Jacob Stallings
Quinton Chudik
Fall 2020

CS 4287-01 Assignment 1 - Effort Expended

Effort:

The first thing we did was set up the local VM's with VirtualBox. The steps were: download and install VirtualBox, configure NAT network, and download Kafka (even though we realized we didn't have to use the NAT network or Kafka on the local VM's). This took around 30 minutes total.

Then we set up a Github repository for our producer and consumer code. This took only 5 minutes as we had a lot of experience with Github prior.

The next step was instantiating the Chameleon VM's. This took about 15 minutes and was simple due to only having to follow the instructions in the Box folder.

Next we had to figure out how to ssh into the VM's. It took a while because we didn't realize what the default user for Chameleon VM's was. We were able to find instructions online for this. This step took about 15 mins.

Then we had to install Kafka on the VM's 2 and 3. This step was pretty straightforward as it's well documented. This took us about 15 minutes.

Next we downloaded and setup CouchDB on VM3. We started this before adding security groups or updating the firewall configuration with ufw, so we got stuck briefly before realizing this and making the appropriate changes. After making the changes we had to delete all the files to redownload and install CouchDB. The second time we also set up the user properly. This took us around an hour due to the couple of things we forgot at the beginning.

Once we got CouchDB running we started looking for a good Python API for it, we first found one just called couchdb, but after looking into it more we decided to go with pycouchdb because it was better maintained and documented. It took us about 30 minutes to research them and find out how to implement the code for our intended purpose. The pycouchdb was well documented and the semantics were straightforward, so the actual coding part of it was easy.

Finally we updated the consumer and producer Python files with the correct IP information and changes to support our setup. This was relatively easy as json is widely used and intuitive and the rest was essentially given to us as scaffolding code. The coding in total took us around 30 minutes.

We then attempted to run and test our implementation. We had earlier changed the kafka configuration files to what we deemed correct, however after our failed server startup I posted in the slack channel and got a response that helped us to correct the config files. We spent a good 20 minutes messing with it before we reached out for help.

After we got the config files up and running we were able to successfully implement the final design, following the steps outlined in our README document.

As a final measure to confirm our system worked as a whole, we queried the database using a curl command (See README) and we were able to see that the output of top was present. It took around 30 minutes to find and successfully run the command to check.

Reflection:

We found this assignment to be slightly tedious and we were glad we only had to set up a few virtual machines. Having to do something similar for lots of VM's would be painstaking. Most of our hitches were only slight and we were able to find solutions to our problems quickly. The most helpful thing was reading documentation and using forums. Putting everything together to run as a whole was the most difficult task for our team, setting up the individual pieces really wasn't too bad.