## Zookeeper 是什么

Zookeeper是⼀个开源的分布式协调服务，其设计⽬标是将那些复杂的且容易出错的分布式⼀致性服务封装起来。zookeeper是⼀个典型 的分布式数据⼀致性的解决⽅案，分布式应⽤程序可以基于它实现诸如数据订阅/发布、负载均衡、命名 服务、集群管理、分布式锁和分布式队列等功能 .zookeeper 有几个基本概念。

①会话（session） Session指客户端会话，⼀个客户端连接是指客户端和服务端之间的⼀个TCP⻓连接，Zookeeper对外的 服务端⼝默认为2181，客户端启动的时候，⾸先会与服务器建⽴⼀个TCP连接，从第⼀次连接建⽴开 始，客户端会话的⽣命周期也开始了，通过这个连接，客户端能够⼼跳检测与服务器保持有效的会话， 也能够向Zookeeper服务器发送请求并接受响应，同时还能够通过该连接接受来⾃服务器的Watch事件 通知。 ② 数据节点（Znode） 在谈到分布式的时候，我们通常说的“节点”是指组成集群的每⼀台机器。然⽽，在ZooKeeper中，“节 点”分为两类，第⼀类同样是指构成集群的机器，我们称之为机器节点；第⼆类则是指数据模型中的数据 单元，我们称之为数据节点——ZNode。ZooKeeper将所有数据存储在内存中，数据模型是⼀棵树 （ZNode Tree），由斜杠（/）进⾏分割的路径，就是⼀个Znode，例如/app/path1。每个ZNode上都 会保存⾃⼰的数据内容，同时还会保存⼀系列属性信息。

③ Watcher（事件监听器） Wathcer（事件监听器），是Zookeeper中⼀个很重要的特性，Zookeeper允许⽤户在指定节点上注册 ⼀些Watcher，并且在⼀些特定事件触发的时候，Zookeeper服务端会将事件通知到感兴趣的客户端， 该机制是Zookeeper实现分布式协调服务的重要特性。

## Zookeeper的搭建

项目中，zookeeper都采用的是伪集群模式搭建的，也就是在⼀台服务器上运⾏多个Zookeeper 实例，项目中使用了3个zookeeper实例。Zookeeper伪集群的搭建流程如下：

1.下载 ⾸先我们下载最新稳定版本的zookeeper http://zookeeper.apache.org/releases.html 2.上传 下载完成后，将zookeeper压缩包 zookeeper- -3.7.0-tar.gz上传到linux系统

