

#前提:安装一个好Linux系统

#1.创建文件夹用于放压缩文件和解压后的文件

```
mkdir -p /export/sofwear #用于放压缩文件
```

```
mkdir -p /export/server #用于放解压后的文件
```

#2.上传文件(win上传) 一个zookeeper 一个jdk ,由于是解压好的文件直接上传到server

```
scp c:/users/liuhua/desktop/zoopeeper3.4.6 root@服务器ip:/export/server -C
```

```
scp c:/users/liuhua/desktop/jdk1.8.0_241 root@服务器ip:/export/server -C
```

#3.修改主机名 和 hosts映射

#修改主机名

```
vim /etc/hostname
```

#修改hosts映射,在最后一行增加

```
vim /etc/hosts #追加内容
```

```
42.194.186.185 node1.itcast.cn node1
```

```
120.77.161.88 node2.itcast.cn node2
```

```
....
```

#4.关闭防火墙 (contos7系统命令)

#关闭firewalld

```
systemctl status firewalld #查看防火墙状态
```

```
systemctl stop firewalld #关闭防火墙
```

```
systemctl disable firewalld #关闭防火墙开机自启
```

#关闭selinux 重启生效

```
vim/etc/selinux/config #打开文件
```

```
SELINUX=disabled #修改此行
```

#时间同步 自行同步

#5.ssh免密登录

#5.1有多少台Linux都执行都生成公钥私钥

```
ssh -keygen [-t rsa]
```

#5.2把各台linux系统的id复制给主机(包括主机也要执行一遍)Linux集合一下 设主机一台

```
ssh-copy-id 主机ip #各台
```

#5.3然后进入主机把authorized_keys复制给其他Linux \$PWD:和主机同目录如果目录不同则需要指定

```
scp /root/.ssh/authorized_keys root@从机ip:$PWD
```

#6.登录主机

#6.1配置jdk环境

```
vim /etc/profile
```

#6.1.1输入以下内容:

```
#set java environment
```

```
export JAVA_HOME=/export/server/jdk1.8.0_241
```

```
export PATH=:$PATH:$JAVA_HOME/bin
```

#6.1.2使环境变量生效

```
source /etc/profile
```

#确认jdk生效 如果失败则可以尝试修改jdk文件夹的权限

```
Java -version
```

#6.2配置zookeeper

#6.2.1修改文件名,防止以后更新版本名字不同而出错

```
cd /export/server
mv zookeeper-3.4.6/ zookeeper
```

#6.2.2修改配置文件

#zk默认加载文件为zoo.cfg 所以需要修改文件名

```
cd zookeeper/conf
mv zoo_sample.cfg zoo.cfg
```

#修改zoo.cfg

```
vim zoo.cfg
```

#修改

```
dataDir=/export/data/zkdata
```

#文件最后添加 2888心跳端口 3888选举端口,如果是本机地址则填0.0.0.0

```
server.1=0.0.0.0:2888:3888
```

```
server.2=ip地址:2888:3888
```

```
server.3=ip地址:2888:3888
```

#6.2.3每台机器上创建文件,名字叫做myid,myid里面的数字就是该台机器上server编号.server.N ;N的数字就是编号

```
mkdir -p /export/data/zkdata
```

```
echo 1 >/export/data/zkdata/myid # 1 为编号多台Linux是不同的
```

#第二台:

```
mkdir -p /export/data/zkdata
```

```
echo 2 >/export/data/zkdata/myid
```

....

#6.2.4把安装包同步在其他Linux上 jdk同理 node2,可以为从机ip

```
cd /export/server
```

```
scp -r zookeeper/ node2:$PWD
```

```
scp -r zookeeper/ node3:$PWD
```

#在每台机器上单独启动服务 默认启动日志叫zookeeper.out

```
/export/server/zookeeper/bin/zkServer.sh start|stop|status #.sh空格加状态
```

#全部机器启动之后查看是否正常运行

```
jps
```

#如果有jps 和 QuorumPeerMain 就成功了

#到此已经配置成功了

#进入zookeeper

```
/export/server/zookeeper/bin/zkCli.sh #进入zookeeper脚本
```

#zookeeper常规操作(只有文件夹的概念没有文件的概念) 文件叫做节点

help #可以查看操作命令

get 路径 #查看文件内容,里面包括属性

ls 路径 #查看路径下的文件

ls2 路径 #可以查看文件的属性

set 路径 内容 #修改内容,需要全部重写

delete 路径 #删除文件(单个)

rmr 路径 #递归删除 文件路径

create 路径 #增加文件,只能一个一个增加,不能直接增加两个路径以上

#zookeeper监听

#监听实现监听需要3步

1. 设置监听

2. 执行监听

3. 事件发生, 触发监听, 通知给设置监听的, 回调callback

#监听是什么

1. 客户端监听zk服务 #谁监听谁

2. 监听zk上目录树znode的变化情况.znode增加了, 删除了, 增加子节点了, 不见了

#实现监听

#1. 设置监听 然后zk服务执行监听 [watch]参数就是设置监听如果没写参数则不设置监听

#可以监听修改文件, 内容, 删除, 增加, 不见了, 需要ls等命令设置监听

ls path [watch] #ls可以为上面常规操作都可以这样执行

#2. 触发监听

set 路径 2222 #修改了被监听的节点数据, 触发监听

#3. 回调通知客户端

WATCHER::

WatchedEvent state:SyncConnected type:NodeDataChanged path:/itheima

#type:发生了什么 path:哪里发生的(路径)

#临时节点 一次连接则是一个会话, 断开连接临时节点则消失

#在 常规操作添加 -e 则就是一个临时节点

create -e 路径

一键启动脚本

```
[root@node1 ~]# vim startZk.sh
```

```
#!/bin/bash
```

```
hosts=(node1 node2 node3)
```

```
for host in ${hosts[*]}
```

```
do
```

```
ssh $host "source /etc/profile;/export/server/zookeeper/bin/zkServer.sh start"
```

```
done
```

一键关闭脚本

```
[root@node1 ~]# vim stopZk.sh
```

```
#!/bin/bash
```

```
hosts=(node1 node2 node3)
```

```
for host in ${hosts[*]}
```

```
do
```

```
ssh $host "/export/server/zookeeper/bin/zkServer.sh stop"
```

```
done
```

- 注意: 关闭java进程时候 根据进程号 直接杀死即可就可以关闭。启动java进程的时候 需要JDK。

- shell程序ssh登录的时候不会自动加载/etc/profile 需要shell程序中自己加载。

