strangeloop

• 2012 •

Building an Impenetrable ZooKeeper

Kathleen Ting, kathleen@cloudera.com, @kate_ting

cloudera

How to Kill ZooKeeper with 8 Misconfigurations

Who Am I?

Kathleen Ting

- Apache ZooKeeper Subject Matter Expert
- Apache Sqoop Committer, PMC member
- Support Manager, Cloudera

What is ZooKeeper?

- **Coordinator of distributed applications**
- **Small clusters reliably serve many coordination needs**
- **Canary in the Hadoop coal mine**

Why is ZooKeeper Important?

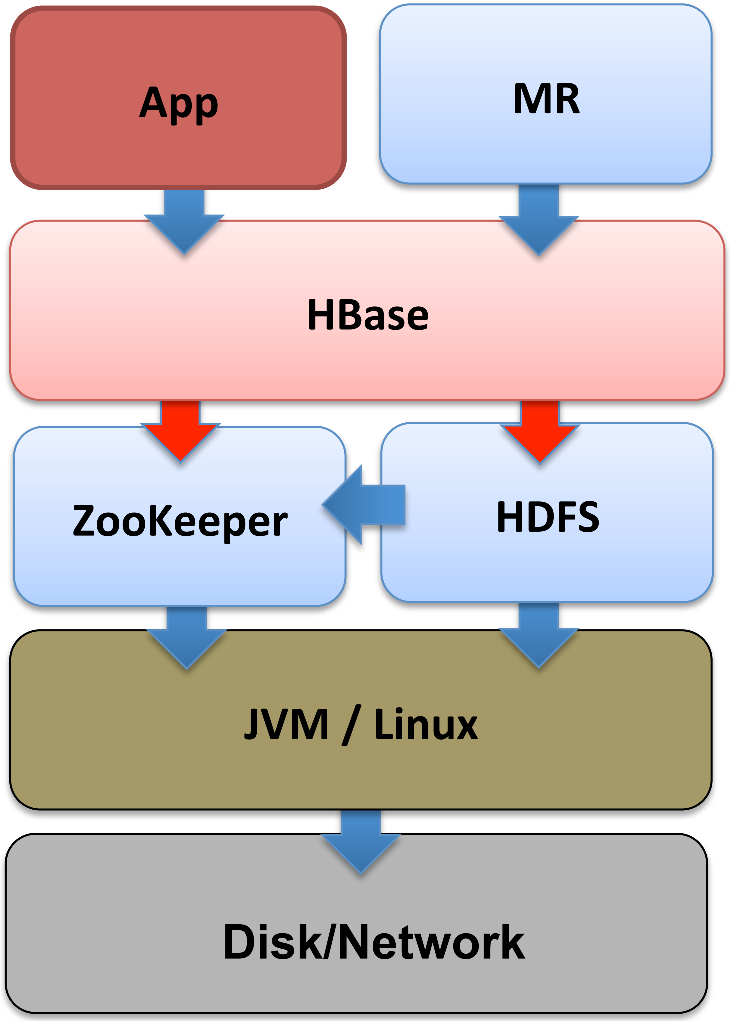
- **High Availability**
  - Replicate to withstand machine failures

- **Distributed Coordination**
  - One consistent framework to rule coordination across all systems
  - Observe every operation by every client in exactly the same order

Who Uses ZooKeeper?

- HBase
- MapReduce (YARN)
- HDFS (High Availability)
- Solr
- Kafka
- S4
- Accumulo
- Numerous custom solutions: https://cwiki.apache.org/confluence/display/ZOOKEEPER/poweredby

