CLOUD **FOUNDRY**

Pivotal Cloud Foundry Developer

Lab Instructions

Application Deployment using Pivotal Cloud Foundry

Version 1.11.a

Pivotal

Copyright Notice

- Copyright © 2017 Pivotal Software, Inc. All rights reserved. This manual and its accompanying materials are protected by U.S. and international copyright and intellectual property laws.

- Pivotal products are covered by one or more patents listed at http://www.pivotal.io/patents.

- Pivotal is a registered trademark or trademark of Pivotal Software, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies. The training material is provided "as is," and all express or implied conditions, representations, and warranties, including any implied warranty of merchantability, fitness for a particular purpose or non-infringement, are disclaimed, even if Pivotal Software, Inc., has been advised of the possibility of such claims. This training material is designed to support an instructor-led training course and is intended to be used for reference purposes in conjunction with the instructor-led training course. The training material is not a standalone training tool. Use of the training material for self-study without class attendance is not recommended.

- These materials and the computer programs to which it relates are the property of, and embody trade secrets and confidential information proprietary to, Pivotal Software, Inc., and may not be reproduced, copied, disclosed, transferred, adapted or modified without the express written approval of Pivotal Software, Inc.

Pivotal

Table of Contents

Chapter 1. Push to the Cloud

Estimated Time: 25 minutes

1.1. Preface

This lab incorporates many preliminary steps. You'll be learning about and accessing the "Apps Manager" for the first time.

If necessary, you'll download and install the *cf* command line tool (the CLI), and use it to login to a Pivotal Cloud Foundry instance. You'll also learn about the concept of the cf *target*.

You'll deploy your first set of simple applications to cloudfoundry with the `cf push` command.

Although this lab is an introduction to Pivotal Cloud Foundry, you'll learn a great deal in a fairly short span of time.

Enjoy.

1.2. Setup

1.2.1. Download the `cf` CLI

If you have already done this, skip to the next section.

Endpoint and credential information for accessing Apps Manager will be provided by your instructor. Apps Manager is a web application that helps you manage your applications, but it is also the place to download the `cf` CLI. The next set of steps walk you through just that.

> **Note**
>
> If you are consuming these materials via e-learning and using [Pivotal Web Services](#) you can login to Apps Manager [here](#). If you are using a Pivotal Cloud Foundry instance please see your system administrators for access to Apps Manager.

1. [Review the Apps Manager Documentation](#). Knowing where to go for help is essential.

2. Login into Pivotal Cloud Foundry with the Apps Manager. If using Pivotal Web Services (PWS), the url is:
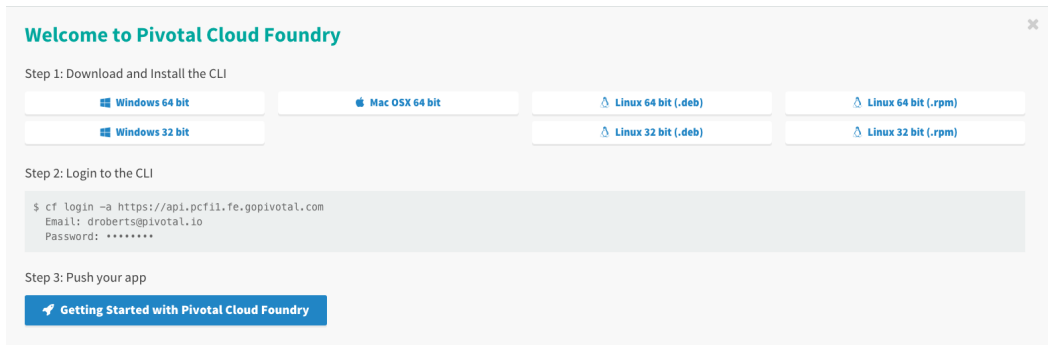
https://console.run.pivotal.io/

Note

It is common to use self signed certs in educational environments, but doing so will result in warnings from your browser. These warnings can safely be ignored (proceed through them).

3. Upon logging into Apps Manager for the first time you will be greeted with a welcome message that will walk you through the installation of the `cf` CLI.

