### **Documentation for Installing a containerized Apache Webserver and Hardening it**

**Getting Started:**

1. Log into your Linode machine via SSH using the steps outlined in the “Virtualization Lab” documentation.
2. Once you have established a connection, you can start installing the Apache web server using Docker by following this documentation.

**Pulling in the Docker Image:**

1. Pull the Apache based Docker image we are using to install the container onto your machine by typing:

***docker run -d --name WebServerLab -e TZ=MST -p 80:80 ubuntu/apache2:2.4- 20.04\_beta***

1. Now open a browser on your local machine and connect to the URL:

***http://[Your Linode Server’s IP]***

Note: This should bring up the default Apache web page for your new server\*

**Getting Started with the Container:**

* We will remote into the container install a text editor so we can configure the server.

1. First figure out your container ID or container name by typing:

***docker ps***

\*Note: If you used to command provided in the previous steps, the container name should be “WebServerLab”.

1. Then, use this name to get into the container by typing:

***docker exec -it [Your Apache Container’s Name] /bin/bash***

1. Now, update the container by typing:

***apt-get update***

1. Finally, you can install the text editor by typing:

***apt-get install nano -y***

**Securing SSH:**

Disabling Remote Root Access and Adding a new “dev” User:

* In this section we will disable remote access to the root account of the server and set up proper accounts to be used with just the level of privilege required for each to operate.
  1. Back in the terminal, create a user account called “dev” for the development team to use by typing:

***useradd –m –s /bin/bash dev***

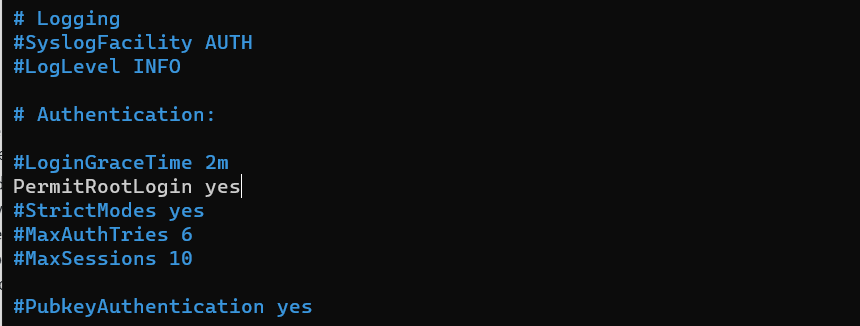
* 1. Now, create a safe and unique password for your new account by using this command to prompt a password change (after which you type in your new password):

***passwd dev***

***[Your New Password]***

* 1. Now you can disable remote root account access by first editing the SSH configuration file by typing:

***nano /etc/ssh/sshd\_config***

* 1. Once in the file, press the down arrow key until you find the line that says: “PermitRootLogin yes” (as is pictured below).
  2. Then, simply add a “***#***” symbol in front of the line (just like all the other lines). Once done, press ***ctrl+X***, and then type ***Y*** and hit ENTER.
  3. Finally, type the command:

***systemctl restart ssh***

* 1. Re-login with the “dev” user credentials you created, since the root account is now disabled on SSH.

Setting up SSH Keys for the “dev” User:

* This part will guide you through setting up private key authentication for the newly created “dev” account.
  1. Open a new PowerShell terminal on your machine as well as any other computer you want to have to access the Linode “dev” account.
  2. Now generate a private SSH key for your new “dev” account by typing:

***ssh-keygen –t rsa***

* 1. Press ENTER when you are prompted to enter a save location.
  2. Then type ***y***. Press ENTER twice when prompted for a new passphrase.
  3. Now copy your public key to the server by typing:

***type $env:[Your User Profile Name]\.ssh\id\_rsa.pub | ssh dev@[Your Linode IP Address] “cat >> .ssh/authorized\_keys”***

Disabling SSH Password Authentication

* Lastly, we will disable the password authentication method completely for SSH connections.
  1. Log back into your Linode using the “dev” account
  2. Switch to the root user by typing:

***su root***

* 1. Edit the SSH config file by typing:

***nano /etc/ssh/sshd\_config***

* 1. Use the down key till you find the line that says: “PasswordAuthentication yes”.
  2. Replace the “yes” with a “no” in that line.
  3. Press ***CTRL+X*** then type ***y*** and press ENTER
  4. Now restart the SSH system by typing:

***systemctl restart ssh***