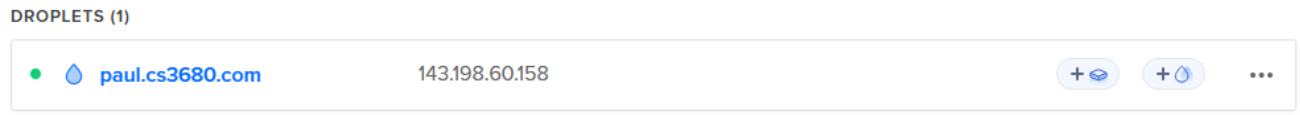
CMPS 3680 Guided Lab 7

Spring 2024

Part 1 - Digitalocean Setup

1. For this assignment you will need a linux server connected to the internet. We will use Digital Ocean for this.
2. Use the following link to sign up for Digital Ocean and receive a $200 credit: https://try.digitalocean.com/freetrialoffer/
3. Once you are logged into Digital Ocean, create a new project called cs3680
4. In your cs3680 project, spin up a new Droplet with the following:
    a. **Region:** San Francisco
    b. **Datacenter:** San Francisco · Datacenter 3 · SFO3
    c. **OS:** Ubuntu (Version 22.04 (LTS) x64)
    d. **Droplet Type:** Basic
    e. **CPU:** Regular ($4/mo)
    f. **Authentication:** Password
    (If you would like to use SSH Key from the get-go, come see me)
    g. **Hostname:** Your site URL (for example, mine would be paul.cs3680.com)
5. Once you create the droplet it will show up in your project like this (notice the ip address):



Part 2 - User Setup (Text in RED should be replaced with YOUR VALUES!)

1. You can now login to your new server using your server's IP address:
```
ssh root@XXX.XXX.XXX.XXX
```
2. Once logged in, create a new user with the following command:
```
adduser username
```
3. Give the new user SUDO capabilities:
```
usermod -aG sudo username
```
4. Log out of your server:
```
exit
```
5. Log back into your server with the new user name:
```
ssh username@XXX.XXX.XXX.XXX
```
6. Once logged in with your new user, run this command to make sure they have SUDO capabilities and update your machine's apt-get package library:
```
sudo apt-get update
```
7. In your home folder, create a www folder with an index.html file in it:
```
cd ~
mkdir www
echo "HELLO WORLD" >> www/index.html
```
8. Set folder permissions so they will be accessible from apache requests:
```
sudo chmod 755 /home /home/username /home/username/www
```

Part 3 - Apache Setup (Text in RED should be replaced with YOUR VALUES!)

1. Install Apache2:
   ```
   sudo apt-get install -y apache2
   ```
2. Once apache is installed, you should be able to enter the IP address from your server in any web browser and see a default apache2 website. In the following steps you will set up your own apache config and replace the default one.
3. In the **/etc/apache2/sites-available** folder, create a new config file:
   ```
   cd /etc/apache2/sites-available
   sudo vim username.cs3680.com.conf
   ```
4. The contents of the config file should look like this:
   ```
   <VirtualHost *:80>
           <Directory /home/username/www/>
                   Options Indexes FollowSymLinks
                   AllowOverride None
                   Require all granted
           </Directory>

           ServerName username.cs3680.com

           ServerAdmin wroyer@csub.edu
           DocumentRoot /home/username/www/

           #LogLevel info ssl:warn

           ErrorLog ${APACHE_LOG_DIR}/error.log
           CustomLog ${APACHE_LOG_DIR}/access.log combined
   </VirtualHost>
   ```
   Explanation:
   - The **<Directory>** section allows apache to use the **www** folder in your home folder (from Part 2)
   - The **<VirtualHost *:80>** section defines how apache should handle requests that come in on port 80
   - **ServerName:** Tells apache which url/domain this virtual host will accept requests from
   - **ServerAdmin:** Defines an email address for the admin of the virtual host. This gets set as a global variable and can be accessed from server side code. For example, in PHP it can be access using the super global **$_SERVER['SERVER_ADMIN']**
   - **DocumentRoot:** Defines the default path where static web files will be served from. Notice we set this to the full path of the **www** folder we created in Part 2
5. Once you have created the new config file, you must tell apache2 to enable it:
   ```
   sudo a2ensite username.cs3680.com.conf
   ```
6. Make sure you also DISABLE the default config:
   ```
   sudo a2dissite 000-default.conf
   ```
7. Finally, restart Apache2 to apply the changes:
   ```
   sudo systemctl reload apache2
   ```
8. Once you have applied the new config and restarted apache, you should be able to revisit the IP address in any web browser and see a blank page with “HELLO WORLD”.

Part 4 - UFW (Firewall) Setup (Text in RED should be replaced with YOUR VALUES!)

1. Now that your apache2 server is up and running, you’ll want to turn on UFW to protect your server.
2. ufw should already be installed on your server.
3. Make sure OpenSSH and Apache are allowed through the firewall:
   ```
   sudo ufw allow OpenSSH
   sudo ufw allow Apache
   ```
4. Enable the firewall:
   ```
   sudo ufw enable
   ```
5. To check the status of the firefall:
   ```
   sudo ufw status
   ```

Useful Commands

- Check status of Apache2:
  ```
  sudo systemctl status apache2
  ```
- Stop Apache2:
  ```
  sudo systemctl stop apache2
  ```
- Start Apache2:
  ```
  sudo systemctl start apache2
  ```
- Restart Apache2:
  ```
  sudo systemctl restart apache2
  ```
- Reload Apache2 configs without restarting:
  ```
  sudo systemctl reload apache2
  ```
- See list of apps allowed through the firewall:
  ```
  sudo ufw app list
  ```
- Check status of Apache2:
  ```
  sudo systemctl status apache2
  ```
- To monitor a file in real time, you can use the command **`tail -f`**
  For example, to see ssh logins/attempts in real time try:
  ```
  sudo tail -f /var/log/auth.log
  ```

Lab Submission

To receive credit for this lab send an email to **`wroyer@csub.edu`**

- Make sure to include 'CMPS3680' in the subject line
- In the body of the email, include:
  - The **`ServerName`** you used in your Apache config (from Part 2)
  - The **`IP address`** of your server

Feel free to edit your **`~/www/index.html`** file and create an interesting landing page for your site.