



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.

Table of Contents

1. Log Drain	1
1.1. Preface	1
1.2. Exercises	1
1.2.1. Create a log drain	1
1.3. Questions	2
1.4. Beyond the Class	2

Chapter 1. Log Drain

Estimated Time: 25 minutes

1.1. Preface

Cloud Foundry's loggregator subsystem routes applications' log streams out of the containers from which they emanate, and makes them accessible via the `cf logs` command.

But what if you needed to maintain a month's worth of logs, and wanted to leverage third party tools or services for log analysis? Cloudfoundry provides the capability to *drain* application logs to some destination, whether it be an internal enterprise system or a third-party service.

In this lab, you'll setup a log drain for your application using a service called Papertail.

1.2. Exercises

1.2.1. Create a log drain

Pivotal Cloud Foundry does not persist logs. For long term storage of logs, drain logs to a third party log management service.

1. Read about [log draining](#).

In this section, you will configure application log draining into a log analysis tool with a User Provided Service Instance. We will use the free Papertrail service, but any tool that supports tcp syslog draining can be used.

2. Sign up for a free Papertrail [account](#). We've got some [instructions](#) to set up Papertrail.

In a nutshell:

- a. Create a Papertrail account, fill in your email and password
- b. Browse to papertrailapp.com/systems/new
- c. Select "I use Cloud Foundry"
- d. "What should we call it?" give it a name - `log-drain`

e. Click "Save"

f. Take note of the url (logs2.papertrail.com:<some-port>)

3. Create a User Provided Service Instance that streams logs to Papertrail.

```
cf create-user-provided-service articulate-log-drain -l syslog://{syslog_drain_url}
```

4. Bind the `articulate-log-drain` service to the `articulate` application.

```
cf bind-service articulate articulate-log-drain
```



Note

It can take up to ten minutes for logs to start flowing into Papertrail. At the time of writing, this was consistently 10 minutes before seeing logs in Papertrail.

Question: Is `cf restart/restage` required for log draining to start working? Why? Why not?

5. View `articulate` in a browser. Refresh as necessary to generate logs.

6. To view the logs in Papertrail do the following:

a. Click on "Dashboard"

b. Click on "log-drain"

7. Experiment. Restart the `articulate` application.

```
cf restart articulate
```

1.3. Questions

- How does this approach differ from how you get your logs today?
- What kinds of alerts or visualizations could you create from your logs data?

1.4. Beyond the Class

- Visualize your logs