3. 解压 压缩包. 项目中 创建⽬录apache-zookeeper-3.7.0-bin。

tar -zxvf zookeeper-3.4.14.tar.gz -C / 【将压缩包解压到根目录下】

mv zookeeper-3.4.14 apache-zookeeper-3.7.0-bin【重命名】

4. 配置多个服务器实例

1.在zookeeper根目录下创建多个实例根目录

mkdir 2181

mkdir 2182

mkdir 2183

2.分别进入 2181，2182，2183目录，在实例目录下创建 data 文件夹，log文件夹，conf文件夹。

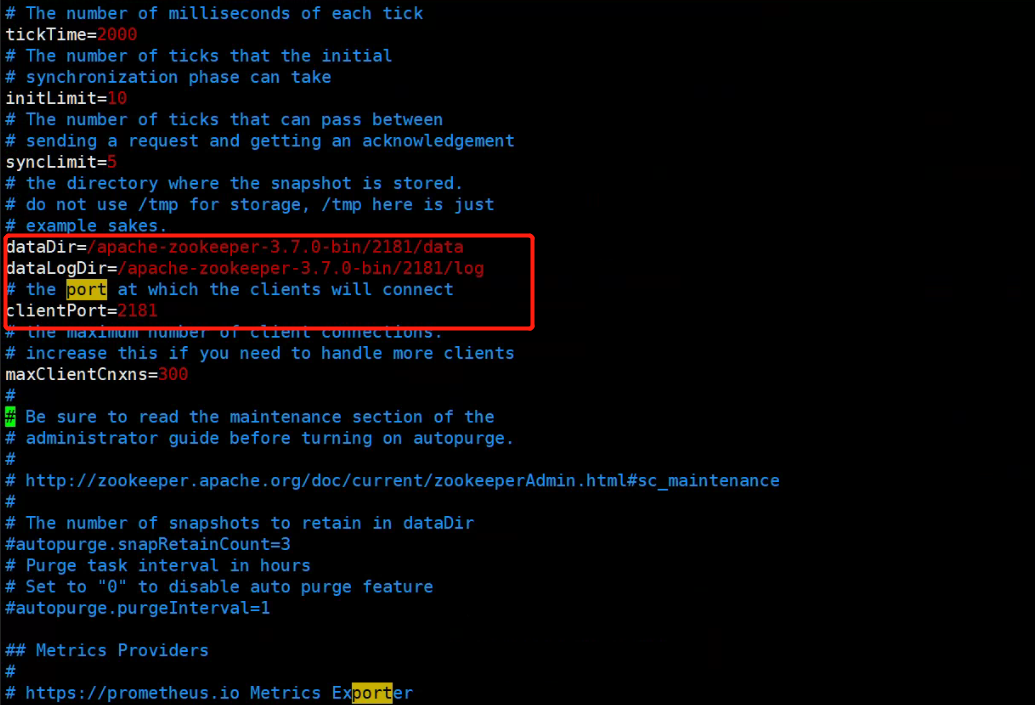
mkdir data

mkdir logs

mkdir conf

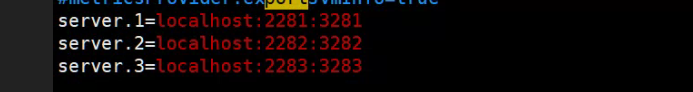
3. 将/apache-zookeeper-3.7.0-bin/conf/zoo\_sample.cfg文件拷贝到 3个根目录下的conf文件夹下。并将文件重命名为 zoo. cfg.

4. 在zookeepe的配置文件中将dataDir（zoo.cfg） clientPort 分别设置为2181 2182 2183.下图为/apache-zookeeper-3.7.0-bin/2181/conf/zoo. cfg 文件的部分内容。



5.在每个zookeeper的 data ⽬录下创建⼀个 myid ⽂件，内容分别是1、2、3 。这个⽂件中的数字就是每个服务器的ID

6.在每⼀个zookeeper 的 zoo.cfg配置客户端访问端⼝（clientPort）和集群服务器IP列表。 #server.服务器ID=服务器IP地址：服务器之间通信端⼝：服务器之间投票选举端⼝



启动集群 依次启动三个zk实例 ，其中2181端口实例的执行启动命令为

cd /apache-zookeeper-3.7.0-bin/bin/

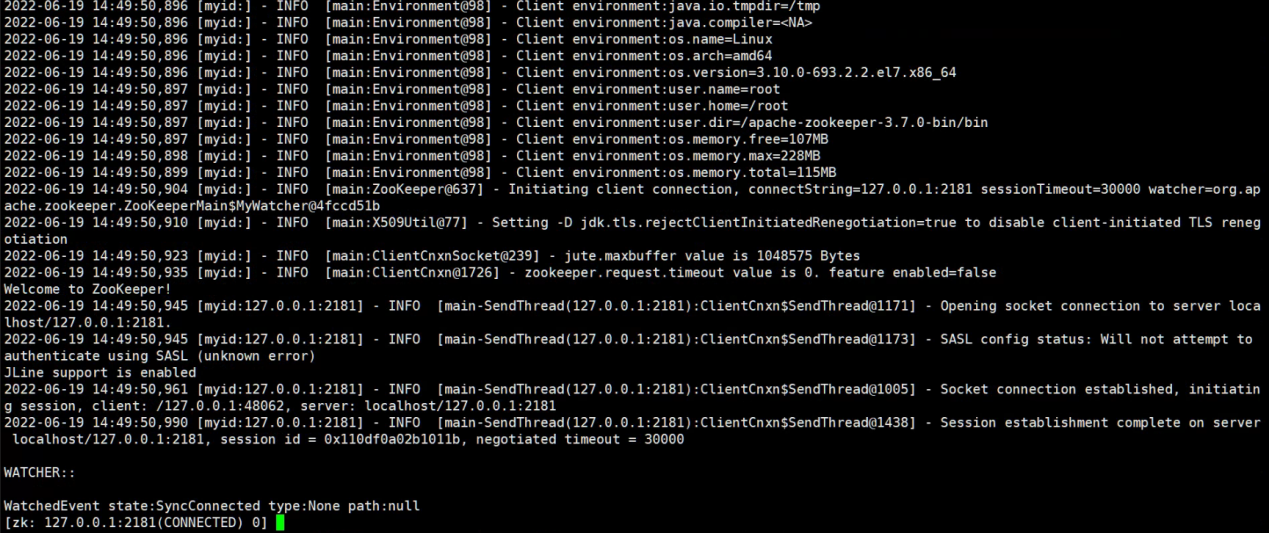
./zkServer.sh start ../2181/conf/zoo.cfg

