



PRODUCT DOCUMENTATION

Scheduler for PCF (Beta)[®]

Version 1.0

User's Guide

© Copyright Pivotal Software Inc, 2013-2018

Table of Contents

Table of Contents	2
Scheduler for PCF (Beta)	3
Installing and Configuring Scheduler for PCF	4
Using Scheduler for PCF	5
Release Notes	9

Scheduler for PCF (Beta)



The Scheduler for Pivotal Cloud Foundry (PCF) tile is currently in beta and is intended for evaluation and test purposes only. Do not use this product in a PCF production environment.

This documentation describes Scheduler for PCF.

Overview

Scheduler for PCF is a service for scheduling the execution of tasks on PCF, such as database migrations, emails, or batch jobs.

Scheduler for PCF enables developers to do the following:

- Create jobs to execute Diego tasks
- Execute jobs on an ad hoc basis
- Schedule jobs to execute on a recurring basis
- View job status and history

You can interact with the service through the Cloud Foundry Command Line Interface (cf CLI) and the [Scheduler HTTP API](#).

Product Snapshot

The following table provides version and version-support information about Scheduler for PCF.

Version	v1.0.4
Release date	August 11, 2017
Compatible Ops Manager version(s)	v1.9.x, v1.10.x
Compatible Elastic Runtime version(s)	v1.9.x, v1.10.x
IaaS support	AWS, Azure, GCP, OpenStack, and vSphere

Requirements

Scheduler for PCF has the following requirements:

- MySQL for PCF v1.8 or v1.9. The service is available on [Pivotal Network](#).

Limitations

- If your app uses a buildpack that does not generate a `web` process type, such as Ruby or Python, you should do the following:

1. Before pushing your app, create a [Procfile](#) in the root directory of the app.
2. Declare a `web` process type in the file.

If you do not declare this process type, your app will not be accessible through the cf CLI after you create Scheduler jobs for it.

- The maximum number of tasks that you can schedule is determined by the memory and disk quotas in the Scheduler for PCF org and space. See [Running Tasks](#) for more information.

Installing and Configuring Scheduler for PCF



The Scheduler for Pivotal Cloud Foundry (PCF) tile is currently in beta and is intended for evaluation and test purposes only. Do not use this product in a PCF production environment.

This topic describes how to install and configure Scheduler for Pivotal Cloud Foundry (PCF).

Prerequisites

Before you install the tile, you need to have a MySQL for PCF service plan with the following configuration:

- **Memory**: **1gb**
- **Cores**: **1000**
- **Disk**: **40**

If you do not already have this plan, see [Add a Plan](#) in the MySQL for PCF documentation.

Install and Configure Scheduler for PCF

1. Download the product file from [Pivotal Network](#).
2. Navigate to the Ops Manager Installation Dashboard and click **Upload Product** to upload the product file.
3. Under the **Staging Area** button, click **Add** next to the version number of Scheduler for PCF. This adds the tile to your staging area.
4. Click the newly added **Scheduler for PCF** tile.
5. Make sure your network and stemcell configuration is complete.
6. Click **Install**.
7. Return to the Ops Manager Installation Dashboard, click **Install**, and click **Install** to install Scheduler for PCF tile.

Using Scheduler for PCF



The Scheduler for Pivotal Cloud Foundry (PCF) tile is currently in beta and is intended for evaluation and test purposes only.

Do not use this product in a PCF production environment.

This topic provides instructions for using Scheduler for PCF.

You can interact with the service through Cloud Foundry Command Line Interface (cf CLI) and the [Scheduler HTTP API](#). For general information, see [Managing Service Instances with the cf CLI](#).

Prerequisites

To start using Scheduler for PCF, you need the following:

- A PCF deployment with [Scheduler for PCF](#) installed and listed in the [Marketplace](#)
- A [Space Developer](#) account
- The cf CLI
- (Optional) The Scheduler for PCF CLI plugin, which is packaged with the Scheduler for PCF tile on Pivotal Network

Create and Bind a Service Instance

Every app and service in PCF is scoped to a [space](#). This means that an app can use a service only if an instance of the service exists in the same space.

The Scheduler for PCF service is a singleton service. Only one service instance can be created in a space.

