Chapter 2: Replication

Lab - Writes with Failovers

Problem:

In this lab, you will attempt to write data with a `writeConcern` to a replica set where one node has failed.

In order to simulate a node failure within your replica set, you will connect to the node individually and shut it down. Connecting back to the replica set and running `rs.status()` should show the failing node with a description like this:

COPY

```
{
  "name" : "m103:27001",
  "health" : 0,
  "stateStr" : "(not reachable/healthy)",
  "lastHeartbeatMessage" : "Connection refused",
  "configVersion" : -1
}
```

With one of your nodes down, attempt to insert a document in your replica set by running the following commands:

Correct! SEE DETAILED ANSWER

```
use testDatabase
db.new_data.insert({"m103": "very fun"}, { writeConcern: { w:
3, wtimeout: 1000 }})
```

This will attempt to insert one record into a collection called `testDatabase.new_data`, while verifying that 3 nodes registered the write. It should return an error, because only 2 nodes are healthy.

Given the output of the insert command, and your knowledge of `writeConcern`, check all that apply:

Attempts Remaining: Correct Answer

Check all answers that apply:

- [x] The unhealthy node will be receiving the inserted document when it is brought back online.
- [ ] The write operation will always return with an error, even if `wtimeout` is not specified.
- [ ] `w: "majority"` would also cause this write operation to return with an error.
- [x] When a `writeConcernError` occurs, the document is still written to the healthy nodes.

See detailed answer

Proceed to next section

Assignment is Due

11d:03hr:03m
Dec 10, 17:00 UTC

Your Grade

PASS/FAIL
Submitted