Summary: Chubby

Here is a quick summary of Chubby for you!

We'll cover the following

Summary#

1. Chubby is a **distributed lock service** used inside **Google** systems.
2. It provides **coarse-grained locking** (for hours or days) and is not recommended for fine-grained locking (for seconds) scenarios. Due to this nature, it is more suited for high-read and rare write scenarios.
3. Chubby's primary use cases include naming service, leader election, small files storage, and distributed locks.
4. A Chubby Cell basically refers to a **Chubby cluster**. A chubby cell has more than one server (typically 3-5 at least) known as replicas.
5. Using **Paxos**, one server is chosen as the master at any point and handles all the requests. If the master fails, another server from replicas becomes the master.
6. Each replica maintains a small database to store files/directories/locks. Master directly writes to its own local database, which gets synced asynchronously to all the replicas for fault tolerance.

7. Client applications use a Chubby library to communicate with the replicas in the chubby cell using RPC.
8. Like Unix, Chubby file system interface is basically a tree of files & directories (collectively called nodes), where each directory contains a list of child files and directories.
9. **Locks**: Each node can act as an advisory reader-writer lock in one of the following two ways:
    - **Exclusive**: One client may hold the lock in exclusive (write) mode.
    - **Shared**: Any number of clients may hold the lock in shared (reader) mode.
10. **Ephemeral nodes** are used as temporary files, and act as an indicator to others that a client is alive. Ephemeral nodes are also deleted if no client has them open. Ephemeral directories are also deleted if they are empty.
11. **Metadata**: Metadata for each node includes Access Control Lists (ACLs), monotonically increasing 64-bit numbers, and checksum.
12. **Events**: Chubby supports a simple event mechanism to let its clients subscribe for a variety of events for files such as a lock being acquired or a file being edited.
13. **Caching**: To reduce read traffic, Chubby clients cache file contents, node metadata, and information on open handles in a consistent, write-through cache in the client's memory.
14. **Sessions**: Clients maintain sessions by sending KeepAlive RPCs to Chubby. This constitutes about 93% of the example Chubby cluster's requests.
15. **Backup**: Every few hours, the master of each Chubby cell writes a snapshot of its database to a GFS file server in a different building.
16. **Mirroring**: Chubby allows a collection of files to be mirrored from one cell to another. Mirroring is used most commonly to copy configuration files to various computing clusters distributed around the world.

System design patterns#

Here is a summary of system design patterns used in Chubby.

- **Write-Ahead Log**: For fault tolerance and to handle a master crash, all database transactions are stored in a transaction log.
- **Quorum**: To ensure strong consistency, Chubby master sends all write requests to the replicas. After receiving acknowledgments from the majority of replicas in the cell, the master sends an acknowledgment to the client who initiated the write.
- **Generation clock**: To disregard requests from the previous master, every newly-elected master in Chubby uses 'Epoch number', which is simply a monotonically increasing number to indicate a server's generation. This means if the old master had an epoch number of '1', the new one would have '2'. This ensures that the new master will not respond to any old request which was sent to the previous master.
- **Lease**: Chubby clients maintain a time-bound session lease with the master. During this time interval, the master guarantees to not terminate the session unilaterally.

References and further reading#

- Chubby paper (https://research.google/pubs/pub27897/)
- Chubby architecture video (https://www.youtube.com/watch?v=PqItueBaiRg)
- Chubby vs. ZooKeeper (https://www.youtube.com/watch?v=zokwJeukDrI)
- Hierarchical Chubby (https://www.scs.stanford.edu/17au-cs244b/labs/projects/bohn_dauterman.pdf)

- Bigtable (https://research.google/pubs/pub27898/)
- Google File System (https://research.google/pubs/pub51/)

← Back

Scaling Chubby

Next →

Quiz: Chubby

☑ Mark as Completed

Report an Issue