

## nginx:按天自动切日志并压缩

默认情况下，nginx将每天网站访问的日志都写在一个文件里面，随着时间的推移，这个文件势必越来越大，最终成为问题。不过我们可以写个脚本来自动地按天(或者小时)切割日志，并压缩(节约磁盘空间)。

脚本的内容如下：

```
#!/bin/bash

logspath="/alidata/log/Nginx/access/"
yesterday=`date -d '-1 day' +%Y%m%d`
mv ${logspath}www.iteblog.com.access.log ${logspath}www.iteblog.com.access$yesterday.log
mv ${logspath}www.iteblog.com.error.log ${logspath}www.iteblog.com.error$yesterday.log
gzip ${logspath}www.iteblog.com.access$yesterday.log
gzip ${logspath}www.iteblog.com.error$yesterday.log

kill -USR1 `cat /alidata/server/Nginx/logs/Nginx.pid`
```

上面的程序是一个bash脚本。logspath是我网站日志存放的决定路径；yesterday是昨天的时间，格式是%Y%m%d；首先将当前nginx服务器正在写的日志移到带日期的文件中，然后调用gzip分别对access以及error日志进行压缩，压缩完的格式为gz。

这里需要注意的是，mv完日志之后，一定需要调用kill -USR1 `cat /alidata/server/nginx/logs/nginx.pid`，否则mv之后网站的日志将不会写入到\${logspath}www.iteblog.com.access.log文件中。

这个脚本可以放到一个目录下面，这里假设放到了我home目录下/home/iteblog，名称是mvlogs.sh。我们需要定时的调用这个脚本才能切日志，可以利用Linux自带的定时功能crontab -e进行设置，内容如下：

```
00 00 * * * /bin/sh /home/iteblog/mvlogs.sh
```

crontab配置计划任务的书写格式

分钟 小时 日月周 [用户名] 命令

说明：

第一个参数定义的是：分钟，表示每个小时的第几分钟来执行。范围是从0-59

第二个参数定义的是：小时，表示从第几个小时来执行，范围是从0-23

第三个参数定义的是：日期，表示从每个月的第几天执行，范围从1-31

第四个参数定义的是：月，表示每年的第几个月来执行，范围从1-12

第五个参数定义的是：周，表示每周的第几天执行，范围从0-6，其中 0表示星期日。

每六个参数定义的是：用户名，也就是执行程序要通过哪个用户来执行，这个一般可以省略

;

第七个参数定义的是：执行的命令和参数。

设置完成之后，这个脚本就会在每天的00:00进行切日志操作。下面是我网站切除日志的格式

```
[root@iteblog access]# ll
total 1516
-rw-r--r-- 1 www root 203120 Jan  2 00:00 www.iteblog.com.access20150101.log.gz
-rw-r--r-- 1 www www 202423 Jan  2 23:59 www.iteblog.com.access20150102.log.gz
-rw-r--r-- 1 www www 33593 Jan  3 00:12 www.iteblog.com.access.log
-rw-r--r-- 1 www root 3533 Jan  1 23:52 www.iteblog.com.error20150101.log.gz
-rw-r--r-- 1 www root 2494 Jan  2 23:44 www.iteblog.com.error20150102.log.gz
-rw-r--r-- 1 www root 0 Jan  3 00:00 www.iteblog.com.error.log
```

本博客文章除特别声明，全部都是原创！

原创文章版权归过往记忆大数据（[过往记忆](#)）所有，未经许可不得转载。

本文链接: [【】](#) ( )