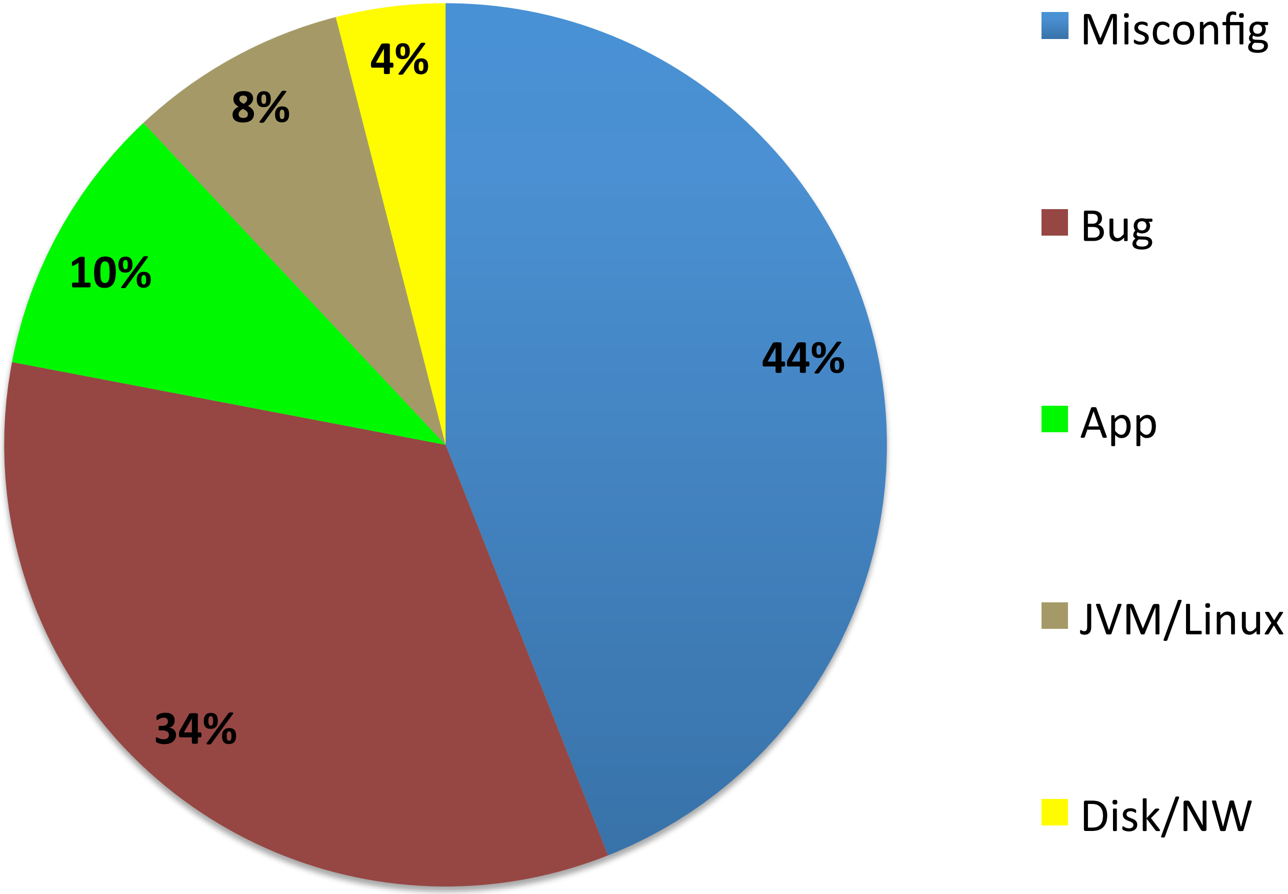
Who Doesn’t Depend on ZooKeeper?



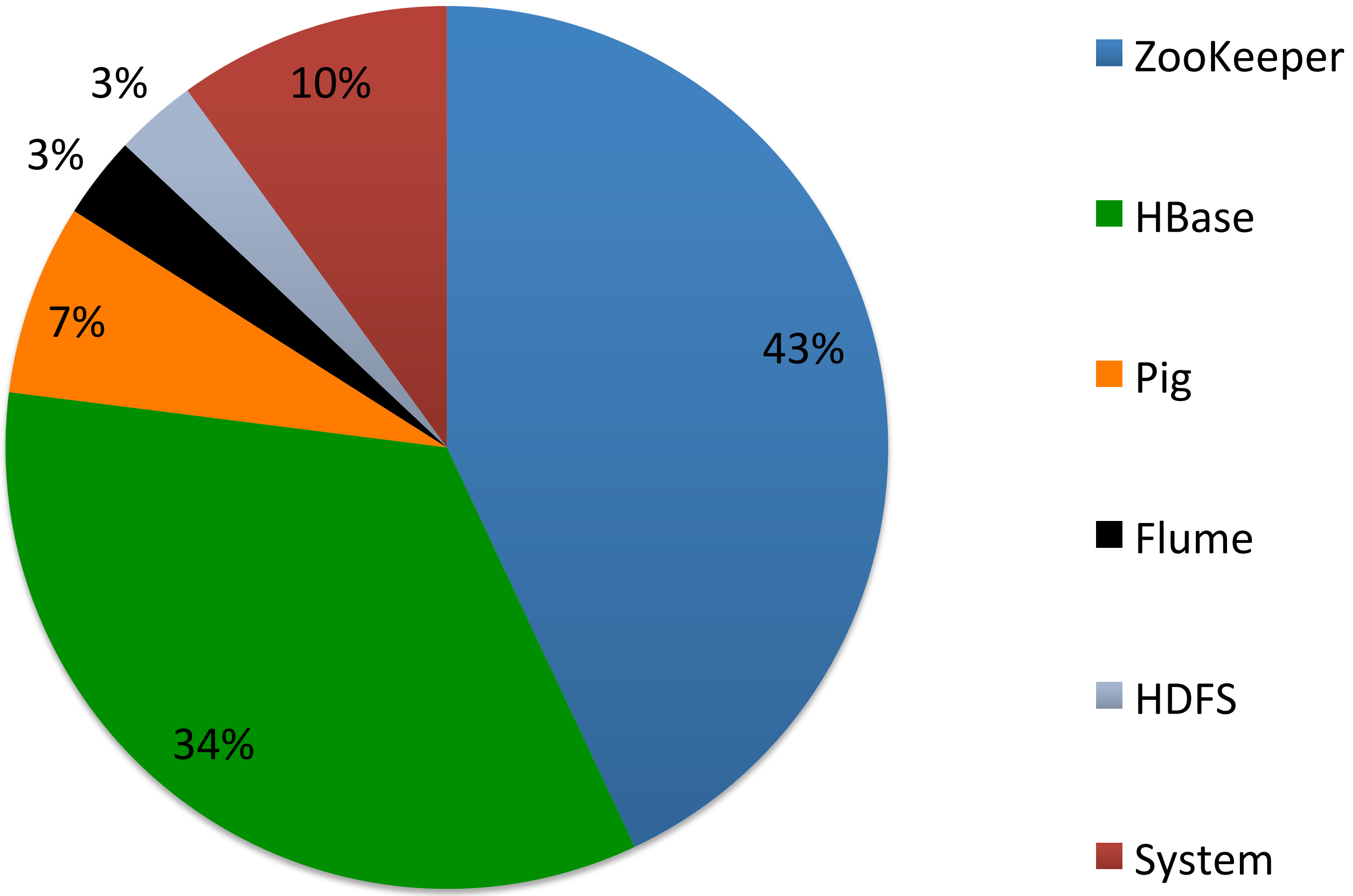
What are Misconfigurations?

- Any diagnostic ticket requiring a change to ZooKeeper (HBase, Hadoop..) or to OS config files
- Comprise 44% of tickets
- e.g. resource-allocation: memory, file-handles, disk-space

