Scheduler for PCF

Documentation®

Version 1.1

Published: 12 Dec 2018

Rev: 01

© 2018 Pivotal Software, Inc.
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table of Contents</td>
<td>2</td>
</tr>
<tr>
<td>Scheduler for PCF</td>
<td>3</td>
</tr>
<tr>
<td>Release Notes</td>
<td>4</td>
</tr>
<tr>
<td>Architecture</td>
<td>6</td>
</tr>
<tr>
<td>Installing and Configuring Scheduler for PCF</td>
<td>7</td>
</tr>
<tr>
<td>Monitoring Scheduler for PCF</td>
<td>9</td>
</tr>
<tr>
<td>Using Scheduler for PCF</td>
<td>10</td>
</tr>
<tr>
<td>Scheduling Jobs</td>
<td>12</td>
</tr>
<tr>
<td>Scheduling Calls</td>
<td>15</td>
</tr>
</tbody>
</table>
Scheduler for PCF

This documentation describes Scheduler for Pivotal Cloud Foundry (PCF).

Overview

Scheduler for PCF is a service for scheduling the execution of Diego tasks, such as database migrations, emails, or batch jobs, as well as the execution of outbound HTTP calls.

Scheduler for PCF enables developers to do the following:

- Create, run, and schedule jobs and view job history.
- Create, run, and schedule calls and view call history.

You can interact with the service through the Cloud Foundry Command Line Interface (cf CLI), Apps Manager, and the Scheduler HTTP API.

Product Snapshot

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

<table>
<thead>
<tr>
<th>Element</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>v1.1.2</td>
</tr>
<tr>
<td>Release date</td>
<td>December 21, 2017</td>
</tr>
<tr>
<td>Compatible Ops Manager version(s)</td>
<td>v1.11.x, v1.12.x, v2.0.x</td>
</tr>
<tr>
<td>Compatible Elastic Runtime version(s)</td>
<td>v1.11.x, v1.12.x</td>
</tr>
<tr>
<td>Compatible Pivotal Application Service version(s)</td>
<td>v2.0.x</td>
</tr>
<tr>
<td>IaaS support</td>
<td>AWS, Azure, GCP, OpenStack, and vSphere</td>
</tr>
</tbody>
</table>

* As of PCF v2.0, Elastic Runtime is renamed to Pivotal Application Service (PAS).

Requirements

Scheduler for PCF has the following requirements:

- MySQL for PCF v1.10. 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.
Release Notes
This topic contains release notes for Scheduler for Pivotal Cloud Foundry (PCF).

v1.1.4
Release Date: December 6, 2018

Features included in this release:
- Update OpenJDK
- Update MariaDB Connector

v1.1.2
Release Date: December 21, 2017

Features included in this release:
- Updates the stemcell
- Fixes bugs

v1.1.0
Release Date: October 16, 2017

Features included in this release:
- Users can do the following:
  - Create jobs
  - Run jobs
  - Schedule jobs
  - List job history
  - Create calls
  - Run calls
  - Schedule calls
  - List call history
- Internal credentials are stored in BOSH CredHub. If you want to access these credentials, you must use the CredHub CLI or the Ops Manager API instead of the Credentials tab of the Scheduler tile. For instructions on how to retrieve Scheduler credentials, see Retrieving Credentials from Your Deployment.

Changes in this release:
- Scheduler for PCF supports all available MySQL for PCF plans.
- HTTP calls can be disabled in the Scheduler for PCF tile.

Known issues in this release:
- Scheduler for PCF provides only one service plan, standard, which allows you to schedule as many tasks and calls as needed and at any interval.
v1.0.4
Release Date: 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
Release Date: 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 `sysadm` org private, Scheduler for PCF does not run.
Architecture

This topic describes the architecture of Scheduler for Pivotal Cloud Foundry (PCF).

HA Topology

In a highly available resource configuration, the Scheduler for PCF service uses the following:

- Three Scheduler instances, each running the Scheduler HTTP API server and a Scheduler Engine.
- Three Scheduler Service Broker instances

You can reduce the number of Scheduler instances to one using the Cloud Foundry Command Line Interface (cf CLI).

