Question

ZooKeeper enables a new leader to update the configuration by deleting the "ready" znode, making changes to configuration znodes, and then recreating "ready."

Suppose a “ready” znode already exists from before. A new leader deletes it and starts making configuration changes. While this was happening, a client still seeing the old “ready” znode started reading the configuration options while the leader was changing them (like a write-read race condition). Can this scenario occur in ZooKeeper? Why or why not?

Answer

Yes, this scenario can occur in ZooKeeper if the client does not set a watch on the "ready" znode.

Watches in ZooKeeper are optional, meaning that if a client does not set one (using a flag), it won’t receive notifications about changes to "ready." Without a watch, the client might see the old "ready" znode and assume the configuration is stable, potentially reading configuration data while the new leader is still making updates.

From the paper: "When a client issues a read operation with a watch flag set, the operation completes as normal except that the server promises to notify the client when the information returned has changed".

When a client has a watch on the "ready" znode, it is notified of any deletion or recreation events in the correct order, ensuring that it sees a fully updated configuration after the new leader finishes changes. Without a watch, however, the client misses these notifications and can’t benefit from ZooKeeper’s ordering guarantees.