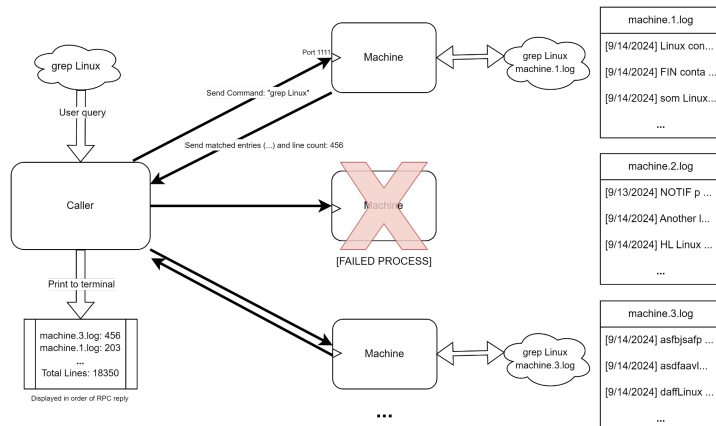


MP 1 Report: Distributed Log Querier: Pranav Pillai and Adish Jain

The diagram below describes our implementation of a distributed log querier.



First, a continuously-running “machine” process (“*go run main.go machine*”) is started on each VM. Then, any VM can be used to perform the query through a “caller” process (“*go run main.go caller grep <pattern>*”). This “caller” will dial into each VM (configurable hosts and ports); if the “caller” connects, it will begin a goroutine to send the user’s grep command via RPC. Each machine

will locally run the grep command on its VM’s log, and will also run a command that counts the number of matching lines. Then, the “machine” sends back the matching log entries and number of matches to the “caller” via the RPC. The “caller” prints the line counts and saves the entries to an output file.

For testing purposes, we have unit and distributed tests. The unit tests check the validity of individual functions, whereas the distributed tests check end-to-end validity. For distributed tests, we randomly generate log files on each VM. Then, we check the line count output of common and uncommon patterns, and complex regular expressions. We also check the line count output when one VM has failed. We don’t check the actual output, because the lines are randomized before the pattern is inserted, but the number of matching lines is controlled.

Finally, we can explore the latencies from various query types with 4 machines of 60MB log files each, and the results are shown on the right. Over all the tests, the average query latency was 0.792 seconds. We test a common query, missing query, uncommon query, and complex query. As we expected, the queries with many matches took significantly longer than the other queries. The plot suggests that the average latency is related to the number of matching lines. In the future, we may want to control for other factors, such as pattern length, for more definite results.

Average Latency by Query Type (5 trials each)

