

## Big Data / Zookeeper Assignment

Set up a zookeeper cluster with 3 nodes.

1. Create 3 Virtual Machines.

Run:

Sudo apt update

Sudo apt upgrade

2. For zookeeper , we need to have java installed so install java on your virtual machine.

Sudo apt install default-jre

Sudo apt install default-jdk

3. Check the version of the java installed via java --version for jre and javac --version for jdk

4. Install and configure the zookeeper now on the machine.

sudo wget

<https://dlcdn.apache.org/zookeeper/zookeeper-3.6.3/apache-zookeeper-3.6.3-bin.tar.gz>

Extract the tar file.

Sudo tar -xvf apache-zookeeper-3.6.3-bin.tar.gz -C /opt/

The screenshot shows a terminal window on a Linux system named 'zookeeper3'. The user runs the command 'sudo wget https://dlcdn.apache.org/zookeeper/zookeeper-3.6.3/apache-zookeeper-3.6.3-bin.tar.gz' to download the tarball. Once downloaded, the user runs 'sudo tar -xvf apache-zookeeper-3.6.3-bin.tar.gz -C /opt/' to extract the contents into the '/opt/' directory. The terminal output shows the progress of the download and extraction, including file sizes and speeds.

```
zookeeper3@zookeeper3-VirtualBox:~$ sudo wget https://dlcdn.apache.org/zookeeper/zookeeper-3.6.3/apache-zookeeper-3.6.3-bin.tar.gz
--2022-03-06 16:46:28-- https://dlcdn.apache.org/zookeeper/zookeeper-3.6.3/apache-zookeeper-3.6.3-bin.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 12516362 (12M) [application/x-gzip]
Saving to: 'apache-zookeeper-3.6.3-bin.tar.gz'

apache-zookeeper-3. 100%[=====] 11.94M 7.00MB/s in 1.7s
2022-03-06 16:46:30 (7.00 MB/s) - 'apache-zookeeper-3.6.3-bin.tar.gz' saved [12516362/12516362]

zookeeper3@zookeeper3-VirtualBox:~$ sudo tar -xvf apache-zookeeper-3.6.3-bin.tar.gz -C /opt/
apache-zookeeper-3.6.3-bin/docs/
apache-zookeeper-3.6.3-bin/docs/skin/
apache-zookeeper-3.6.3-bin/docs/images/
```

Change the name of the folder to zookeeper for the convenience purpose only.

Create a zoo.conf file in the conf directory of the /opt/zookeeper and copy the content of zoo\_sample.conf to that file.

```
tu VirtualBox:~$ cd /opt
...VirtualBox:/opt$ sudo mv apache-zookeeper-3.6.3-bin zookeeper
VirtualBox:/opt/zookeeper$ cd zookeeper
VirtualBox:/opt/zookeeper$ cd conf
VirtualBox:/opt/zookeeper/conf$ sudo mv zoo_sample.cfg zoo.cfg
VirtualBox:/opt/zookeeper/conf$ ls
sl log4j.properties zoo.cfg
```

Add this information in the zoo.cfg file.

```
GNU nano 4.8          /opt/zookeeper/conf/zoo.cfg
# The number of milliseconds of each tick
tickTime=2000
# The number of ticks that the initial
# synchronization phase can take
initLimit=5
# The number of ticks that can pass between
# sending a request and getting an acknowledgement
syncLimit=2
# the directory where the snapshot is stored.
# do not use /tmp for storage, /tmp here is just
# example sakes.
dataDir=/var/lib/zookeeper
dataLogDir=/var/log/zookeeper/logs
# the port at which the clients will connect
clientPort=2181
server.1=10.0.2.2:2888:3888
server.2=10.0.2.6:2888:3888
server.3=10.0.2.10:2888:3888

# the maximum number of client connections.
# increase this if you need to handle more clients
# myid=1
```

Now create a directory for your apache zookeeper in the /var/lib directory where you can create and assign the myid to each respective server on the virtual machine.

```
Sudo mkdir /var/lib/zookeeper
Sudo touch /var/lib/zookeeper/myid
Sudo sh -c "echo '3' > /var/lib/zookeeper/myid"
```

Repeat the above steps on all the three machine.

