

第1章 定时任务介绍

1.什么是定时任务

- 1 | 简单来说，就是某些我们需要定时执行的任务，可以是某条命令，也可以是某个脚本。

2.定时任务使用场景

- 1 | 1.按天日志切割
- 2 | 2.定时同步互联网时间
- 3 | 3.每天备份数据库的数据
- 4 | 4.定时删除不需要的日志文件或临时文件
- 5 | 5.定时获取系统的状态信息

第2章 定时任务实践

1.定时任务语法

定时任务配置文件：

```
1 [root@linux ~]# cat /etc/crontab
2 SHELL=/bin/bash                                #执行命令的解释器
3 PATH=/sbin:/bin:/usr/sbin:/usr/bin             #环境变量
4 MAILTO=root                                     #发送给指定用户
   邮件
5
6 # For details see man 4 crontabs
7
8 # Example of job definition:
9 # .----- minute (0 - 59)      #分钟
10 # | .----- hour (0 - 23)     #小时
11 # | | .----- day of month (1 - 31) #日期
12 # | | | .----- month (1 - 12) OR jan,feb,mar,apr ... #月
13 # | | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR
   sun,mon,tue,wed,thu,fri,sat
14 # | | | | | #星期
15 # * * * * * user-name command to be executed
```

简单说明：分时日月周

1	*	*	*	*	*	user	command
2	分钟	小时	日期	月份	星期	执行用户	执行命令

间隔符号：

- | | | |
|---|---|-----------------------------------|
| 1 | * | 星号表示任意，最小单位，每分/每小时/每周/每月 |
| 2 | - | 表示一个时间段范围，例如8-10点 |
| 3 | , | 表示分隔，表示每当，如果日期举例：2,4,6表示每月的2号4号6号 |
| 4 | / | 表示每隔多久，如果是小时，*/2 表示每隔2小时 |

2.定时任务时间表达练习

1	00 00 * * *	每天0点执行
2	00 01 * * *	每天凌晨1点30分执行
3	* * * * *	每分钟都执行
4	00 * * * *	每小时执行
5	00 */2 * * *	每隔2小时执行
6	*/30 * * * *	每隔30分钟执行
7	00 01 15 * *	每个月的15号1点执行
8	00 05 1-14 * *	每个月1-14号凌晨5点执行
9	00 6-8 */5 * *	每隔5天的上午6点到8点执行
10	00 20-23/1 * * *	晚上8-11点之间，每隔1小时执行
11	00 23 * * 1-3	周一到周三每天晚上11点执行

3.定时任务配置管理

命令格式：

```
1 | crontab [options]
```

常用参数：

```
1 | -e      #编辑定时任务
2 | -l      #查看定时任务
3 | -r      #删除定时任务
4 | -u      #指定其他用户的定时任务
```

操作步骤：

```
1 | #1.编辑定时任务,操作命令和VIM一样
2 | [root@linux ~]# crontab -e
3 | * * * * * /usr/sbin/ntpdate time1.aliyun.com > /dev/null 2>&1
4 |
5 | #2.查看定时任务
6 | [root@linux ~]# crontab -l
7 | * * * * * /usr/sbin/ntpdate time1.aliyun.com > /dev/null 2>&1
8 |
9 | #3.删除定时任务
10 | [root@linux ~]# crontab -r
11 |
12 | #4.再次查看定时任务
13 | [root@linux ~]# crontab -l
14 | no crontab for root
```

4.定时任务实战练习

练习1:每分钟和阿里云时间服务器同步一下时间

```
1 | * * * * * /usr/sbin/ntpdate time1.aliyun.com > /dev/null 2>&1
```

练习2:每天凌晨1点执行数据库的备份任务--模拟输出即可

1 |

练习3:每周日执行一次删除7天以前日志的任务

1 |

练习4: 备份任务，需求如下

- 1.每天备份一次/etc目录到/backup目录下
- 2.备份格式为2021-04-07_hostname_etc.tar.gz
- 3.将执行的结果保存到日志里/tmp/backup.log
- 4.要求备份文件只保留3天，多余的删掉

1 |

第4章 定时任务调试

1.工作中定时任务的坑

坑1:没打日志，不知道定时任务有没有执行

正确姿势：

- 1 定时任务执行过程最好输出到日志里，这样有助于排查问题，如果不需要输出，可以定向到/dev/null。
- 2 定时任务不需要输出到屏幕上。

坑2:命令行可以执行，但是放进定时任务里执行不成功

正确姿势：

- 1 定时任务可识别的PATH变量有限，最佳实践是将要执行的命令放进脚本里，然后定时任务执行这个脚本。
- 2 还有一点需要注意的就是，定时任务要执行的命令，最好使用绝对路径，这样就不会因为PATH变量不全导致执行不了命令了。

坑3:时间久了，不知道这条定时任务是干嘛的了

- 1 定时任务一定要加注释，不然容易被人打

2.调试定时任务的思路

- 1 1.先把定时任务的命令在命令终端执行测试好，确定没问题之后再写进定时任务里
- 2 2.调试的时候可以将执行过程详细的输出到日志里，方便排查问题
- 3 3.定时任务执行的命令使用绝对路径
- 4 4.如果是执行的脚本，最好加上可执行权限
- 5 5.调试的阶段可以缩短定时任务执行的间隔，方便测试出结果
- 6 6.最后别忘了给定时任务添加注释。方便自己，也方便他人阅读