Ticket Breakdown by Type



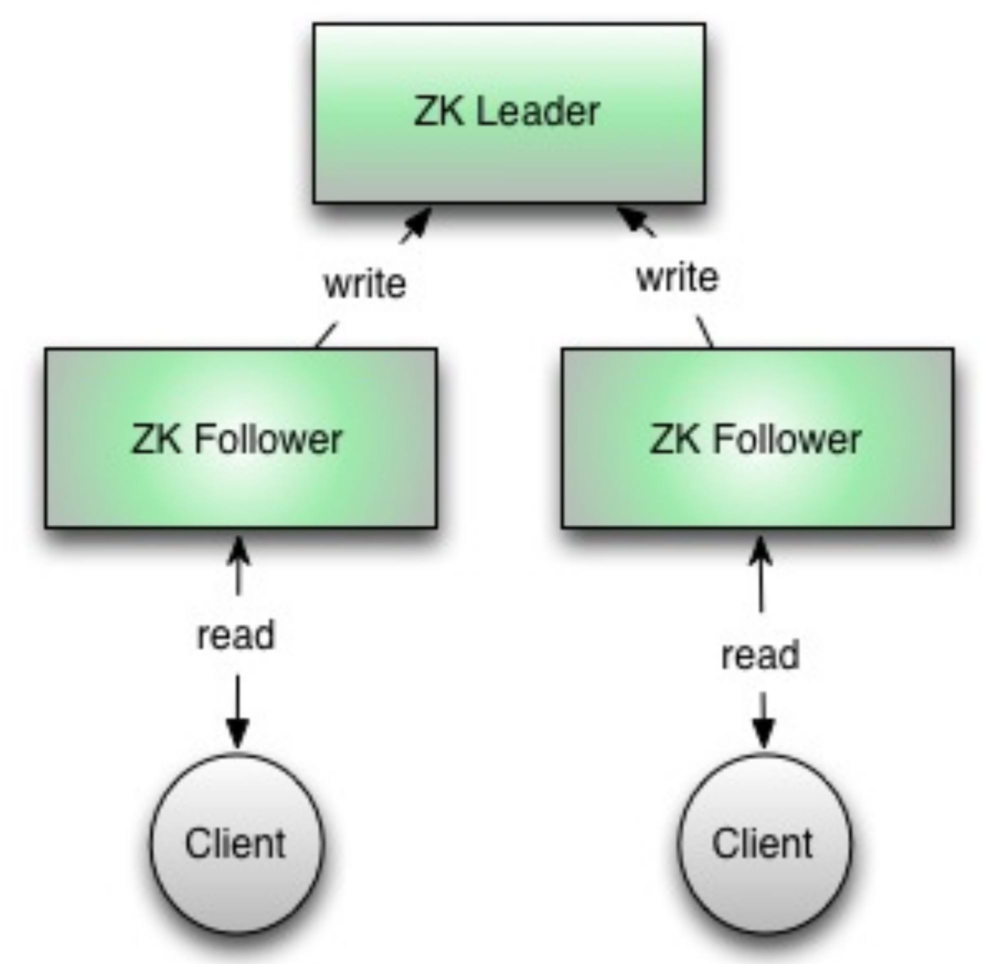
Ticket Breakdown by Component



Analysis of a Year’s ZooKeeper Tickets

- Typically, ZK is straight-forward to set up and operate
- Issues tend to be client issues rather than ZK issues
- Our examples tend to be HBase and Hadoop centric
  - But solutions are applicable to other systems using ZK for coordination

3 ZooKeeper Ensemble



cloudera

Common Issues

- **Connection Mismanagement**
- **Time Mismanagement**
- **Disk Mismanagement**

Common Issues

- Connection Mismanagement
- Time Mismanagement
- Disk Mismanagement

1. Too Many Connections

```
WARN [NIOServerCxn.Factory:
0.0.0.0/0.0.0.0:2181:NIOServerCnxn$Factory@247] - Too many
connections from /xx.x.xx.xxx - max is 60
```

How can it be resolved?

- Running out of ZK connections?
  - Set maxClientCnxns=200 in zoo.cfg
- HBase client leaking connections?
  - Manually close connections
  - Fixed in HBASE-3777, HBASE-4773, and HBASE-5466

2. Connection Closes Prematurely

```
ERROR:
org.apache.hadoop.hbase.ZooKeeperConnectionException:
HBase is able to connect to ZooKeeper but the connection
closes immediately.
```

How can it be resolved?

- If hbase.cluster.distributed = true in hbase-site, then in zoo.cfg, quorum can't be set to localhost
- Bring up an interface with the same IP address from the downed ZK without any service running on port 2181 so the client can fail over to the next ZK server from the quorum
- In hbase-site, set hbase.zookeeper.recoverable.waittime=30000ms
  - Provides enough time for HBase client to try another ZK server
  - Fixed in HBASE-3065

3. Pig Hangs Connecting to HBase

```
WARN org.apache.zookeeper.ClientCnxn: Session 0x0 for
server null, unexpected error, closing socket connection
and attempting reconnect java.net.ConnectionException:
Connection refused
```

What causes this?

- Location of ZK quorum is not known to Pig (default 127.0.0.1:2181 fails)

How can it be resolved?

