Thomas Kao (tkao4), Melissa Jin (mqjin2)

CS 425 MP1 Report

Design:

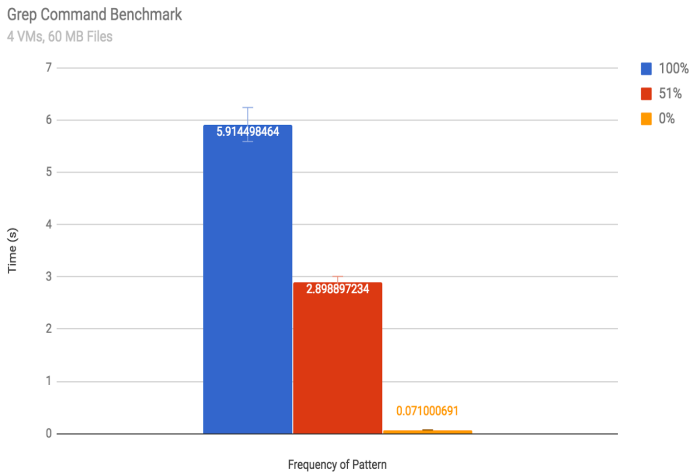
We designed our distributed log querier using a client-server model. In the client, the user inputs the grep flags and the pattern that they want to include and then the querier builds the grep command. Then, for each machine, the querier spawns a goroutine that connects to the server and sends over the command. The server runs the grep command on its log file and sends the results back over the connection. The client then prints the results to a file on the query machine with a header denoting the log file that it came from.

Unit tests:

We wrote unit tests for the different components that we used in our system. To test the functionality of the grep search function, we generated a log file and ran the function on a frequent and an infrequent pattern in the file. We tested the socket connection by creating a local dummy server, connecting with our client, and then sending information through that connection. We also tested other helper functions, such as writing the output to a file and extracting the machine number from the host address.

Query Latency:

We ran three different tests on our grep implementation to determine the query latency. The test was run on 4 vm's each containing a 60 MB log file. The first test queried for a frequent pattern "GET" which found a total of 2,308,449 lines on all 4 machines. Next, we tested the somewhat frequent pattern "image" which had a total of 1,176,132 lines. Lastly, we tested the rare pattern "test" which showed up in 2215 total lines.



Run #	"GET" (~100%)	"Image" (~51%)	"Test" (~0%)
1	7.203710701s	3.33137767s	81.221831ms
2	5.743475522s	2.791103207s	69.385039ms
3	5.470773162s	2.85299395s	67.873663ms
4	5.616432143s	2.795820283s	66.880194ms
5	5.538100791s	2.723191061s	69.642729ms
Average	5.914498464s	2.898897234s	0.071000691s
Standard Error	0.32547871837573s	0.11005987196053s	0.00000260463907s