



链滴

zookeeper 的单机多实例配置和分布式配置

作者: [zwxbest](#)

原文链接: <https://ld246.com/article/1544446644867>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

单机多实例配置:

配置文件相同部分

```
cp /data/apache/zookeeper/conf/zoo_sample.cfg /data/apache/zookeeper/conf/zoo.cfg
vi /data/apache/zookeeper/conf/zoo.cfg
```

配置文件内容:

```
tickTime=2000
initLimit=5
syncLimit=5
dataDir=/data/apache/zookeeper/data # 目录需要手工建立, 存放 zk 数据, 主要是快照
clientPort=2181 # dataLogDir事务日志存放目录, 最好配置, 事务日志的写入速度严重影响zookee
er的性能
dataLogDir=/data/apache/zookeeper/datalog
server.1=192.168.130.170:2889:3889
server.2=192.168.130.170:2890:3890
server.3=192.168.130.170:2891:3891
```

配置文件不同部分

生成三个配置文件: zoo-slave1.cfg: (另外两个 zoo-slave2.cfg 和 zoo-slave3.cfg) 。

zoo-slave1.cfg需要为dataDir和dataLogDir设置目录, 改动内容如下:

```
dataDir=/opt/zookeeper-3.4.8/zkdata/slave1
dataLogDir=/opt/zookeeper-3.4.8/zkdatalog/slave1
clientPort=2182
```

zoo-slave2.cfg :

```
dataDir=/opt/zookeeper-3.4.8/zkdata/slave2
dataLogDir=/opt/zookeeper-3.4.8/zkdatalog/slave2
clientPort=2183
```

zoo-slave3.cfg :

```
dataDir=/opt/zookeeper-3.4.8/zkdata/slave3
dataLogDir=/opt/zookeeper-3.4.8/zkdatalog/slave3
clientPort=2184
```

如上配置相同的本机IP, 不同的端口号, 这里配置了三个实例

id配置

如何区分到底是第几个实例呢, 就要有个id文件, 名字必须是myid

```
echo "1" > /opt/zookeeper-3.4.8/zkdata/slave1/myid
echo "2" > /opt/zookeeper-3.4.8/zkdata/slave2/myid
echo "3" > /opt/zookeeper-3.4.8/zkdata/slave3/myid
```

启动

在三个窗口下迅速启动：

```
bin/zkServer.sh start zoo-slave1.cfg
bin/zkServer.sh start zoo-slave2.cfg
bin/zkServer.sh start zoo-slave3.cfg
```

查看zookeeper选出来的leader，通过下面的脚本，分别指定配置文件，就可以查看哪一个实例是leader：

```
bin/zkServer.sh status zoo-slave1.cfg
```

如果提示Error contacting service. It is probably not running，用start-foreground启动，输出详细信息

我的就是JAVA_HOME没配置对

没问题的话，可以看下如下输出信息：

```
[root@mj1 bin]# sh zkServer.sh status zoo-slave1.cfg
ZooKeeper JMX enabled by default
Using config: /opt/zookeeper-3.4.8/bin/../conf/zoo-slave1.cfg
Mode: follower
[root@mj1 ~]# sh zkServer.sh status zoo-slave2.cfg
ZooKeeper JMX enabled by default
Using config: /opt/zookeeper-3.4.8/bin/../conf/zoo-slave2.cfg
Mode: leader
[root@mj1 ~]# sh zkServer.sh status zoo-slave3.cfg
ZooKeeper JMX enabled by default
Using config: /opt/zookeeper-3.4.8/bin/../conf/zoo-slave3.cfg
Mode: follower
```

注意在启动时，由于不一定是哪个zookeeper做leader，所以启动过程中日志文件会有错误，不用管等到所有zookeeper启动了之后，选出了leader，一切就都ok了。

分布式部署：

与单机多实例差不多，不过不用三个目录，只需要一个目录，一个配置文件，三个不同IP，分发不同的IP上，仍然是不同的myid。假设我有三台独立机器，那么zoo.cfg（只需要一个默认的配置文件的oo.cfg）配置文件如下所示：

```
tickTime=2000
initLimit=5
syncLimit=5
dataDir=/data/apache/zookeeper/data # 目录需要手工建立，存放 zk 数据，主要是快照
clientPort=2181 # dataLogDir事务日志存放目录，最好配置，事务日志的写入速度严重影响zookeeper的性能
dataLogDir=/data/apache/zookeeper/datalog
server.1=192.168.130.101:2888:3888
server.2=192.168.130.102:2888:3888
server.3=192.168.130.103:2888:3888
```

然后在三台机器上分别写入不同的myid：

```
#在192.168.130.101执行下面一条
echo "1" > /data/apache/zookeeper/data/myid
#在192.168.130.102执行下面一条
echo "2" > /data/apache/zookeeper/data/myid
#在192.168.130.103执行下面一条
echo "3" > /data/apache/zookeeper/data/myid
```

这样分别在三台机器迅速启动zookeeper，此时不需要指定配置文件，默认配置文件就是zoo.cfg:

```
bin/zkServer.sh start
```

查看启动后的状态:

```
bin/zkServer.sh status
```

进入zookeeper的shell客户端:

```
bin/zkCli.sh #进入本机zookeeper的shell
```

进入指定机器的shell客户端:

```
bin/zkCli.sh -server 192.168.130.101
```

在查看启动状态有错误的情况下，稍等一会再查看，因为zookeeper启动后需要一点时间选出leader。