



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. Application Security Groups	1
1.1. Preface	1
1.2. Exercises	1
1.2.1. Review Application Security Groups	1
1.3. Administration of application security groups	2
1.3.1. Managing ASGs	2
1.3.2. Binding ASGs to Running Applications	2
1.3.3. Binding ASGs to Application Staging	3
1.3.4. Org and Space specific bindings	3
1.3.5. Questions	3

Chapter 1. Application Security Groups

Estimated Time: 10 minutes

1.1. Preface

Application Security Groups, often abbreviated as ASGs, are a feature designed for cloudfoundry operators. Nevertheless it's important for developers to learn what they are and how they work, as it might explain why a particular application may in some cases be unable to communicate with a backing service, and how to remedy the issue.

Below is a summary of this feature, and of the cf CLI commands associated with managing application security groups. Be mindful that, as a non-administrative user, you have permission to invoke only a subset of these commands.

1.2. Exercises

1.2.1. Review Application Security Groups

1. Review the [documentation](#) on [application security groups](#).
2. List the security groups in your environment.

```
cf security-groups
```

3. Pivotal Cloud Foundry ships with multiple security groups. Pick two security groups from the list and note the differences in the details of those security groups.

```
cf security-group public_networks
```

1.2.1.1. Questions

- Do application security groups use a whitelist or blacklist approach to firewall rules?
- What are the differences in the two application security groups that you looked at?
- What are some reasons why security groups could be used?

- How could security groups affect the staging and running of your application?

1.3. Administration of application security groups

The cf CLI exposes a dozen or so commands for `admin` users to use to create and manage application security groups. The rest of this lab provides an overview.



Note

Unless you're an administrator of the PCF instance you're working with, you won't be able to experiment with these commands. Nevertheless it's important that you be aware of them.

1.3.1. Managing ASGs

When logged in as `admin`, the command `cf security-groups` lists all security groups:

```
cf security-groups
```

The following commands support managing this list:

```
$ cf create-security-group
$ cf update-security-group
$ cf delete-security-group
```

The **create** and **update** commands require that you supply the security group definition as a JSON file that encodes a list of egress rules.

The creation of a security group is only a first step. Additional commands exist to control which security groups are applied in different situations.

1.3.2. Binding ASGs to Running Applications

The command `cf running-security-groups` lists all security groups that apply to all running applications:

```
cf running-security-groups
```

Use these commands to apply (or unapply) a security group to all running applications:

```
$ cf bind-running-security-group all_open
$ cf unbind-running-security-group all_open
```

1.3.3. Binding ASGs to Application Staging

Security groups can also control the behavior of scripts that stage an application. To list all security groups that apply to the staging of applications, use the command:

```
cf staging-security-groups
```

Use these commands to apply (or unapply) a security group to the application staging phase:

```
$ cf bind-staging-security-group all_open  
$ cf unbind-staging-security-group all_open
```

1.3.4. Org and Space specific bindings

The **running-security-groups** binding is coarse-grained; it applies to all running applications in a PCF installation. For finer-grained control of security groups to specific running applications, these commands can scope the binding to a specific org or space:

```
$ cf bind-security-group all_open my-org my-space  
$ cf unbind-security-group all_open my-org my-space
```

1.3.5. Questions

- What are the differences between the different types (running and staging) of security groups?

Imagine a situation in which an application is bound to a backing database service and deployed to PCF.

- In the absence of ASG bindings, would the application be able to communicate with the backing service?

Let's assume that the backing database is a MySQL instance, which by default accepts connections on port 3306.

- As a Cloud Foundry administrator, what would you have to do in order to allow **any** running application to communicate with this type of backing service?
- What would your security group rules JSON file look like? Should the rules allow all ports over all ip addresses? Can the rules be constrained to allow only port 3306 over a specific range of ip addresses? Try to construct a security group rules file (JSON) that you'd use to create and then bind to running applications. What command would you use to define the security group? What would you name it? What command would you use to bind it?