Introduction

Bear with me, this write-up is going to be a long one. It's mostly procedural documentation though!

So, if you're a member of NextHop you're probably a *networking kind of guy or gal*. You know that the internet is more than web pages and such. However, web pages and such are a BIG part of the modern internet. How else would the masses get their cat pictures?

The likelihood is that most of you have some basic exposure to HTML (Hypertext Markup Language). Flat HTML pages are fine and dandy for small websites, but once you start scaling up sites become more dynamic. Languages run on the back end, in the application "stack," to dynamically render and generate content.

While there are many flavors of stacks out there, today we'll be focusing on a basic LAMP stack. LAMP stands for Linux, Apache, MySQL and PHP. Using the services in a LAMP stack, you'll be able to support a dynamic web application.

Keep in mind we're a networking club. Our goal tonight isn't to make pretty websites. We're trying to build the stack to hold up a website.

First Thing First…

In addition to being a *networking kind of guy*, I'm also a CentOS kind of guy. Because my word as NextHop president is law (ha!) we're going to be doing this lab in a CentOS environment. To get started, open up VMWare and activate your CentOS 7 VM.

Open up the terminal and install the LAMP stack using the following command.

```
sudo yum install httpd php mariadb-client mariadb-server -y
```

The stack is now installed. Done? Nope.

We're going to need you to point your VMs DNS over to our NextHop DNS server. The IP will be written on the whiteboard. If you need help, just ask!

Configuring httpd (Apache)

No good Systems Administrators ever write their own code. Also I'm lazy. Mainly, there's no point in reinventing the wheel.

Apache is configurable in two different ways, the easy way (single host) and the hard way (multiple hosts). This is NextHop. We're pretty bad-ass. Go big or go home. Use the provided DigitalOcean tutorial to setup virtual hosts on your CentOS 7 box.

First and foremost add a new directory to the HTTPD directory (/etc/httpd). Call the directory **`virtualHosts`**. Use your favorite text editor (I hope it's Vim) and add the following to the end of the **`/etc/httpd/conf/httpd.conf`** file, **`IncludeOptional virtualHosts/*.conf`** .

By adding this directory you're including individual host declarations to be included from the **`virtualHosts`** directory.
Each file in that directory needs to end with **`.conf`** .

We're not very creative people, so name your site after the PC you're working on **`.local`** , so if you're working on Ariel 1, call your .conf file **`ariel1.nexthop`** . Come see one of the friendly NextHop people and tell us your website name and IP address.

Inside of the conf file, include the following (replacing ariel1 where appropriate):

```
<VirtualHost *:80>
    ServerName www.ariel1.nexthop
    ServerAlias ariel1.nexthop
    DocumentRoot /var/www/ariel1.nexthop/public_html
    ErrorLog /var/www/ariel1.nexthop/error.log
    CustomLog /var/www/ariel1.nexthop/requests.log combined
</VirtualHost>
```

This is your virtual host entry. This is what will let your web server know about your virtual host and where to find it. If you couldn't guess, you're going to need to create the DocumentRoot directories specified in the VirtualHost declaration. To do so, issue the following command:

```
sudo mkdir -p /var/www/ariel1.nexthop/public_html
```

Create an **`index.php`** file in the newly created **`public_html`** directory. Make a super basic webpage, with the added twist of PHP.

```
<!DOCTYPE html>
<html>
<head>
     <title>NextHop | LAMP</title>
</head>
<body>
     <?php echo “Hello, World!”; ?>
</body>
</html>
```

Restart httpd with the sudo systemctl restart httpd command and navigate over to ariel1.nexthop and check out your snazzy new web page. Please note that whatever computer you navigate on has to have the DNS server listed on the whiteboard set.

Configuring MariaDB (MySQL)

So now we've covered the L, A and P in LAMP stack. Databasing can be a bit of a daunting task to cover, especially in the context of a Build-It-Night. For everyone's sanity, head over to http://nexthop.network/bin/lamp and download the database and PHP file provided to your desktop. Put those to the side, we're about to configure MariaDB.

There's no shame in using setup wizards! In fact, we're going to use one to configure our database environment. To do so issue the command sudo **`mysql_secure_installation`** . Once you've gone through the configuration, enter into the MariaDB client as root. To do so, issue the command mysql -u root -p and hit enter. It will then prompt you for a password (the one you set in the secure installation setup wizard).

From there we're going to import the database that you downloaded from the NextHop site. To do so type **`.\ path\to\your\LAMP.sql`** into the MariaDB command line and hit enter. Once it successfully displays an entry, type **`exit`** to close out of the MariaDB client.

Next, copy the dbTest.php file to your public_html folder. To do so, use the command

```
cp /path/to/your/dbTest.php /var/www/ariel1.com/public_html/dbTest.php
```

Open up the **`dbTest.php`** file and edit the login credentials for MySQL. Once that's all done you should be able to navigate over to http://ariel1.nexthop/dbTest.php and see some cool stuff. Check out the **`dbTest.php`** file. Feel free to ask any questions as to how it works.

Conclusion

So yeah… that's pretty much it. You have a fully functional LAMP stack. Hopefully you understand a bit about how it works!