Cloud Lab Assignment-4 | Apache Zookeeper

Details

- SRN : PES2UG20CS237
- Name : P K Navin Shrinivas
- Section : D

Lab Deliverables

Task 1 : Installing Zookeeper

- We first download the zookeeper tar and do the following :

```
$ wget https://dlcdn.apache.org/zookeeper/zookeeper-
3.8.0/apache-zookeeper-3.8.0-bin.tar.gz
$ tar zxf apache-zookeeper-3.8.0-bin.tar.gz
```

- Copy the extracted folder to the `/opt` folder.

```
$ sudo cp -r apache-zookeeper-3.8.0-bin /opt
$ cd /opt/apache-zookeeper-3.8.0-bin
```

- Create a new conf file in called zoo.cfg in te `conf` folder. With the following content :

```
tickTime = 2000
dataDir = /opt/apache-zookeeper-3.8.0-bin/data
clientPort = 2181
initLimit = 5
syncLimit = 2
```

- We now start the server [1a] :

```
$ sudo ./bin/zkServer.sh start
```

> Note : You can use start-foreground to see error/detailed logs.

```
2023-03-23 22:00:32,518 [myid:] - INFO  [main:o.a.z.s.p.FileTxnSnapLog@479] - Snapshotting: 0x0 to /opt/apache-zookeeper-3.8.0-bin/data/version-2/snapshot.0
2023-03-23 22:00:32,520 [myid:] - INFO  [main:o.a.z.s.ZKDatabase@289] - Snapshot loaded in 7 ms, highest zxid is 0x0, digest is 1371985504
2023-03-23 22:00:32,521 [myid:] - INFO  [main:o.a.z.s.p.FileTxnSnapLog@479] - Snapshotting: 0x0 to /opt/apache-zookeeper-3.8.0-bin/data/version-2/snapshot.0
2023-03-23 22:00:32,522 [myid:] - INFO  [main:o.a.z.s.ZooKeeperServer@547] - Snapshot taken in 1 ms
2023-03-23 22:00:32,529 [myid:] - INFO  [ProcessThread(sid:0 cport:2181)::o.a.z.s.PrepRequestProcessor@138] - PrepRequestProcessor (sid:0) started, reconfigEnabled=false
2023-03-23 22:00:32,529 [myid:] - INFO  [main:o.a.z.s.RequestThrottler@75] - zookeeper.request_throttler.shutdownTimeout = 10000 ms
2023-03-23 22:00:32,543 [myid:] - INFO  [main:o.a.z.s.ContainerManager@84] - Using checkIntervalMs=60000 maxPerMinute=10000 maxNeverUsedIntervalMs=0
2023-03-23 22:00:32,544 [myid:] - INFO  [main:o.a.z.a.ZKAuditProvider@42] - ZooKeeper audit is disabled.
^C^[[A
→ bin sudo ./zkServer.sh start
/usr/bin/java
ZooKeeper JMX enabled by default
Using config: /opt/apache-zookeeper-3.8.0-bin/bin/../conf/zoo.cfg
Starting zookeeper ... STARTED
→ bin
```

- you can connect to the `CLI` and `stop` the serviec using the scripts in the `/bin` folder.

Task 2 : Getting familiar with zookeeper CLI

- Connect to the cli using the zkCLI.sh shell script.
- We can not create Znode, these are Zookeeper server that serve the clients (servers for the application). These Znodes can be specific to a session or permanenetly part of the Zookeeper service.
- Session based temporary ones are called ephemeral nodes.
- We now create few nodes the observe this behavoiur :

```
create -s /PES2UG20CS237_seq "a sequential node"
create -e /PES2UG20CS237_eph "an ephemeral node"
```

- we can now see nodes `ls /` [2a]

```
ACl is not valid : null
[zk: localhost:2181(CONNECTED) 24] create -s /pes2ug20cs237seq a-sequential-node
Created /pes2ug20cs237seq0000000001
[zk: localhost:2181(CONNECTED) 25] create -e /PES2UG20CS237_eph an-ephemeral-node
Created /PES2UG20CS237_eph
[zk: localhost:2181(CONNECTED) 26] ls /
[PES2UG20CS237_eph, pes2ug20cs237seq0000000001, zookeeper]
[zk: localhost:2181(CONNECTED) 27]
```

