Programming Assignment #2

COEN241 Cloud Computing
Department of Computer Engineering
Santa Clara University

Dr. Ming-Hwa Wang
Phone: (408) 805-4175
Course website: http://www.cse.scu.edu/~mwang2/cloud/
Office Hours: Sunday 9am-12:00pm

Spring Quarter 2019
Email address: m1wang@scu.edu
Website: http://www.cse.scu.edu/~mwang2/cloud/
Office Hours:

Due date: midnight April 28, 2019

Distributed Consensus, Part II (200 points)

Distributed consensus builds reliable systems from unreliable components. Please implement a simplified and modified (using client-server communication instead of RPCs) Raft algorithm for building fault-tolerant applications on a cluster of replicated state machines using C, C++, Java, Python, or Go.

There are the following processes need to be implemented:

- **a)** support leader/coordinator selection, and the newly selected leader should print “Leader is here!” on stdout
- **b)** support requests: ClientCommit(\(<\text{var}>\)) which should print return \(<\text{value}>\) to stdout if success or print “Failed!” to stdout, ClientCommit(\(<\text{var}>, <\text{value}>\)) which should print either “Success!” or “Failed!” to stdout, RequestVote(\(<\text{term}>, <\text{id}>\)), and empty (or heartbeat) and non-empty AppendEntries(\(<\text{index}, \langle\text{var}>, \langle\text{value}\rangle\)) or AppendEntries(\(<\text{index}, <\text{var}, <\text{value}>\)> and echo
- **c)** support timeouts: follower timeout (31 sec), candidate timeout (10 sec), and client-server timeout (60 sec)
- **d)** support log replication with add(\(<\text{var}, \langle\text{value}\rangle) and find(\(<\text{var}\)) operations and dumpLog(\(<\text{id}>\)) which display log from lowest index to most recent index

The first part of this implementation includes a), b), c) and d) above assume there is no failure, and the second part handles non-Byzantine failures. When your P2 is graded, any \(f\) of the \(2f + 1\) nodes in the cluster can be killed, and your program still works.

Nodes are statically configured with a knowledge of all other nodes in the cluster, cluster membership cannot change dynamically. Assume the first node (assume it is at 129.210.16.88) running P1 is the leader with the command:


The above command will print a port number, assume it is 9999. Other nodes running P1 are followers which each of them runs the command:

\[ \$ \text{P2 129.210.16.88 9999} \]

Student Name:

ID:

Score:

Correctness and boundary condition (55%):
Error Handling (5%): Automatic available port finding (2.5%):
Support both host name and IP address (2.5%):
Display output on both server and client windows (10%):
Modular design, file/directory organizing, showing input, documentation, coding standards, sympathy/typing point with README (20%):
Automation for client and server side (5%):

Subtotal:

Late penalty (20% per day):
Special service penalty (5%):

Total score: