CS312 Homework #5

February 25, 2015

# Instructions

Please submit all answers as a single text file via T.E.A.C.H using the naming format `$onidusername-hw5.txt`. This homework is due at 4pm on Wednesday, Mar 4th.

# Questions

1. Define both horizontal and vertical scaling. Explain the differences between them. Give one example of each that is not in the slides.
2. Explain the CoreOS update process. What is it modeled after? Do you think this is a good or bad update model? Explain.
3. Which consensus algorithm does CoreOS use for `etcd`? Give a very brief summary of how the algorithm works.
4. Name two features of the Linux Kernel that makes container technology possible for Linux.
5. Describe the primary differences between virtual machines and containers. Why would you use one over the other?
6. Name three scenarios that virtual ips are good for increasing redundancy.
7. Name three ways that horizontal scaling can add complexity. Briefly explain why the complexity is necessary to scale horizontally.

8. Use a Dockerfile to build the cs312 repo and output it over nginx. Use the following as a base:

```
FROM nginx
MAINTAINER <your email here>

RUN apt-get update

# Install build-essential, python-virtualenv, git here

# Clone the cs312 repo
RUN git clone https://github.com/osuosl/cs312

# Run the cs312 build script

# Copy the resulting build to /usr/share/nginx/html
RUN cp -r cs312/build/html /usr/share/nginx/html
```

   Hint: You can use `docker build -t <tag> .` to build your docker image and test it

9. Create a systemd unit file for the docker image in the previous question. Here is a base file to get you going:

```
[Unit]
Description=cs312 service
BindsTo=cs312.service

[Service]
ExecStartPre=-/usr/bin/docker <kill old container here>
ExecStartPre=-/usr/bin/docker <remove old container here>
ExecStart=/usr/bin/docker <run new container here, make sure
    ↪  to expose port 80!>
ExecStop=/usr/bin/docker <stop running container here>
```

10. Suppose you have a system running with a single load balancer, three web nodes, and two database nodes. What is the single point of failure? How could you get rid of this single point of failure? After fixing it, what other points of failure might be singular?