Riley Cooper : cooperra@vt.edu
Sarah Kharimah : sarahkh@vt.edu
Zachary Yee : zjwy99@vt.edu
Anthony Clifton : antclift@vt.edu
ECE 4564, Assignment 2

Section 1: Objectives

In this assignment, we must create a system that monitors cpu and network utilization of two host raspberry pis, sends the information through a message broker, and stores the information in a database on a monitor. The message broker uses RabbitMQ to control the flow of messages. The monitor uses MongoDB to store the information, and must display the current, highest, and lowest utilization statistics whenever it receives information.

The message broker must support multiple user credentials to various virtual hosts that isolate access to resources on the server. The programs on the sender and recipient provide this information through command line arguments. The programs also must handle exceptions thrown from rejected credentials. The broker uses direct exchange to separate messages from the two hosts so that the monitor can differentiate the data it receives.

Section 2: Team Responsibilities

Section 2.1 CPU Utilization and Network Throughput

The calculation of CPU utilization and network throughput was done by Sarah Kharimah. The CPU utilization value varies between 0 and 1, and these values are obtained from parsing /proc/stat file and calculating the CPU usage over a time difference (delta t). Similarly, network throughput values were obtained by parsing /proc/net/dev file for lo, wlan0, and eth0 receive and transmit bytes and observing the value difference between receive and transmit bytes over a time difference (delta t). The CPU utilization and network throughput values were then packaged into a JSON object and sent to the message broker.

Section 2.2 RabbitMQ

Riley Cooper configured the RabbitMQ server and wrote the communication code to and from the server. He created a new virtualhost for the team and created users for each member of the team on that new host. None of the new users possess access to the virtual host '/'. Only guest has that access. He wrote the communication code with the pika library. It uses an exchange called pi_utilization and the routing keys host1 and host2. The program uses command line arguments to pass the necessary information to the pika library to contact the host.

Section 2.3 NoSQL

Zachary Yee used the NoSQL database program MongoDB to store the CPU utilization and network throughout data. The database ran on a localhost in order to minimize the use of another

network. The database can accept json objects and query data within those objects to find highest and lowest values for various fields. The pistatsview had to import the pymongo library in order to communicate with the MongoDB database. The code sends the query and remove commands to the database. He also had to parse the json object to output the necessary data.

Section 2.4 Interfacing
The interfacing and connecting of the modules was done by Anthony Clifton. This process involved allowing the user to connect with their information of choice from the command line, should they wish to connect to a different message broker. The modules also had the be connected with each other this meant making alterations to the code produced in the other parts to make it applicable, such as putting the database interactions on a loop. Finally, the monitor Raspberry Pi needed to be configured to support an LED indicator that would display the CPU usage values. The LED was connected and controlled with the Pi's GPIO which would display either green for low usage, red for high usage, or yellow (green + red) for moderate usage.

Section 3: Conclusions
We were able to calculate the Raspberry Pi stats and utilize RabbitMQ and NoSQL by the end of this assignment. We gained knowledge in and deeper understanding of Python file parsing, proc filesystem, JSON encoding and decoding, and database. Navigating through the proc filesystem and finding and parsing the right values to calculate were particularly tricky. Perusing the proc filesystem API consumed a significant amount of time.

The largest effort in configuring the RabbitMQ server lay in learning about the access control and configuring the users. The code we mostly reused from the RabbitMQ tutorials. We also had to go through the pika documentation to find out how to submit the credentials to the server to authenticate the different users. We still have more to learn about how the permissions each user has within the virtualhosts restricts their abilities to perform certain actions.