Once all the machine are set up.

Restart the zookeeper and test if the configuration is done properly .

Test & Run the Zookeeper.

Start the zookeeper with the help of zkServer.sh as shown below:

```
zookeeper2@zookeeper2-VirtualBox:~$ cd /opt/zookeeper
zookeeper2@zookeeper2-VirtualBox:/opt/zookeeper$ sudo bin/zkServer.sh start
/usr/bin/java
ZooKeeper JMX enabled by default
Using config: /opt/zookeeper/bin/../conf/zoo.cfg
Starting zookeeper ... STARTED
```

We can check if the zookeeper is working properly with the help of zkCli.sh As shown below the zookeeper cluster is working fine. It is able to connect to all the nodes of the zookeeper cluster.

1.

```
zookeeper1@zookeeper1-VirtualBox:/opt/zookeeper$ sudo bin/zkCli.sh -server 10.0
.2.2:2181
/usr/bin/java
Connecting to 10.0.2.2:2181
2022-03-06 19:29:24,912 [myid:] - INFO  [main:Environment@98] - Client environm
ent:zookeeper.version=3.6.3--6401e4ad2087061bc6b9f80dec2d69f2e3c8660a, built on
 04/08/2021 16:35 GMT
2022-03-06 19:29:24,916 [myid:] - INFO  [main:Environment@98] - Client environm
ent:host.name=zookeeper1-VirtualBox
2022-03-06 19:29:24,916 [myid:] - INFO  [main:Environment@98] - Client environm
ent:java.version=11.0.13
2022-03-06 19:29:24,917 [myid:] - INFO  [main:Environment@98] - Client environm
ent:java.vendor=Ubuntu
2022-03-06 19:29:24,917 [myid:] - INFO  [main:Environment@98] - Client environm
ent:java.home=/usr/lib/jvm/java-11-openjdk-amd64
2022-03-06 19:29:24,917 [myid:] - INFO  [main:Environment@98] - Client environm
ent:java.class.path=/opt/zookeeper/bin/../zookeeper-server/target/classes:/opt/
zookeeper/bin/../build/classes:/opt/zookeeper/bin/../zookeeper-server/target/li
b/*.jar:/opt/zookeeper/bin/../build/lib/*.jar:/opt/zookeeper/bin/../lib/zooke
per-prometheus-metrics-3.6.3.jar:/opt/zookeeper/bin/../lib/zookeeper-jute-3.6.3.
jar:/opt/zookeeper/bin/../lib/zookeeper-3.6.3.jar:/opt/zookeeper/bin/../lib/sna
ppy-java-1.1.7.jar:/opt/zookeeper/bin/../lib/slf4j-log4j12-1.7.25.jar:/opt/zoo
keeper/bin/../lib/slf4j-api-1.7.25.jar:/opt/zookeeper/bin/../lib/simpleclient_se
rvlet-0.6.0.jar:/opt/zookeeper/bin/../lib/simpleclient_hotspot-0.6.0.jar:/opt/z
ookeeper/bin/../lib/simpleclient_common-0.6.0.jar:/opt/zookeeper/bin/../lib/sim
pleclient-0.6.0.jar:/opt/zookeeper/bin/../lib/netty-transport-native-unix-commo
```

```
Welcome to ZooKeeper!
2022-03-06 19:29:25,022 [myid:10.0.2.2:2181] - INFO  [main-SendThread(10.0.2.2:2181):ClientCnxn$SendThread@1181] - Opening socket connection to server 10.0.2.2/10.0.2.2:2181.
2022-03-06 19:29:25,023 [myid:10.0.2.2:2181] - INFO  [main-SendThread(10.0.2.2:2181):ClientCnxn$SendThread@1183] - SASL config status: Will not attempt to authenticate using SASL (unknown error)
JLine support is enabled
2022-03-06 19:29:25,068 [myid:10.0.2.2:2181] - INFO  [main-SendThread(10.0.2.2:2181):ClientCnxn$SendThread@1013] - Socket connection established, initiating session, client: /10.0.2.2:43166, server: 10.0.2.2/10.0.2.2:2181
2022-03-06 19:29:25,138 [myid:10.0.2.2:2181] - INFO  [main-SendThread(10.0.2.2:2181):ClientCnxn$SendThread@1448] - Session establishment complete on server 10.0.2.2/10.0.2.2:2181, session id = 0x10000419e0c0000, negotiated timeout = 3000
0

WATCHER::

WatchedEvent state:SyncConnected type:None path:null
```