Welcome to Pivotal Cloud Foundry

Step 1: Download and Install the CLI

⊞ Windows 64 bit	🍎 Mac OSX 64 bit	△ Linux 64 bit (.deb)	△ Linux 64 bit (.rpm)
⊞ Windows 32 bit		△ Linux 32 bit (.deb)	△ Linux 32 bit (.rpm)

Step 2: Login to the CLI

```
$ cf login -a https://api.pcfi1.fe.gopivotal.com
  Email: droberts@pivotal.io
  Password: ••••••••
```

Step 3: Push your app

🚀 **Getting Started with Pivotal Cloud Foundry**

If you do not see this dialog, the same information can be found under `Tools` (Left side).

4. Download and install the `cf` CLI for your platform.

5. Open a terminal window. Explore the `cf` CLI. In the terminal window type `cf`.

```
cf
```

6. Review the commands available with the `cf` CLI. Help on a specific command can be found using the `--help` option (e.g. `cf login --help`).

```
cf login --help
```

1.2.2. Login to Pivotal Cloud Foundry with the `cf` CLI

If you have already done this, skip to the next section.

1. Continue to follow directions for logging in with the `cf` CLI. You can copy and paste the `cf login`

command directly from Apps Manager.

```
cf login -a {{api_endpoint}}
```

> **Note**
>
> Are SSL errors preventing you from logging in? Try adding the `--skip-ssl-validation` option.

2. Review your current API endpoint, user, org, and space

```
cf target
```

What Just Happened?

You have logged into Pivotal Cloud Foundry from two different clients (Apps Manager and the `cf` CLI). You have installed the `cf` CLI and targeted a Pivotal Cloud Foundry instance. You are ready to deploy (`push`) applications.

1.3. Exercises

1.3.1. Pushing apps

Next we will push (deploy) several applications.

1. You should have been given a zip file containing most of the lab files needed by this course: `pcf-developer-XXX-labs.zip` (where xxx indicates the version of the course).

2. Unzip this file and copy `pivotal-cloud-foundry-developer-workshop` somewhere convenient. We recommend:

   - **Unix, Linux:** Copy it to you home directory: `~/pivotal-cloud-foundry-developer-workshop/`

   - **Windows:** Copy it to the root of you `C:\` drive: `C:\pivotal-cloud-foundry-developer-workshop\`

   > **Note**
   >
   > From   now   on   in   these   notes,   we   will   simply   refer   to   the   …

3

/pivotal-cloud-foundry-developer-workshop directory.

3. In pivotal-cloud-foundry-developer-workshop you will find a demo-apps directory.

   It contains four sub-directories; each directory contains an application developed in a different language.

4. Open a terminal window. First cd into the node directory:

```
cd .../pivotal-cloud-foundry-developer-workshop/demo-apps/node
```

The CloudFoundry push command assumes that the artifacts for your application are located in the current working directory.

Important

node *must* be your current directory, not demo-apps. This is a common-mistake.

5. Now push the node application:

```
cf push node --random-route -m 128M
```

6. First, let's find out what our application url is, with the cf apps command:

```
cf apps
```

Now, visit the node application in your browser, or use curl:

Example:

```
curl {{application_url}}
```

The output should be a simple Hello node message.

7. Repeat steps 3 - 4 for the php, python, and ruby applications.

Important

Make sure you cd into each directory before pushing.

Version 1.11.a

> **What Just Happened?**
>
> You just deployed four applications each based on a different language and runtime. Pivotal Cloud Foundry is a polyglot platform, meaning it supports multiple languages and does so in a pluggable way (via buildpacks)!

1.3.1.1. Questions

• What are some common items in the output that occurred when pushing each application?

1.3.2. Explore Apps Manager

1. Review the following views:

   • Org

   • Space

   • App

> **What Just Happened?**
>
> You have interfaced with Pivotal Cloud Foudry from two separate clients (`cf` and Apps Manager). Many of the operations that are available in `cf` CLI are also available in Apps Manager.

1.3.3. Clean up

1. Delete the applications you just pushed.

```
cf delete node
```

⚠️ **Warning**

This is very important for resource constrained environments. On PWS, all your application instances together may only use 2G of memory, after which you will not be able to push. Stop some previous applications if necessary to reclaim memory.

5

Version 1.11.a

Repeat to delete the `php`, `python`, and `ruby` applications.

1.4. Spring Music

Check out the Cloud Foundry [sample applications](#).

[Spring Music](#) is a favorite.

1. `spring-music.war` is also included in your lab files (under `demo-apps`) Push it to Cloud Foundry. To keep the memory usage down specify 768M.

   Note

   Remember to use `--random-route` or the `-n <hostname>` to ensure a unique URL.

2. Once it is running, view it in the App Manager.

3. Finally stop and delete the application.