

二

## 27 crontab 与 PurgeTxnLog: 线上系统日志清理的最佳时间和方式

---

本节课，我们主要学习对线上 ZooKeeper 服务器日志进行维护的操作，主要维护方式是备份和清理。几乎所有的生产系统都会产生日志文件，用来记录服务的运行状态，在服务发生异常的时候，可以用来作为分析问题原因的依据。ZooKeeper 作为分布式系统下的重要组件，在分布式网络中会处理大量的客户端请求，因此也会产生大量的日志文件，对这些问题的维护关系到整个 ZooKeeper 服务的运行质量。接下来我们就来学习如何维护这些日志文件。

### 日志类型

首先，我们先来介绍线上生产环境中的 ZooKeeper 集群在对外提供服务的过程中，都会产生哪些日志类型。我们在之前的课程中也介绍过了，在 ZooKeeper 服务运行的时候，一般会产生数据快照和日志文件，数据快照用于集群服务中的数据同步，而数据日志则记录了 ZooKeeper 服务运行的相关状态信息。其中，数据日志是我们在生产环境中需要定期维护和管理文件。

### 清理方案

如上面所介绍的，面对生产系统中产生的日志，一般的维护操作是备份和清理。备份是为了之后对系统的运行情况进行排查和优化，而清理主要因为随着系统日志的增加，日志会逐渐占用系统的存储空间，如果一直不进行清理，可能耗尽系统的磁盘存储空间，并最终影响服务的运行。但在实际工作中，我们不能 24 小时监控系统日志情况，因此这里我们介绍一种定时任务，可以自动清理和备份 ZooKeeper 服务运行产生的相关日志。

### 清理工具

#### corntab

首先，我们介绍的是 Linux corntab，它是 Linux 系统下的软件，可以自动地按照我们设定的时间，周期性地执行我们编写的相关脚本。下面我们就用它来写一个定时任务，实现每周

定期清理 ZooKeeper 服务日志。

## 创建脚本

我们通过 Linux 系统下的 Vim 文本编辑器，来创建一个叫作 “logsCleanWeek” 的定时脚本，该脚本是一个 shell 格式的可执行文件。如下面的代码所示，我们在 usr/bin/ 文件夹下创建该文件，该脚本的主要内容是设定 ZooKeeper 快照和数据日志的对应文件夹路径，并通过 shell 脚本和管道和 find 命令 查询对应的日志下的日志文件，这里我们保留最新的 10 条数据日志，其余的全部清理。

```
#!/bin/bash

dataDir=/home/zk/zk_data/version-2

dataLogDir=/home/zk/zk_log/version-2

ls -t $dataLogDir/log.* | tail -n +$count | xargs rm -f

ls -t $dataDir/snapshot.* | tail -n +$count | xargs rm -f

ls -t $logDir/zookeeper.log.* | tail -n +$count | xargs rm -f

find /home/home/zk/zk_data/version-2 -name "snap*" -mtime +1 | xargs rm -f

find /home/home/zk/zk_data/version-2 -name "snap*" -mtime +1 | xargs rm -f

find /home/home/zk/zk_data/logs/ -name "zookeeper.log.*" -mtime +1 | xargs rm -f
```

## 创建定时任务

创建完定时脚本后，我们接下来就利用 corntab 来设置脚本的启动时间，如下面的代码所示。corntab 命令的语法比较简单，其中 -u 表示设定指定的用户，因为 Linux 系统是一个多用户操作系统，而 crontab 的本质就是根据使用系统的用户来设定程序执行的时间计划表。因此当命令的执行者具有管理员 root 账号的权限时，可以通过 -u 为特定用户设定某一个程序的具体执行时间。

```
crontab [ -u user ] { -l | -r | -e }
```

接下来我们打开系统的控制台，并输入 crontab -e 命令，开启定时任务的编辑功能。如下图所示，系统会显示出当前已有的定时任务列表。整个 crontab 界面的操作逻辑和 Vim 相同，为了新建一个定时任务，我们首先将光标移动到文件的最后一行，并敲击 i 键来开启编辑模式。

```
*/1 * * * * /usr/local/qcloud/stargate/admin/start.sh > /dev/null 2>&1 &
0 0 * * * /usr/local/qcloud/YunJing/YDCrontab.sh > /dev/null 2>&1 &
```

@拉勾教育

©拉勾教育

12/21/2022, 6:26 PM

候调用 ZooKeeper 为我们提供的工具类完成日志清理工作。

如下面的代码所示, 首先, 我们在 /usr/bin 目录下创建一个 PurgeLogsClean 脚本。注意这里的脚本也是一个 shell 文件。在脚本中我们只需要编写 PurgeTxnLog 类的调用程序, 系统就会自动通过 PurgeTxnLog 工具类为我们完成对应日志文件的清理工作。

```
#!/bin/sh

java -cp "$CLASSPATH" org.apache.zookeeper.server.PurgeTxnLog

echo "清理完成"
```

PurgeTxnLog 方式与 crontab 相比, 使用起来更加容易而且也更加稳定安全, 不过 crontab 方式更加灵活, 我们可以根据不同的业务需求编写自己的清理逻辑。

## 结束

本节课我们介绍了线上 ZooKeeper 服务日志和数据快照的清理和维护工作, 可以通过 crontab 和 PurgeTxnLog 两种方式实现。这两种方式唯一的不同在清理日志脚本的实现方式上, crontab 是通过我们自己手动编写的 shell 脚本实现的, 在执行上需要考虑脚本权限相关的问题, 而 PurgeTxnLog 则是 ZooKeeper 提供的专门用来处理日志清除相关的工具类, 使用起来更加容易, 开发人员不用考虑底层的实现细节。这里希望你结合自身工作中的生产环境来选择一种适合自己的 ZooKeeper 数据维护方式。

[上一页](#)

[下一页](#)