Data Persistence

Scheduler relies on a MySQL datastore to persist data, including jobs, calls, and history. Each call or schedule can require up to 10 MB of database capacity to store history.
Installing and Configuring Scheduler for PCF

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

Prerequisites

Before you install the Scheduler for PCF tile, you need to configure a MySQL for PCF service plan. The minimum resource configuration of the MySQL plan requires the following:

- Service plan name: 1gb
- Storage quota: 1,000 MB
- Concurrent connections quota: 40

For more information, see Add a Plan in the MySQL for PCF documentation.

Download and Install Scheduler for PCF

1. Download the product file from Pivotal Network.

2. Navigate to the Ops Manager Installation Dashboard and click Import a Product to upload the product file.

3. Under the Import a Product button, click + 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.

Configure Scheduler for PCF

Follow the steps below to configure the Scheduler for PCF tile.

Configure AZs and Networks

Follow the steps below to choose an Availability Zone (AZ) to run the Scheduler Service Broker and to select networks.

1. Click Assign AZs and Networks.

2. Configure the fields as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Place singleton in</td>
<td>Select the AZ for executing the Scheduler Errands. All Scheduler components execute as apps and do not require VMs.</td>
</tr>
<tr>
<td>Balance other jobs in</td>
<td>Ignore this field.</td>
</tr>
<tr>
<td>Network</td>
<td>Select a subnet for the Scheduler Errands. Use the subnet that includes the Elastic Runtime component VMs.</td>
</tr>
</tbody>
</table>

NOTE: The network selected is used only by Errand VMs and does not apply to the Scheduler runtime.

3. Click Save.

Configure Scheduler Options

1. Click Scheduler Configuration.

2. Configure the fields as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable outbound HTTP calls</td>
<td>The field is enabled by default. Disable this feature if you want to prevent users from scheduling outbound HTTP calls from the Scheduler for PCF service.</td>
</tr>
</tbody>
</table>
3. Click Save.

Verify Resource Config

1. Click Resource Config.
2. Verify the settings.
3. Click Save.

Verify Stemcell Version

1. Click Stemcell.
2. Verify the settings. If you need to import a new stemcell version, see the Download Stemcell section for your IaaS: AWS, Azure, GCP, or vsphere.
3. Click Save.
4. Return to the Ops Manager Installation Dashboard and click Apply Changes.
Monitoring Scheduler for PCF

This topic describes the logs and metrics you can use to monitor the health and performance of Scheduler for Pivotal Cloud Foundry (PCF). For general information about logging and metrics in PCF, see Logging and Metrics of.

Scheduler Metrics

Scheduler for PCF emits metrics to Loggregator if the metrics-forwarder service is bound to the scheduler app in the system org and p-scheduler space.

Scheduler for PCF emits the following metrics:

<table>
<thead>
<tr>
<th>Data Source</th>
<th>Description</th>
<th>Metric Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>gauge.scheduler.jobs.executed</code></td>
<td>The number of jobs Scheduler has executed.</td>
<td>Integer</td>
</tr>
<tr>
<td><code>gauge.scheduler.jobs.failed</code></td>
<td>The number of jobs Scheduler failed to execute.</td>
<td>Integer</td>
</tr>
<tr>
<td><code>gauge.scheduler.jobs.rejected</code></td>
<td>The number of jobs Scheduler requested the Cloud Controller to execute but failed to start a task.</td>
<td>Integer</td>
</tr>
<tr>
<td><code>gauge.scheduler.jobs.total</code></td>
<td>The number of jobs that currently exists in Scheduler.</td>
<td>Integer</td>
</tr>
<tr>
<td><code>gauge.scheduler.calls.executed</code></td>
<td>The number of calls Scheduler has executed.</td>
<td>Integer</td>
</tr>
<tr>
<td><code>gauge.scheduler.calls.failed</code></td>
<td>The number of calls Scheduler failed to execute.</td>
<td>Integer</td>
</tr>
<tr>
<td><code>gauge.scheduler.calls.total</code></td>
<td>The number of calls that currently exists in Scheduler.</td>
<td>Integer</td>
</tr>
</tbody>
</table>
Using Scheduler for PCF

This topic provides instructions for using Scheduler for Pivotal Cloud Foundry (PCF).

You can interact with the service through the Cloud Foundry Command Line Interface (cf CLI), Apps Manager, and the Scheduler HTTP API to configure jobs and outbound HTTP calls and to review history. 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.
- (Optional) The cf CLI v6.23.0 or greater and the Scheduler for PCF CLI plugin installed on your local machine. The Scheduler for PCF CLI plugin is packaged with the Scheduler for PCF tile on Pivotal Network.

