

Linux基础 - 定时任务

目录

01

什么是定时任务?

02

如何配置定时任务?

目录

01

什么是定时任务?

02

如何配置定时任务?

什么是定时任务？

- Linux定时任务是指一种在特定时间或时间间隔内执行特定命令或脚本的机制。
- 常用的Linux定时任务主要有以下两种：
 - atd定时任务
 - crond定时任务
- at和cron定时任务的主要区别是：
 - atd是一次性任务，在设定的特定时间内执行一次。
 - crond是周期性任务，在设定的时间周期内重复执行。

什么是定时任务？

■ 为什么要使用定时任务？

■ 定时任务有以下优点：

| 优点 | 说明 |
|---------|---|
| 自动化执行 | 定时任务可以自动执行一些常规性、重复性的任务。 |
| 可靠性高 | 定时任务可以在指定的时间内自动执行，不需要人工干预，避免了因人为原因而出现的误操作，提高了操作的可靠性 |
| 灵活性强 | 定时任务的执行时间、执行频率等可以自由设置，可以根据不同的业务需求和情况进行调整 |
| 节省时间和成本 | 通过定时任务可以自动完成一些重复的、耗时的工作，减少了人工操作的时间和成本 |

什么定时任务?

■ 定时任务的使用场景都有哪些?

■ 定时任务可以应用到以下场景:

| 场景 | 说明 |
|----------|---|
| 定时数据备份 | 自动备份数据库、文件等数据, 确保数据的安全性 |
| 定时发送邮件 | 定时发送日报、周报、月报等, 减轻了人工操作的负担。 |
| 定时执行清理工作 | 可以定期清理一些不必要的数据、日志等, 释放服务器的空间和资源。 |
| 定时检查系统状态 | 可以定期检测系统的状态, 如 CPU 使用率、内存使用率等, 确保系统的正常运行。 |

目录

01

什么是定时任务?

02

如何配置定时任务?

如何配置定时任务?

- atd一次性定时任务配置, 主要使用at命令进行配置
- at命令使用语法如下:

```
at <options> <time>
```

| 选项 | 说明 |
|-----------|---------------------|
| -m | 当指定的任务完成后, 给用户发送邮件 |
| -l | atq的别名 |
| -d | atrm的别名 |
| -v | 显示任务执行的详细信息 |
| -c | 打印任务内容到标准输出 |
| -q <队列> | 使用指定的队列 |
| -f <文件> | 从指定文件读入任务而不是从标准输入读入 |
| -t <时间参数> | 以时间参数的形式提交要运行的任务 |

如何配置定时任务?

- atd一次性定时任务配置实例
 - 实例一：2分钟执行ls命令并把命令执行结果输出到/tmp/ls.txt文件，4分钟后删除/tmp/ls.txt文件

```
[root@rocky ~]# at now +2 minutes
at> ls -al > /tmp/ls.txt
at> <EOT>
job 5 at Sat Feb 18 15:34:00 2023
[root@rocky ~]# at now +4 minutes
at> rm -rf /tmp/ls.txt
at> <EOT>
job 6 at Sat Feb 18 15:36:00 2023
[root@rocky ~]# atq
5          Sat Feb 18 15:34:00 2023 a root
6          Sat Feb 18 15:36:00 2023 a root
[root@rocky ~]#
```

```
[root@rocky ~]# atq
6          Sat Feb 18 15:36:00 2023 a root
[root@rocky ~]# ls /tmp/ls.txt
/tmp/ls.txt
[root@rocky ~]# ls /tmp/ls.txt
ls: 无法访问/tmp/ls.txt: 没有那个文件或目录
[root@rocky ~]# atq
[root@rocky ~]#
```

如何配置定时任务?