2.

```
zookeeper1@zookeeper1-VirtualBox:/opt/zookeeper$ sudo bin/zkCli.sh -server 10.0.2.10:2181
/usr/bin/java
Connecting to 10.0.2.10:2181
2022-03-06 19:36:00,502 [myid:] - INFO  [main:Environment@98] - Client environment:zookeeper.version=3.6.3--6401e4ad2087061bc6b9f80dec2d69f2e3c8660a, built on 04/08/2021 16:35 GMT
2022-03-06 19:36:00,505 [myid:] - INFO  [main:Environment@98] - Client environment:host.name=zookeeper1-VirtualBox
2022-03-06 19:36:00,505 [myid:] - INFO  [main:Environment@98] - Client environment:java.version=11.0.13
2022-03-06 19:36:00,506 [myid:] - INFO  [main:Environment@98] - Client environment:java.vendor=Ubuntu
2022-03-06 19:36:00,506 [myid:] - INFO  [main:Environment@98] - Client environment:java.home=/usr/lib/jvm/java-11-openjdk-amd64
2022-03-06 19:36:00,506 [myid:] - INFO  [main:Environment@98] - Client environment:java.class.path=/opt/zookeeper/bin/../zookeeper-server/target/classes:/opt/zookeeper/bin/../build/classes:/opt/zookeeper/bin/../zookeeper-server/target/lib/*.jar:/opt/zookeeper/bin/../build/lib/*.jar:/opt/zookeeper/bin/../lib/zookeeper-prometheus-metrics-3.6.3.jar:/opt/zookeeper/bin/../lib/zookeeper-jute-3.6.3.jar:/opt/zookeeper/bin/../lib/zookeeper-3.6.3.jar:/opt/zookeeper/bin/../lib/snappy-java-1.1.7.2.jar:/opt/zookeeper/bin/../lib/slf4j-log4j12-1.7.25.jar:/opt/zookeeper/bin/../lib/slf4j-api-1.7.25.jar:/opt/zookeeper/bin/../lib/simpleclient-se
```

```

        value is 1048575 bytes
2022-03-06 19:36:00,561 [myid:] - INFO  [main:ClientCnxn@1736] - zookeeper.request.timeout value is 0. feature enabled=false
Welcome to ZooKeeper!
2022-03-06 19:36:00,615 [myid:10.0.2.10:2181] - INFO  [main-SendThread(10.0.2.10:2181):ClientCnxn$SendThread@1181] - Opening socket connection to server 10.0.2.10/10.0.2.10:2181.
2022-03-06 19:36:00,615 [myid:10.0.2.10:2181] - INFO  [main-SendThread(10.0.2.10:2181):ClientCnxn$SendThread@1183] - SASL config status: Will not attempt to authenticate using SASL (unknown error)
JLine support is enabled
2022-03-06 19:36:00,645 [myid:10.0.2.10:2181] - INFO  [main-SendThread(10.0.2.10:2181):ClientCnxn$SendThread@1013] - Socket connection established, initiating session, client: /10.0.2.2:37168, server: 10.0.2.10/10.0.2.10:2181
2022-03-06 19:36:00,677 [myid:10.0.2.10:2181] - INFO  [main-SendThread(10.0.2.10:2181):ClientCnxn$SendThread@1448] - Session establishment complete on server 10.0.2.10/10.0.2.10:2181, session id = 0x3000038cedd0000, negotiated timeout = 30000

WATCHER::

WatchedEvent state:SyncConnected type:None path:null
[zk: 10.0.2.10:2181(CONNECTED) 0]