- If we quit zookeeper and return back after a while (enough for zookeeper to miss a ping from the client then se eph node will close). [2b] :

```
WatchedEvent state:SyncConnected type:None path:null
[zk: localhost:2181(CONNECTED) 0] ls /
[pes2ug20cs237seq0000000001, zookeeper]
[zk: localhost:2181(CONNECTED) 1] _
```

- we can also change the data and get the data set in the znodes :

```
WatchedEvent state:SyncConnected type:None path:null
[zk: localhost:2181(CONNECTED) 0] ls /
[pes2ug20cs237seq0000000001, zookeeper]
[zk: localhost:2181(CONNECTED) 1] get -s /pes2ug20cs2370000000001
Node does not exist: /pes2ug20cs2370000000001
[zk: localhost:2181(CONNECTED) 2] get -s /pes2ug20cs237seq0000000001
a-sequential-node
cZxid = 0x5
ctime = Thu Mar 23 22:20:20 IST 2023
mZxid = 0x5
mtime = Thu Mar 23 22:20:20 IST 2023
pZxid = 0x5
cversion = 0
dataVersion = 0
aclVersion = 0
ephemeralOwner = 0x0
dataLength = 17
numChildren = 0
[zk: localhost:2181(CONNECTED) 3] _
```

- Znode can have children, only those of permanent can have children [2d].

```
[zk: localhost:2181(CONNECTED) 3] create /pes2ug20cs237seq0000000001/navin
Created /pes2ug20cs237seq0000000001/navin
[zk: localhost:2181(CONNECTED) 4] create /pes2ug20cs237seq0000000001/shrinivas
Created /pes2ug20cs237seq0000000001/shrinivas
[zk: localhost:2181(CONNECTED) 5] ls /pes2ug20cs237seq0000000001
[navin, shrinivas]
[zk: localhost:2181(CONNECTED) 6] _
```

- You can also delete nodes using `delete` or `rmr`.

Task 3 : leader election using Zookeeper.