- Use Pig 10, which includes PIG-2115
- If there is overlap between TaskTrackers and ZK quorum nodes
  - Set hbase.zookeeper.quorum to final in hbase-site.xml
  - Otherwise, add "hbase.zookeeper.quorum=hadoophbasemaster.lan:2181" to "pig.properties" (fixed in PIG-2821)

Common Issues

- Connection Mismanagement

- Time Mismanagement

- Disk Mismanagement

4. Client Session Timed Out

```
INFO org.apache.zookeeper.server.ZooKeeperServer: Expiring
session <id>, timeout of 40000ms exceeded
```

How can it be resolved?

- ZK and HBase need same session timeout values:
  - zoo.cfg: maxSessionTimeout=180000
  - hbase-site.xml: zookeeper.session.timeout=180000
- Don’t co-locate ZK with IO-intense DataNode or RegionServer
- Make sure your session timeout is sufficiently long
- Specify right amount of heap and tune GC flags
  - Turn on Parallel/CMS/Incremental GC

5. Clients Lose Connections

```
WARN  org.apache.zookeeper.ClientCnxn  - Session <id> for
server <name>, unexpected error, closing socket connection
and attempting reconnect
java.io.IOException: Broken pipe
```

Don’t use SSD drive for ZK transaction log

- ZK optimized for mechanical spindles and for sequential IO
- SSD provides little benefit and suffers from high latency spikes
  - http://storagemojo.com/2012/06/07/the-ssd-write-cliff-in-real-life/
  - SSD pre-allocates disk extents to avoid directory updates but that doubles the load on the SSD
  - SSD disk stops for 40 sec (which is greater than session timeout)

Common Issues

- Connection Mismanagement
- Time Mismanagement
- Disk Mismanagement

6. Unable to Load Database – Unable to Run Quorum Server

```
FATAL Unable to load database on disk
java.io.IOException: Failed to process transaction type: 2
error: KeeperErrorCode = NoNode for <file> at
org.apache.zookeeper.server.persistence.FileTxnSnapLog.res
tore(FileTxnSnapLog.java:152)
```

How can it be resolved?

- Archive and wipe /var/zookeeper/version-2 if other two ZK servers are running

7. Unable to Load Database – Unreasonable Length Exception

```
FATAL Unable to load database on disk
java.io.IOException: Unreasonable length = 1048583
at org.apache.jute.BinaryInputArchive.readBuffer
(BinaryInputArchive.java:100)
```

How can it be resolved?

- Server allows a client to set data larger than the server can read from disk
- If a znode is not readable, increase jute.maxbuffer
  - Look for "Packet len <xx> is out of range" in the client log
  - Increase it by 20%
  - Set in JVMFLAGS="-Djute.maxbuffer=yy" bin/zkCli.sh
- Fixed in ZOOKEEPER-1513

8. Failure to Follow Leader

```
WARN org.apache.zookeeper.server.quorum.Learner: Exception
when following the leader java.net.SocketTimeoutException:
Read timed out
```

What causes this?

- Disk IO contention, Network Issues
- ZK snapshot is too large (lots of ZK nodes)

How can it be resolved?

- Reduce IO contention by putting dataDir on dedicated spindle
- Increase initLimit on all ZK servers and restart, see ZOOKEEPER-1521
- Monitor network (e.g. ifconfig)

Optimal Ensemble Size

# of ZK Servers	Purpose
1	Coordination
3	Reliability for production environment
5	Permits taking one server down for maintenance

Why not run 11 ZK servers?

Trust But Verify

- **zk-smoketest**
  - https://github.com/phunt/zk-smoketest
  - Verify new, updated, & existing installations
  - Identify latency issues

- **zk-top**
  - https://github.com/phunt/zktop
  - Unix “top” like utility for ZK

- **4 letter words/JMX (e.g. ruok, srvr)**
  - http://zookeeper.apache.org/doc/current/zookeeperAdmin.html#sc_zkCommands
  - Use "stat" to get an idea what your request latency looks like

Best Practices

DOs

- Separate spindles for dataDir & dataLogDir
  - Avoids competition between logging and snapshots
  - Improves throughput and latency
- Allocate 3 or 5 ZK servers
- Tune Garbage Collection
- Run zkCleanup.sh script via cron

DON'Ts

- Don't co-locate ZK with I/O intense DataNode or RegionServer
  - ZK is latency sensitive
- Don't use SSD drive for ZK transaction log

Configure ZooKeeper Correctly..

..and it'll be as impenetrable as a distributed system allows.

Questions?