```

3.

```

zookeeper1@zookeeper1-VirtualBox:/opt/zookeeper$ sudo bin/zkcli.sh -server 10.0.2.6:2181
/usr/bin/java
Connecting to 10.0.2.6:2181
2022-03-06 19:30:26,455 [myid:] - INFO  [main:Environment@98] - Client environment:zookeeper.version=3.6.3--6401e4ad2087061bc6b9f80dec2d69f2e3c8660a, built on 04/08/2021 16:35 GMT
2022-03-06 19:30:26,459 [myid:] - INFO  [main:Environment@98] - Client environment:host.name=zookeeper1-VirtualBox
2022-03-06 19:30:26,459 [myid:] - INFO  [main:Environment@98] - Client environment:java.version=11.0.13
2022-03-06 19:30:26,461 [myid:] - INFO  [main:Environment@98] - Client environment:java.vendor=Ubuntu
2022-03-06 19:30:26,461 [myid:] - INFO  [main:Environment@98] - Client environment:java.home=/usr/lib/jvm/java-11-openjdk-amd64
2022-03-06 19:30:26,461 [myid:] - INFO  [main:Environment@98] - Client environment:java.class.path=/opt/zookeeper/bin/./zookeeper-server/target/classes:/opt/zookeeper/bin/./build/classes:/opt/zookeeper/bin/./zookeeper-server/target/lib/*.jar:/opt/zookeeper/bin/./build/lib/*.jar:/opt/zookeeper/bin/./lib/zookeeper-prometheus-metrics-3.6.3.jar:/opt/zookeeper/bin/./lib/zookeeper-jute-3.6.3.jar:/opt/zookeeper/bin/./lib/zookeeper-3.6.3.jar:/opt/zookeeper/bin/./lib/snappy-java-1.1.7.jar:/opt/zookeeper/bin/./lib/slf4j-log4j12-1.7.25.jar:/opt/zookeeper/bin/./lib/slf4j-api-1.7.25.jar:/opt/zookeeper/bin/./lib/simpleclient_ser...Terminal .0.jar:/opt/zookeeper/bin/./lib/simpleclient_hotspot-0.6.0.jar:/opt/zookeeper/bin/./lib/simpleclient_common-0.6.0.jar:/opt/zookeeper/bin/./lib/simpleclient-0.6.0.jar:/opt/zookeeper/bin/./lib/netty-transport-native-unix-common-4.1.63.Final.jar:/opt/zookeeper/bin/./lib/netty-transport-native-epoll-4.1.63.Final.jar:/opt/zookeeper/bin/./lib/netty-transport-4.1.63.Final.jar:/opt/zookeeper/bin/./lib/netty-resolver-4.1.63.Final.jar:/opt/zookeeper/bin/./lib/net

```

```
Welcome to ZooKeeper!
2022-03-06 19:30:26,572 [myid:10.0.2.6:2181] - INFO  [main-SendThread(10.0.2.6:2181):ClientCnxn$SendThread@1181] - Opening socket connection to server 10.0.2.6/10.0.2.6:2181.
2022-03-06 19:30:26,575 [myid:10.0.2.6:2181] - INFO  [main-SendThread(10.0.2.6:2181):ClientCnxn$SendThread@1183] - SASL config status: Will not attempt to authenticate using SASL (unknown error)
JLine support is enabled
2022-03-06 19:30:26,606 [myid:10.0.2.6:2181] - INFO  [main-SendThread(10.0.2.6:2181):ClientCnxn$SendThread@1013] - Socket connection established, initiating session, client: /10.0.2.2:60406, server: 10.0.2.6/10.0.2.6:2181
2022-03-06 19:30:26,628 [myid:10.0.2.6:2181] - INFO  [main-SendThread(10.0.2.6:2181):ClientCnxn$SendThread@1448] - Session establishment complete on server 10.0.2.6/10.0.2.6:2181, session id = 0x200004110b90000, negotiated timeout = 30000
WATCHER::

WatchedEvent state:SyncConnected type:None path:null
[zk: 10.0.2.6:2181(CONNECTED) 0]
```