- By default, the sequential node (permanenet node) is the leader of the ensemble (ensemble is a buch of zookeeper nodes).
- Each "greater sequence" znode keeps a `watch` on the previous node. Hence the node is informed when the the previous smaller node fails. This is nowhere near foolproof.
- As for the lab requirement, you can open up a tmux session to notice many nodes at the same time.
- Follow the document to create a root persistent node and a bunch of `eph` nodes for each client [3a] :

```
2023-03-23 22:32:13,055 [myid:] - INFO  [main:o.a.z.s.w.WatchManagerFactory@42] - Using org.apache.zookeeper.server.watch.WatchManager as watch manager
2023-03-23 22:32:13,055 [myid:] - INFO  [main:o.a.z.s.w.WatchManagerFactory@42] - Using org.apache.zookeeper.server.watch.WatchManager as watch manager
2023-03-23 22:32:13,056 [myid:] - INFO  [main:o.a.z.s.ZKDatabase@132] - zookeeper.snapshotSizeFactor = 0.33
2023-03-23 22:32:13,056 [myid:] - INFO  [main:o.a.z.s.ZKDatabase@152] - zookeeper.commitLogCount=500
2023-03-23 22:32:13,058 [myid:] - INFO  [main:o.a.z.s.p.SnapStream@61] - zookeeper.snapshot.compression.method = CHECKED
2023-03-23 22:32:13,059 [myid:] - INFO  [main:o.a.z.s.p.FileSnap@85] - Reading snapshot /opt/apache-zookeeper-3.8.0-bin/data/version-2/snapshot.0
2023-03-23 22:32:13,061 [myid:] - INFO  [main:o.a.z.s.DataTree@1712] - The digest value is empty in snapshot
2023-03-23 22:32:13,070 [myid:] - INFO  [main:o.a.z.a.ZKAuditProvider@42] - ZooKeeper audit is disabled.
2023-03-23 22:32:13,070 [myid:] - INFO  [main:o.a.z.s.p.FileTxnSnapLog@372] - 11 txns loaded in 6 ms
2023-03-23 22:32:13,070 [myid:] - INFO  [main:o.a.z.s.ZKDatabase@289] - Snapshot loaded in 14 ms, highest zxid is 0xb, digest is 8189940572
2023-03-23 22:32:13,071 [myid:] - INFO  [main:o.a.z.s.p.FileTxnSnapLog@479] - Snapshotting: 0xb to /opt/apache-zookeeper-3.8.0-bin/data/version-2/snapshot.b
2023-03-23 22:32:13,072 [myid:] - INFO  [main:o.a.z.s.ZooKeeperServer@547] - Snapshot taken in 1 ms
2023-03-23 22:32:13,078 [myid:] - INFO  [main:o.a.z.s.RequestThrottler@75] - zookeeper.request_throttler.shutdownTimeout = 10000 ms
2023-03-23 22:32:13,078 [myid:] - INFO  [ProcessThread(sid:0 cport:2181)::o.a.z.s.PrepRequestProcessor@138] - PrepRequestProcessor (sid:0) started, reconfigEnabled=false
2023-03-23 22:32:13,089 [myid:] - INFO  [main:o.a.z.s.ContainerManager@84] - Using checkIntervalMs=60000 maxPerMinute=10000 maxNeverUsedIntervalMs=0
2023-03-23 22:32:22,589 [myid:] - INFO  [SyncThread:0:o.a.z.s.p.FileTxnLog@285] - Creating new log file: log.c
```

```
127.0.0.1:49114, server: localhost/127.0.0.1:2181        2023-03-23 22:32:33,437 [myid:localhost:2181] - INFO  [main-Se  delquota              get
JLine support is enabled                                 ndThread(localhost:2181):o.a.z.ClientCnxn$SendThread@1005] - S  getAcl                getAllChildrenNumber
2023-03-23 22:32:43,914 [myid:localhost:2181] - INFO  [main-S  ocket connection established, initiating session, client: /127  getEphemerals         history
endThread(localhost:2181):o.a.z.ClientCnxn$SendThread@1444] -  .0.0.1:47730, server: localhost/127.0.0.1:2181                 listquota             ls
 Session establishment complete on server localhost/127.0.0.1  JLine support is enabled                                        printwatches          quit
:2181, session id = 0x1000200fc720002, negotiated timeout = 3  2023-03-23 22:32:33,444 [myid:localhost:2181] - INFO  [main-Se  reconfig              redo
0000                                                     ndThread(localhost:2181):o.a.z.ClientCnxn$SendThread@1444] - S  removewatches         set
                                                         ession establishment complete on server localhost/127.0.0.1:21  setAcl                setquota
WATCHER::                                                 81, session id = 0x1000200fc720001, negotiated timeout = 30000  stat                  sync
                                                                                                                         version               whoami
WatchedEvent state:SyncConnected type:None path:null     WATCHER::                                                       [zk: localhost:2181(CONNECTED) 0] create -s -e
[zk: localhost:2181(CONNECTED) 0] create /pes2ug20cs237                                                                  /pesu2ug20cs237/candidate
Created /pes2ug20cs237                                    WatchedEvent state:SyncConnected type:None path:null           Node does not exist: /pesu2ug20cs237/candidate
[zk: localhost:2181(CONNECTED) 1] create -s -e /pes2ug20cs237  [zk: localhost:2181(CONNECTED) 0] create -s  -e /pes2ug20cs237  [zk: localhost:2181(CONNECTED) 1] create -s -e
/candidate                                               /candidate                                                      /pes2ug20cs237/candidate
Created /pes2ug20cs237/candidate0000000000               Created /pes2ug20cs237/candidate0000000001                     Created /pes2ug20cs237/candidate0000000002
[zk: localhost:2181(CONNECTED) 2]                        [zk: localhost:2181(CONNECTED) 1]                              [zk: localhost:2181(CONNECTED) 2]
[5] 0:zsh*                                                                                                                                "zsh" 22:35 23-Mar-23
```

- Now killing of nodes one by one should cause the chain to detect the failures and become the leader [3b] :

- But in a event the old node comes back alone, it does not get the initial eph znode it had, it gets a new eph node under the root znode, this is to ensure there are no 2 leader in the system [3c] :