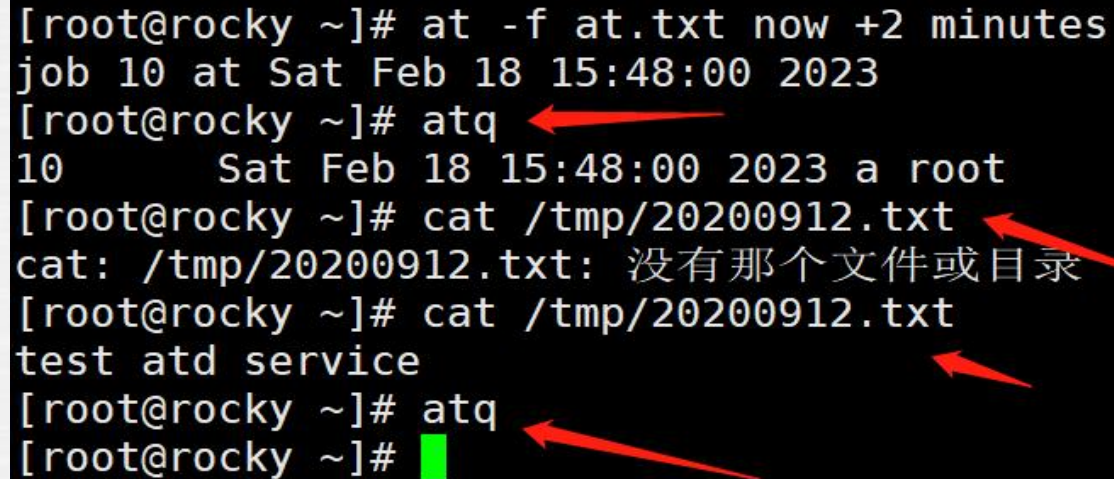
■ atd一次性定时任务配置实例

■ 实例二：从指定文件读入定时任务，而不是从标准输入读入定时任务

■ 指定文件内容为：

```
echo 'test atd service'>/tmp/20200912.txt
```

```
[root@rocky ~]# at -f at.txt now +2 minutes  
job 10 at Sat Feb 18 15:48:00 2023  
[root@rocky ~]# atq  
10      Sat Feb 18 15:48:00 2023 a root  
[root@rocky ~]# cat /tmp/20200912.txt  
cat: /tmp/20200912.txt: 没有那个文件或目录  
[root@rocky ~]# cat /tmp/20200912.txt  
test atd service  
[root@rocky ~]# atq  
[root@rocky ~]# █
```



如何配置定时任务?

- cron周期性定时任务配置, 主要使用crontab命令配置
- crontab命令使用语法如下:

```
crontab <options>  
crontab <options> <file>
```

| 选项 | 说明 |
|-----------|--|
| -u <user> | 设定某个用户的crontab服务 |
| -e | 编辑某个用户的crontab文件内容 |
| -l | 显示某个用户的crontab文件内容 |
| -r | 从/var/spool/cron目录中删除某个用户的crontab文件 |
| -i | 在删除用户的crontab文件时给确认提示 |
| file | file是命令文件的名字,表示将file做为crontab的任务列表文件并载入crontab |

如何配置定时任务?

■ crontab配置文件的格式

```
* * * * * command to be executed
- - - - -
| | | | |
| | | | | +----- day of the week (0 - 6) (Sunday=0)
| | | | | +----- month (1 - 12)
| | | | | +----- day of the month (1 - 31)
| | | | | +----- hour (0 - 23)
| | | | | +----- min (0 - 59)
```

- 第1列分钟0 ~ 59
- 第2列小时0 ~ 23 (0表示子夜)
- 第3列日1 ~ 31
- 第4列月1 ~ 12
- 第5列星期0 ~ 7 (0和7表示星期天)
- 第6列要运行的命令

如何配置定时任务？

■ crontab配置文件的格式说明

■ 每一分钟执行脚本

```
* * * * * myCommand
```

■ 每小时的第3和第15分钟执行脚本

```
3,15 * * * * myCommand
```

■ 每周一上午8点到11点的第3和第15分钟执行

```
3,15 8-11 * * 1 myCommand
```

如何配置定时任务?

■ crontab周期性定时任务配置实例

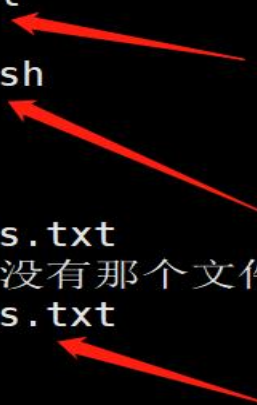
■ 实例一：每一分钟执行一次test.sh脚本

```
* * * * * /root/test.sh
```

```
[root@rocky ~]# crontab -e  
crontab: installing new crontab  
[root@rocky ~]#
```

```
* * * * * /root/test.sh  
~
```

```
[root@rocky ~]# crontab -l  
* * * * * /root/test.sh  
[root@rocky ~]# cat test.sh  
#!/bin/bash  
  
ls -al > /tmp/ls.txt  
[root@rocky ~]# ls /tmp/ls.txt  
ls: 无法访问/tmp/ls.txt: 没有那个文件或目录  
[root@rocky ~]# ls /tmp/ls.txt  
/tmp/ls.txt  
[root@rocky ~]#
```



The background is a solid red color with a white circuit board pattern. The pattern consists of various lines, including straight lines, right-angle turns, and some circular components, resembling a printed circuit board (PCB) layout.

谢谢