ELEC 477 Distributed Systems
Assignment 3 - Primary Backup

Due March 18, 2024, 23:99.

1. Purpose

In this assignment you will extend the KV server from Assignment2 to prove a simplified version of the Primary-Backup version of consistency described in Class 3 of Week 7. In the class model, a client could send a write request to any of the replicas and the write request would be forwarded to the primary for that replica.

To simplify the problem for this assignment, only the primary server will register with the Service Directory Server and all client requests will go through the primary server. If it is a put request, the request will first be applied to the primary. If successful, then it will be forwarded to all replicas, and the reply will only be sent back to the client once all agree. For the purpose of this assignment, do not worry or handle missing or delayed messages.

2. Starting Point

Again, you have the choice of using your assignment2 submission as a starting point, or the posted solution.

3. KV Server/Service Changes

As with the previous assignments, the code for the KV server and service must be the same for all of the servers. The difference will be that one will be configured by your main program as the primary, and the others will be configured as the replicas. You will have to have a instance variable that tracks if your service is the primary or the replica. All servers must have different GDMB files.

The main program will give the names and ports of the replicas to the primary server, and the name of the primary to the replicas. This can be done as arguments to the constructor or with methods to set the values. Each of the values must be forwarded in some way to the associated KV service. That is, the names and ports of the replicas to

the service on the primary server, etc. You will have to add appropriate instance variables to the KV service class to track these values.

Since the simulated DNS does not have the address associated with a given Node until the init() call has been made, your service should not attempt to translate names to addresses until init() has been called by all of the servers. This can be as simple as not calling startServices() for any of the servers until all have been setup with init(). You can also decide to delay translating replica/leader names to addresses until a message has been received by your KV service.

Modify the dispatch part of your service to first call the put method and, if successful, then send the message to each of the replicas. If the put was not successful (i.e returned false) for some reason, there is no point in updating the replicas.

As a security measure, your replica should check that the source address of a message matches the address of the primary. In the full version of the primary/backup consistency approach, this would be used to forward the message to the primary, but you do not have to implement this for this assignment. In a full version, it would also only be checked if it was a put message, but to simplify for this assignment you can check for all messages.

4. Testing

You have already tested all of the key functionality for the KV server in assignments 1 and 2. For this assignment, you should only check that the put messages are forwarded to the replica servers and the values are changed. You can do this with trace methods.

5. What to hand in

As with assignment 1, you should have multiple test cases and both a document explaining how you solved the program and a test document that explains how you tested your system to ensure it works.