

执行一次的任务计划

安装

在CentOS 6.x 和 CentOS 7.x 版本的最小化安装中默认没有安装 atd 服务。

```
1 yum install at
```

atd服务开启及开机启动

```
1 #CentOS 6.x
2 service atd stop #停止atd服务
3 service atd start #启动atd服务
4 service atd restart #重启atd服务
5 service atd status #查看atd服务运行状态
6 chkconfig atd on #设定开机启动
7
8 #CentOS 7.x
9 systemctl stop atd.service #停止atd服务
10 systemctl start atd.service #启动atd服务
11 systemctl restart atd.service #重启atd服务
12 systemctl status atd.service #查看atd服务运行状态
13 systemctl enable atd.service #设定开机启动
```

atd工作方式

at指令产生的任务以文本的方式写入 `/var/spool/at/` 目录内，atd服务按照这个目录内的记录执行安排的任务。但为了安全，并不是所有用户都有权限去执行这个指令。

```
1 [ localhost@root ~]#ll -d /etc/at*
2 -rw-r--r--. 1 root root 0 1月 10 16:41 at.allow
3 -rw-r--r--. 1 root root 1 11月 6 02:16 at.deny
```

在 `at.allow` 和 `at.deny` 分别记录着允许执行 `at` 指令的用户名和不允许执行 `at` 指令的用户名，每一个用户名写在一行。但 `atd` 服务是优先按照 `at.allow` 文件的规则来执行的，CentOS 发行版中，在 `/etc/` 目录下默认没有 `at.allow` 这个文件，如果有需要可以自行创建。即如果没有 `at.allow` 文件，则写入 `at.deny` 文件内的用户不能执行 `at` 命令，如果有 `at.allow` 文件，则只有写入该文件内的用户才有权限执行 `at` 命令。

创建及管理 `at` 任务计划

```
1 #格式
2 at [optin] time|work_id
```

```
1 #option
2 -m: 无论有没有输出信息都以mail的形式通知用户
3 -l: =atq。列出档当前系统的计划任务
4 -d: =atrm。取消一个计划任务，后跟work_id
5 -c: 显示执行该任务计划的命令，后跟work_id
```

```
1 #time。相对时间 | 绝对时间
2 #绝对时间
3 at HH: MM
4 at HH: MM YYYY: MM: DD
5
6 #相对时间
7 at now +N {minutes|hours|days|weeks}
```

周期性任务计划

`cron` 是一个强大的任务计划程序。为了系统的安全，`cron` 和 `at` 有着同样的机制来避免用户滥用计划任务。`/etc/cron.deny` 作为黑名单和 `/etc/cron.allow` 作为白名单。

在 CentOS 上 `cron` 以一个名称为 `crond` 的守护进程提供服务，在使用 `cron` 做计划任务时，要确保 `crond` 服务开启且是开机启动状态。

```
1 #CentOS 6.x
2 service crond start
3 chkconfig crond on
4
5 #CentOS 7.x
6 systemctl start crond.service
7 systemctl enable crond.service
```

用户任务计划

用户执行的任务计划，记录在 `/var/spool/cron/` 目录下以用户名命名的文件单独存在。但不建议直接编辑此目录下的文件，建议使用 `crontab` 命令，以避免语法错误。

```
1 #crontab命令
2 -e: 编辑任务计划
3 -l: 列出任务计划
4 -r: 移除所有任务计划
```

```
1 #格式
2 * * * * * command
3
4 #示例
5 #每分钟执行一次
6 0-59 * * * * command
7
8 #每小时执行一次
9 01 * * * * command
10
11 #每小时的第1分钟和第5分钟执行一次
12 01,05 * * * * command
13
14 #每天上午8-11点的第3和15分钟执行
15 03,15 8-11 * * * command
16
17 #每隔2天的上午8-11点的第3和15分钟执行
18 03,15 8-11 */2 * * command
19
```

```
20 #每个星期一的上午8点到11点的第3和第15分钟执行
21 3,15 8-11 * * 1 command
22
23 #每晚的21:30执行
24 30 21 * * * command
25
26 #每月1、10、22日的4 : 45执行
27 45 4 1,10,22 * * command
28
29 #每周六、周日的1 : 10执行
30 10 01 * * 6,7 command
31
32 #每天18 : 00至23 : 00之间每隔30分钟执行
33 00,30 18-23 * * * command
34
35 #晚上11点到早上7点之间，每隔一小时执行
36 * 23-7/1 * * * command
37
38 #每月的4号与每周一到周三的11点执行
39 00 11 4 * 1-3 command
```

系统任务计划

```

1 [xuekaixin@root /etc/cron.d]#ls -ld /etc/cron*
2 drwxr-xr-x. 2 root root 4096 Jan 10 20:19
  /etc/cron.d #系统任务计划的配置文件也可以单独放置在此目录下
3 drwxr-xr-x. 2 root root 4096 Dec 30 05:24
  /etc/cron.daily #CentOS默认每小时执行此目录下的所有可执行
  脚本
4 -rw-----. 1 root root 0 Mar 30 2015
  /etc/cron.deny #cron使用权限黑名单设置文件
5 -rw-r--r--. 1 root root 457 Sep 27 2011
  /etc/crontab #系统任务计划的配置文件
6 drwxr-xr-x. 2 root root 4096 Sep 27 2011
  /etc/cron.hourly
7 drwxr-xr-x. 2 root root 4096 Sep 27 2011
  /etc/cron.monthly
8 drwxr-xr-x. 2 root root 4096 Sep 27 2011
  /etc/cron.weekly

```

系统任务计划可以直接写入 `crontab` 文件内，也可以写成单独的文件放到 `cron.d` 目录下。系统任务配置文件的格式为：

```

1 SHELL=/bin/bash
2 PATH=/sbin:/bin:/usr/sbin:/usr/bin
3 MAILTO=root
4 * * * * * user_name command
5 01 * * * * root run-parts /etc/cron.hourly #每小时执
  行此目录下的所有脚本，按脚本名称排序然后依次执行

```

cron日志查看

```
1 tail -f /var/log/cron
```

关于环境变量

如果我们写了一个脚本放到cron中定时执行时，会经常无法执行的情况。是因为cron默认不会读取环境变量，所以需要在脚本手动读取环境变量。在被执行的脚本中添加：

```
1  #! /bin/bash
2  . /etc/profile
3  . ~/.bash_profile
```

可唤醒停机期间的服务

在制定了任务计划，但因为机器故障或关机等原因，到时任务未能执行，这时候 `anacron` 程序会去执行到时应该执行但未执行的任务。

`anacron` 是一支程序，被 `cron` 每小时执行一次。`anacron` 的执行脚本文件是 `/etc/cron.hourly/0anacron`，此脚本中执行 `anacron` 命令。