Lab 1: Push an Application

Description: During this lab we will push an application, customize our push and review the files generated during the pushing process.

1. Verify your access to Pivotal Web Services. You will set the API endpoint to Pivotal Web Services (api.run.pivotal.io) and login in with your credentials.

*cf api httsp://api.run.pivotal.io*

*cf login*

1. Change to the odca-paas-workshop directory. We will push a simple “Hello World” Node.js application called web.js (web.js).

During this first installation we will see how easy it is to push applications to Cloud Foundry. The command below pushes an application named web.js to Pivotal Web Services (PWS).

All applications pushed to PWS share the same domain name (cfapps.io), so we need to provide a unique name for our application. We suggest adding your initials or a unique number to your application name. The URL to your application will be the appName.cfapps.io. Custom domains can be added to your account and reference the internal URL.

The first step is to open the manifest-web.yml file using your favorite editor and modify the application name to be unique.

---

applications:

- name: webjs-<user number or initials>

memory: 256M

Then the app can be pushed to your account in cloud foundry with the following command:

*cf push -f manifest-web.yml –i 2*

Open a web browser and visit http://web.js-<user number or initials>.cfapps.io

[Note: this is not currently working. We can add with time to both apps]

* otice the IP address displayed at the top, this is the IP address of the container serving your instance.
* Verify the mongo service is not bound to the application. We’ll complete this in the next lab.

1. When an application is pushed, a number of steps are completed and we want to see what the platform has configured. This can easily be done using the Files command. Let’s review what the files command can do.

The first step is to list all the files associated with our application.

*cf files map.js-userX*

Next we can get more granular to look at specific files. For example, let’s review the environment variables in our container. We can see 2 special variables, VCAP\_APPLICATION which gives runtime information about our application and VCAP\_SERVICES which gives runtime information about the services bound to our application. You can review the assigned port and also assign your own environment variables when deploying.

*cf files web.js log/env.log*

[Note: this is not currently working. Has this command changed?]

*C02HV0SADV31:PaaSTutorial wlund$ cf files web.js log/env.log*

*Getting files for app* ***web.js*** *in org* ***Northwest*** */ space* ***development*** *as* ***wlund@gopivotal.com****...*

***FAILED***

*Server error, status code: 404, error code: 0, message:*

1. Now log into “run.pivotal.io” and let’s check the health, running instances, route and other details for the application we’ve pushed. Access http://console.cfdemo5.fe.gopivotal.com with your credentials.

You’ll need to select the same space you’ve first targeted your environment to.