*　　command  
分　时　日　月　周　命令  
第1列表示分钟1～59 每分钟用\*或者 \*/1表示  
第2列表示小时1～23（0表示0点）  
第3列表示日期1～31  
第4列表示月份1～12  
第5列标识号星期0～6（0表示星期天）  
第6列要运行的命令  
crontab文件的一些例子：  
30 21 \* \* \* /usr/local/etc/rc.d/lighttpd restart  
上面的例子表示每晚的21:30重启apache。  
45 4 1,10,22 \* \* /usr/local/etc/rc.d/lighttpd restart  
上面的例子表示每月1、10、22日的4 : 45重启apache。  
10 1 \* \* 6,0 /usr/local/etc/rc.d/lighttpd restart  
上面的例子表示每周六、周日的1 : 10重启apache。  
0,30 18-23 \* \* \* /usr/local/etc/rc.d/lighttpd restart  
上面的例子表示在每天18 : 00至23 : 00之间每隔30分钟重启apache。  
0 23 \* \* 6 /usr/local/etc/rc.d/lighttpd restart  
上面的例子表示每星期六的11 : 00 pm重启apache。  
\* \*/1 \* \* \* /usr/local/etc/rc.d/lighttpd restart  
每一小时重启apache  
\* 23-7/1 \* \* \* /usr/local/etc/rc.d/lighttpd restart  
晚上11点到早上7点之间，每隔一小时重启apache  
0 11 4 \* mon-wed /usr/local/etc/rc.d/lighttpd restart  
每月的4号与每周一到周三的11点重启apache  
0 4 1 jan \* /usr/local/etc/rc.d/lighttpd restart  
一月一号的4点重启apache  
名称 : crontab  
使用权限 : 所有使用者  
使用方式 :  
crontab file [-u user]-用指定的文件替代目前的crontab。  
crontab-[-u user]-用标准输入替代目前的crontab.  
crontab-1[user]-列出用户目前的crontab.  
crontab-e[user]-编辑用户目前的crontab.  
crontab-d[user]-删除用户目前的crontab.  
crontab-c dir- 指定crontab的目录。  
crontab文件的格式：M H D m d cmd.  
M: 分钟（0-59）。  
H：小时（0-23）。  
D：天（1-31）。  
m: 月（1-12）。  
d: 一星期内的天（0~6，0为星期天）。  
cmd要运行的程序，程序被送入sh执行，这个shell只有USER,HOME,SHELL这三个环境变量  
说明 :  
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。  
参数 :  
crontab -e : 执行文字编辑器来设定时程表，内定的文字编辑器是 VI，如果你想用别的文字编辑器，则请先设定 VISUAL 环境变数  
来指定使用那个文字编辑器(比如说 setenv VISUAL joe)  
crontab -r : 删除目前的时程表  
crontab -l : 列出目前的时程表  
crontab file [-u user]-用指定的文件替代目前的crontab。  
时程表的格式如下 :  
f1 f2 f3 f4 f5 program  
其中 f1 是表示分钟，f2 表示小时，f3 表示一个月份中的第几日，f4 表示月份，f5 表示一个星期中的第几天。program 表示要执  
行的程序。  
当 f1 为 \* 时表示每分钟都要执行 program，f2 为 \* 时表示每小时都要执行程序，其馀类推  
当 f1 为 a-b 时表示从第 a 分钟到第 b 分钟这段时间内要执行，f2 为 a-b 时表示从第 a 到第 b 小时都要执行，其馀类推  
当 f1 为 \*/n 时表示每 n 分钟个时间间隔执行一次，f2 为 \*/n 表示每 n 小时个时间间隔执行一次，其馀类推  
当 f1 为 a, b, c,... 时表示第 a, b, c,... 分钟要执行，f2 为 a, b, c,... 时表示第 a, b, c...个小时要执行，其馀类推  
使用者也可以将所有的设定先存放在档案 file 中，用 crontab file 的方式来设定时程表。  
例子 :  
#每天早上7点执行一次 /bin/ls :  
0 7 \* \* \* /bin/ls  
在 12 月内, 每天的早上 6 点到 12 点中，每隔3个小时执行一次 /usr/bin/backup :  
0 6-12/3 \* 12 \* /usr/bin/backup  
周一到周五每天下午 5:00 寄一封信给 [alex@domain.name](mailto:alex@domain.name) :  
0 17 \* \* 1-5 mail -s "hi" [alex@domain.name](mailto:alex@domain.name) < /tmp/maildata  
每月每天的午夜 0 点 20 分, 2 点 20 分, 4 点 20 分....执行 echo "haha"  
20 0-23/2 \* \* \* echo "haha"  
注意 :  
当程序在你所指定的时间执行后，系统会寄一封信给你，显示该程序执行的内容，若是你不希望收到这样的信，请在每一行空一格之  
后加上 > /dev/null 2>&1 即可  
例子2 :  
#每天早上6点10分  
10 6 \* \* \* date  
#每两个小时  
0 \*/2 \* \* \* date  
#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 \* \* \* date  
#每个月的4号和每个礼拜的礼拜一到礼拜三的早上11点  
0 11 4 \* mon-wed date  
#1月份日早上4点  
0 4 1 jan \* date  
范例  
$crontab -l 列出用户目前的crontab.