Confirm Service Availability

For apps to use a service, the service must be available in the Marketplace. To confirm the availability of Scheduler for PCF, perform the following steps:

1. Run `cf marketplace` from the command line.
2. If the output lists `scheduler-for-pcf` in the `service` column, Scheduler for PCF is available. If the service is not available, install it. See [Installing and Configuring Scheduler for PCF](#) for more information.

```
$ cf marketplace
Getting services from marketplace in org my-org / space my-space as user@example.com...
OK
service      plans      description
[...]
scheduler-for-pcf  standard  Scheduler service
[...]
```

Create a Service Instance

To create an instance of the Scheduler for PCF service, run `cf create-service scheduler-for-pcf standard SERVICE-INSTANCE-NAME`, replacing `SERVICE-INSTANCE-NAME` with a name of your choice. After you create the service instance, this instance name appears under `name` in the output of the `cf services` command.

See the following example:

```
$ cf create-service scheduler-for-pcf standard my-instance

Creating service my-instance in org my-org / space my-space as user@example.com...
OK

$ cf services

Getting services in org my-org / space my-space as user@example.com...
OK
name      service      plan   bound apps  last operation
my-instance  scheduler-for-pcf  standard          create succeeded
```

You can create only one instance in a space. If you attempt to create more than one instance in a space, you receive an error response.

Bind a Service Instance to Your App

For an app to use a service, you must bind it to a service instance. Do this after you push or re-push the app using `cf push`.

To bind an app to a Scheduler for PCF instance, run `cf bind-service APP-NAME SERVICE-INSTANCE-NAME`, replacing `APP-NAME` with the name of the app you want to use the Scheduler for PCF service for and `SERVICE-INSTANCE-NAME` with the name you provided when you ran `cf create-service`.

```
$ cf bind-service my-app my-instance

Binding service my-instance to my-app in org my-org / space my-space as user@example.com...
OK
TIP: Use 'cf push' to ensure your env variable changes take effect
```

Manage Jobs Using the cf CLI

See the following sections to learn about the operations you can perform with Scheduler for PCF.

Create a New Job

To execute tasks related to an app, create a new job by running `cf create-job APP-NAME JOB-NAME COMMAND`:

- `APP-NAME` is the app you want to execute a task against.
- `JOB-NAME` is the name for your job.
- `COMMAND` is the command you wish to execute.

See the following example:

```
$ cf create-job my-app my-job "pwd"

Creating job my-job for my-app with command pwd in org my-org / space my-space as user@example.com...
Job Name  App Name   Command
my-job    my-app     pwd
OK
```

List Jobs

Use the cf CLI to list all jobs in a space by running `cf jobs`. See the following example:

```
$ cf jobs

Listing jobs for org my-org / space my-space as user@example.com...
Job Name  App Name   Command
my-job    my-app     pwd
OK
```

Execute a Job

You can execute a job manually. This is often useful to test the proper configuration of a job prior to scheduling it for recurring execution.

Run `cf run-job JOB-NAME`. See the following example:

```
$ cf run-job my-job
Enqueuing job my-job for app my-app in org my-org / space my-space as user@example.com...
OK
```

View Job History

You can review job history by running `cf job-history JOB-NAME`:

```
$ cf job-history my-job
Getting scheduled job history for my-job in org my-org / space my-space as user@example.com...
1 - 1 of 1 Total Results
Execution GUID          Execution State   Scheduled Time      Execution Start Time    Execution End Time    Exit Message
8a7e808a5b883a25015b89b4a12c0001  SUCCEEDED     Mon, 10 Apr 2017 13:00:00 UTC  Mon, 10 Apr 2017 13:00:00 UTC  Mon, 10 Apr 2017 13:00:01 UTC  202 - Cloud Controller Ac
```

View Logs

You can view logs for jobs by running `cf logs APP-NAME --recent`.

 : Scheduler jobs are executed as [CF Tasks](#).