7.启动后，命令行连接服务器查看是否成功搭建。执行命令

cd /apache-zookeeper-3.7.0-bin/bin/

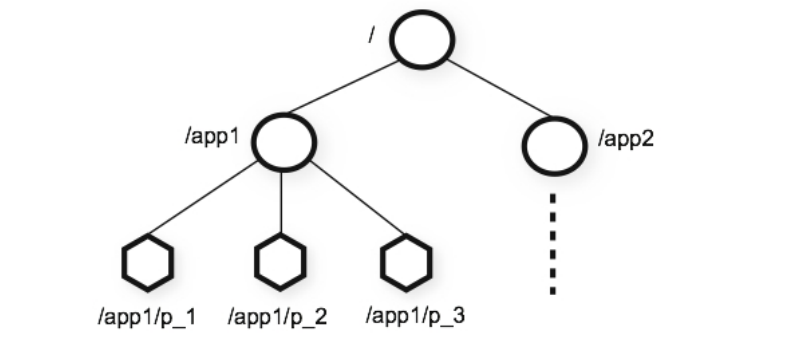
./zkCli.sh -server 127.0.0.1:2181 【连接集群中的一个节点】

如果显示下图内容，表明搭建成功。



Zookeeper的基本使用

在ZooKeeper中，数据信息被保存在⼀个个数据节点上，这些节点被称为znode。ZNode 是 Zookeeper 中最⼩数据单位，在 ZNode 下⾯⼜可以再挂 ZNode，这样⼀层层下去就形成了⼀个层次化 命名空间 ZNode 树，我们称为 ZNode Tree，它采⽤了类似⽂件系统的层级树状结构进⾏管理。⻅下图 ：



上图根⽬录下有两个节点，分别是：app1 和 app2，其中 app1 下⾯⼜有三个⼦节点,所有ZNode按层次化进⾏组织，形成这么⼀颗树，ZNode的节 点路径标识⽅式和Unix⽂件系统路径⾮常相似，都是由⼀系列使⽤斜杠（/）进⾏分割的路径表示，可以向这个节点写⼊数据，也可以在这个节点下⾯创建⼦节点。

Zookeeper 节点类型可以分为三⼤类： 持久性节点（Persistent） 临时性节点（Ephemeral） 顺序性节点（Sequential） 在开发中在创建节点的时候通过组合可以⽣成以下四种节点类型：持久节点、持久顺序节点、临时节 点、临时顺序节点。不同类型的节点则会有不同的⽣命周期 持久节点：是Zookeeper中最常⻅的⼀种节点类型，所谓持久节点，就是指节点被创建后会⼀直存在服 务器，直到删除操作主动清除 持久顺序节点：就是有顺序的持久节点，节点特性和持久节点是⼀样的，只是额外特性表现在顺序上。 顺序特性实质是在创建节点的时候，会在节点名后⾯加上⼀个数字后缀，来表示其顺序。 临时节点：就是会被⾃动清理掉的节点，它的⽣命周期和客户端会话绑在⼀起，客户端会话结束，节点 会被删除掉。与持久性节点不同的是，临时节点不能创建⼦节点。 临时顺序节点：就是有顺序的临时节点，和持久顺序节点相同，在其创建的时候会在名字后⾯加上数字 后缀。

数据变更通知 ，ZooKeeper 允许客户端向服务端注册⼀个 Watcher 监听，当服务端的节点被删除或者被修改时，zookeeper服务端会主动告知客户端数据节点发生了变化，客户端接收到通知后，会重新去服务端获取所做的修改 。

## ZooKeeper命令⾏操作

创建节点：

create [-s] [-e] path data [acl]

-s：创建顺序节点，不加此参数则默认创建非顺序节点

-e：创建临时节点，不加此参数则默认创建持久节点

alc：权限相关修饰符

删除节点：

delete path # 删除指定节点，当节点下包含子节点则无法删除

deleteall path# 递归删除节点

查看节点：

get path

修改节点数据：

set path data

## Zookeeper在项目中的使用

在阿里云服务器中，zookeeper部署在服务中心（47.93.26.215）服务器上。他存储着两部分内容。第一部分是各个子服务器的项目配置参数。第二是各个项目暴露的远程调用服务，这些服务的信息都存储在临时节点上。

使用ls 命令可以清晰的知道zookeeper上存储着哪些内容。其中以/alpha ,/beta类似的以 “/“+环境 的命名 ，就表示该节点及其子节点上存储这环境的服务信息。

下图表示 beta环境下暴露了3个服务的信息。



下图表示 部署在/tjxfs.mrworlds.cn/7980上的服务的相关配置。

