

MP0: Event Logging Report

Sirui Wang(siruiw4)

Yichuan Hu(yh42)

GitHub repo link: <https://github.com/Stranger3333/CS425-MP0.git>

Full revision number: b2c81f6d46d3ad9f036b1feb8fccd0c5ad38db22

Cluster Number

Cluster Number: 05

Instructions for Building and Running Code

We followed the instructions of MP0 page and Campuswire guidance. We used the tcp connection for this MP.

To start the central logger, which named as logger.py, simply type `% ./logger 1234`, this used the #43 post on campuswire as a reference.

Note: When testing my code, I used the port 5566, and I used my first assigned VM ip address as the logger's address, which is 172.22.158.15. So when grading my program, please use the same ip or you need to change the variable HOST to the ip address that you want to test.

For the Node, which is the client side, please start them after you activated the logger.py. The Node is named as node.py. To run the node.py, we used the same command on MP0 page, `% python3 -u generator.py 0.1 | ./node node1 10.0.0.1 1234` Similarly, we used the 172.22.158.15 and 5566 when testing.

Note: To end one node, please use ctrl+c. I did not deal with the manual terminate on the client side, however, on the logger/server side, as the requirement said, it will print <timestamp> <node_name> disconnected to the console.

Graphs were generated by Jupyter Notebook, which named as 3&10nodeplot.ipynb. Data that used to generate the graph were stored in the 3node and 8node folder, and all the data were saved as .csv files. Each .csv file contains 5 columns, timestamp of generator, node name, message, time difference between first column and receiving timestamp of logger, string length.

Delay and Bandwidth Measurement

Based on our understanding of the MP0 requirement, we start each node by their name sequence. So firstly node1, then node2, by this pattern. There is some time error during this process since we activate each node one by one, but we think it can be neglected.

Delay is just the entries in the fifth column. We set the timestamp of the first received message as start time and count the delay in 100 seconds after that.

Bandwidth is counted as the total bytes of message received by logger per second, which is the sum of message length, as the instruction suggested.

4 Graphs of the Evaluation

1. 3 nodes, 2 Hz each, running for 100 seconds
2. 8 nodes, 5 Hz each, running for 100 seconds