## How to Identify which zookeeper is the leader?

```
echo srvr | nc localhost 2181 | grep Mode
```

This can be used if the zookeeper node is a leader or a follower.

```
zookeeper2@zookeeper2-VirtualBox:~$ echo srvr | nc localhost 2181 | grep Mode
Mode: leader
```

```
zookeeper1@zookeeper1-VirtualBox:~$ echo srvr | nc localhost 2181 | grep Mode
Mode: follower
```

```
zookeeper3@zookeeper3-VirtualBox:~$ echo srvr | nc localhost 2181 | grep Mode
Mode: follower
```

## How to identify the connections on a zookeeper node?

This command can be used to identify the connections that are currently present on the zookeeper node:

```
echo cons | nc localhost 2181
```

```
zookeeper1@zookeeper1-VirtualBox:/opt/zookeeper$ sudo bin/zkServer.sh stop
/usr/bin/java
ZooKeeper JMX enabled by default
Using config: /opt/zookeeper/bin/../conf/zoo.cfg
Stopping zookeeper ... STOPPED
zookeeper1@zookeeper1-VirtualBox:/opt/zookeeper$ sudo bin/zkServer.sh start
/usr/bin/java
ZooKeeper JMX enabled by default
Using config: /opt/zookeeper/bin/../conf/zoo.cfg
Starting zookeeper ... STARTED
zookeeper1@zookeeper1-VirtualBox:/opt/zookeeper$ echo cons | nc localhost 2181
/127.0.0.1:49208[0](queued=0,recved=1,sent=0)

zookeeper1@zookeeper1-VirtualBox:/opt/zookeeper$ echo cons | nc localhost 2181
/10.0.2.6:47800[1](queued=0,recved=1,sent=1,sid=0x1000083ccd20000,lop=SESS,est
=1646579513440,to=30000,lcxid=0x0,lzxid=0x600000001,lresp=8663403,llat=20,minla
t=0,avglat=20,maxlat=20)
/127.0.0.1:49208[0](queued=0,recved=1,sent=0)
Show Applications
```

## What will happen if the leader goes down?

A new zookeeper leader is elected if the failed node is the current user.  
Here zookeeper node-2 is the leader.

```
zookeeper2@zookeeper2-VirtualBox:~$ echo srvr | nc localhost 2181 | grep Mode
Mode: leader
```

Once that zookeeper node-2 failed , it immediately elects the zookeeper node-3 as the leader as shown in the picture.

```
zookeeper3@zookeeper3-VirtualBox:~$ echo srvr | nc localhost 2181 | grep Mode
Mode: leader
```

## What will happen if you purge snapshot?

Zookeeper contains files which are copy of the znodes stored by a particular serving ensemble as snapshots and transactional log files. Any changes that are made to a znodes get appended to a transaction log, when a large data grows , a snapshot of the current state of all znodes will be written to the filesystem .

There is no automatic cleanup on these snapshot or transactional logs, so in order to purge this is the responsibility of the operator or we can invoke the auto purge for the snapshots and the transactional logs. It will retain a certain snap count and delete the rest. We can do this on the basis of an interval and the count as well.

#### **autopurge.snapRetainCount**

When enabled, ZooKeeper auto purge feature retains the autopurge.snapRetainCount most recent snapshots and the corresponding transaction logs in the dataDir and dataLogDir respectively and deletes the rest.

- Type: int
- Default: 3
- Importance: high

#### **autopurge.purgeInterval**

The time interval in hours for which the purge task has to be triggered. Set to a positive integer (1 and above) to enable the auto purging.

- Type: int
- Default: 0
- Importance: high

## What will happen if you add 1gb znode?

Zookeeper has a maximum size limit of 1MB. For a 1 Gb znode, Zookeeper will not be suitable for this. Zookeeper keeps dumping or snapshotting the data tree periodically.

The data limit of the znode is limited to 1 MB only. It can be increased but it is strongly discouraged because it causes performance drop and seriously disrupts operation of the whole system or even causes its malfunction. So it will start snapshotting and remove the data tree periodically.