

Experiment 13: Cloud based server

Aarya R Shankar

May 13, 2017

1 Aim

Application deployment on a cloud-based LAMP stack/server with PHP eg: OpenShift, Linode etc.

2 Procedure

We know how to write a php application. So let us write a simple php application to show the multiplication table of the input number. Push those into a git repository. If you already have a PHP application ready in a git repository, you may skip this step.

Now we need to host it on a cloud based LAMP stack or server. For this purpose, I have chosen OpenShift Online, which is Red Hats public cloud application development and hosting platform that automates the provisioning, management and scaling of applications so that you can focus on writing the code only. You may also use other softwares like Linode, etc, but others may be proprietary softwares. Let us install OpenShift in our system. (Currently we are installing for Debian/Ubuntu. Please refer to the official documentation of Openshift for installation on other Linux distributions.) First we need to install Ruby by running this command:

```
sudo apt-get install ruby-full
```

To check if Ruby is installed correctly, run

```
ruby -e 'puts "Welcome to Ruby"'
```

If the output is `Welcome to Ruby`, ruby is installed correctly. Rubygems is a part Ruby package now. But older Ubuntu repositories have it separately,

so if that's the case, we may need to run the following command to install Rubygems or to verify that it is already installed.

```
sudo apt-get install rubygems
```

Now we need to install client tools by running this command.

```
sudo gem install rhc
```

Now we need to sign in to OpenShift Online [here](#).

We may either use our RedHat account, if we already have one or are planning to create one, or we may sign in using our one of our social accounts. I created a RedHat account, and signed in using that.

After you sign up and log in, apply for a server. You may choose either East US or West US as a server location. If it says that you have to wait to avail a server, log out and log in again. Then you'll see a page having a blue button named "Open Web Console". Click that.

Create a new project having the name as, say, foss-lab. Provide the link of your git repository to add to host it in the server.

Web console is very helpful in managing our application, but for a higher level of control, we may use the command like tools. Let us use the *oc* command line tool. It can be downloaded and installed by running the following commands:

```
curl https://mirror.openshift.com/pub/openshift-v3/clients/3.5.5.10/linux/oc.tar.gz -o oc.tar.gz
tar xvf oc.tar.gz
```

We may run the command by

```
./oc
```

Now we can login in to the server by running:

```
./oc login https://api.starter-us-east-1.openshift.com
```

Now we need to switch to our project foss-lab, by running:

```
./oc project foss-lab
```

Next we can see the current status of the project by running:

```
./oc status
```

To view your application, open your web console, navigate to your application, go to Applications, then click on Routes tab. Click on the hostname. You can now see your application running in the browser.

3 Result

The php application is successfully deployed in the cloud based server, OpenShift, and is available at <http://multiplication-table-foss-lab.1d35.starter-us-east-1.openshiftapps.com/calculate.php>.