See the following example:

```
$ cf logs my-app --recent
Connected, dumping recent logs for app my-app in org my-org / space my-space as user@example.com...
[...]
2017-04-19T23:04:13.79-0600 [APP/TASK/cc6fab7f-32a9-4404-4574-b0c430a96cd9 |- 0d30f4f0-11a4-4d6a-7e77-5e1cdc1aa5ec/0]OUT Creating container
2017-04-19T23:04:14.01-0600 [APP/TASK/cc6fab7f-32a9-4404-4574-b0c430a96cd9 |- 0d30f4f0-11a4-4d6a-7e77-5e1cdc1aa5ec/0]OUT Successfully created container
2017-04-19T23:04:14.22-0600 [APP/TASK/cc6fab7f-32a9-4404-4574-b0c430a96cd9 |- 0d30f4f0-11a4-4d6a-7e77-5e1cdc1aa5ec/0]OUT bin
2017-04-19T23:04:14.22-0600 [APP/TASK/cc6fab7f-32a9-4404-4574-b0c430a96cd9 |- 0d30f4f0-11a4-4d6a-7e77-5e1cdc1aa5ec/0]OUT db
2017-04-19T23:04:14.23-0600 [APP/TASK/cc6fab7f-32a9-4404-4574-b0c430a96cd9 |- 0d30f4f0-11a4-4d6a-7e77-5e1cdc1aa5ec/0]OUT Exit status 0
2017-04-19T23:04:14.24-0600 [APP/TASK/cc6fab7f-32a9-4404-4574-b0c430a96cd9 |- 0d30f4f0-11a4-4d6a-7e77-5e1cdc1aa5ec/0]OUT Destroying container
2017-04-19T23:04:14.55-0600 [APP/TASK/cc6fab7f-32a9-4404-4574-b0c430a96cd9 |- 0d30f4f0-11a4-4d6a-7e77-5e1cdc1aa5ec/0]OUT Successfully destroyed container
[...]
```

Schedule a Job

You can schedule a job to execute at any time using a schedule expression. Scheduler for PCF requires cron expressions in the `MIN HOUR DAY-OF-MONTH MONTH DAY-OF-WEEK` format.

For example, to execute a job at noon every day, run the following command:

```
$ cf schedule-job my-job "0 12 ? * *"
```

A single job can have multiple schedules. Each schedule has a GUID to help distinguish it from similar schedules.

View Schedules for Jobs

You can review schedules for all jobs in a space by running `cf job-schedules`. See the following example:

```
$ cf job-schedules my-job "0 12 ? * *"  
Getting scheduled jobs for org my-org / space my-space as user@example.com...  
App Name: my-app  
my-job      pwd  2b69e0c2-9664-46bb-4817-54afcedbb65d 0 12 ? * *  
OK
```

Delete a Job Schedule

You can delete a specific schedule by running `cf delete-job-schedule SCHEDULE_GUID`, where `SCHEDULE_GUID` is the GUID found in the output of the `job-schedules` command. See the following example:

```
$ cf delete-job-schedule 2b69e0c2-9664-46bb-4817-54afcedbb65d  
Really delete the schedule 2b69e0c2-9664-46bb-4817-54afcedbb65d / 0 12 ? * * and all associated history?> [yN]: y  
OK
```

Delete a Job

You can delete a job by running `cf delete-job JOB_NAME`. See the following example:

```
$ cf delete-job my-job  
Really delete the job my-job with command pwd and all associated schedules and history?> [yN]:y  
OK
```

Manage Jobs Using the Scheduler API

For information about managing jobs through the Scheduler HTTP API, see the [Scheduler for PCF API Documentation](#).

Release Notes



The Scheduler for Pivotal Cloud Foundry (PCF) tile is currently in beta and is intended for evaluation and test purposes only.

Do not use this product in a PCF production environment.

This topic contains release notes for Scheduler for Pivotal Cloud Foundry (PCF).

v1.0.4

August 11, 2017

Features included in this release:

- Create jobs.
- Run jobs.
- Schedule jobs.
- List job history.

Changes in this release:

- Scheduler for PCF supports floating stemcells in 3363 version line.

Known issues in this release:

- Users must have a `p-mysql` service plan named `1gb` with at least 1,000 MB of disk space.
- Scheduler for PCF provides only one service plan, `standard`, which allows you to schedule as many tasks as needed and at any interval.

v1.0.2

April 19, 2017

Features included in this release:

- Create jobs.
- Run jobs.
- Schedule jobs.
- List job history.

Known issues in this release:

- Users must have a `p-mysql` service plan named `1gb` with at least 1,000 MB of disk space.
- Scheduler for PCF provides only one service plan, `standard`, which allows you to schedule as many tasks as needed and at any interval.
- If you make the system domain in the `system` org private, Scheduler for PCF does not run.