Create and Bind a Service Instance Using the cf CLI

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       plan      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 and Calls

For information about the CLI operations that you can perform to manage jobs and calls in Scheduler for PCF, see Using Jobs and Using Calls.

If you want to manage jobs and calls through the Scheduler HTTP API, see the Scheduler for PCF API Documentation.

Using Scheduler for PCF in Apps Manager

For information about binding Scheduler for PCF to your app and scheduling tasks through Apps Manager, see Managing Apps and Service Instances Using Apps Manager.
Scheduling Jobs

This topic provides instructions for managing jobs in Scheduler for Pivotal Cloud Foundry (PCF).

Manage Jobs

You can use Scheduler for PCF to schedule execution of tasks on PCF, including database migrations, emails, and batch jobs. See the following sections to learn more about creating, running, and scheduling jobs and viewing job history.

Note: If you want to use the Cloud Foundry Command Line Interface (cf CLI) for managing jobs, you must install the Scheduler for PCF CLI plugin on your local machine. This plugin is packaged with the Scheduler for PCF tile on Pivotal Network cf. For more information, see the Prerequisites section of the Using Scheduler for PCF topic.

Create a Job

To execute a task related to an app, create a job by running the `cf create-job APP-NAME JOB-NAME COMMAND` command, where:

• **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 want to execute.

See the following example:

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

Executing job my-job for app 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

Execute a Job

You can execute a job manually. This is often useful to test the 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

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 distinguish it from similar schedules.
View Jobs

You can 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       pvd
OK
```

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
Getting scheduled jobs for org my-org / space my-space as user@example.com...
App Name: my-app
my-job     pvd 2b569c0c-9664-460b-4817-54afcd8bb65d 0 12 ? * *
OK
```

View Job History

You can view job history by running `cf job-history JOB-NAME`. See the following example:

```
$ cf job-history my-job
Getting scheduled job history for job in org my-org / space my-space as user@example.com...
1 / of 1 Total Results
Execution GUID  Execution Status  Scheduled Time  Execution Start Time  Execution End Time  Exit Message
a7e065a6b583250f15b896b12@000  SUCCEEDED  Mon, 10 Apr 2017 13:00:00 UTC  Mon, 10 Apr 2017 13:00:00 UTC  Mon, 10 Apr 2017 13:00:00 UTC  202 - Cloud Controller Agent
```

View Logs

You can view logs for jobs by running `cf logs APP-NAME`. See the following example:

```
$ cf logs my-app
Connected, dumping recent logs for app my-app in org my-org / space my-space as user@example.com...
[---]
2017-04-19T23:04:13.79-0700 [APP/Task/66f97732-9440-4574-8c56e15e6e9c] ~: ~: ~: /bin:114-466a-7e77-5c4ed1a5e0c8/OUT Creating container
2017-04-19T23:04:13.79-0700 [APP/Task/66f97732-9440-4574-8c56e15e6e9c] ~: ~: ~: /bin:114-466a-7e77-5c4ed1a5e0c8/OUT Successfully created container
2017-04-19T23:04:13.79-0700 [APP/Task/66f97732-9440-4574-8c56e15e6e9c] ~: ~: ~: /bin:114-466a-7e77-5c4ed1a5e0c8/OUT Running
2017-04-19T23:04:13.79-0700 [APP/Task/66f97732-9440-4574-8c56e15e6e9c] ~: ~: ~: /bin:114-466a-7e77-5c4ed1a5e0c8/OUT Exit status 0
2017-04-19T23:04:13.79-0700 [APP/Task/66f97732-9440-4574-8c56e15e6e9c] ~: ~: ~: /bin:114-466a-7e77-5c4ed1a5e0c8/OUT Successfully destroyed container
```

Delete a Job

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

```
$ cf delete-job my-job
```

© Copyright Pivotal Software Inc, 2013-2018
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 `cf job-schedules` command. See the following example:

```
$ cf delete-job-schedule 2b69e0c3-9664-460b-4817-54a0cedb6e65d
Really delete the schedule 2b69e0c3-9664-460b-4817-54a0cedb6e65d / 0 12 * * * and all associated history? [yn]: y
OK
```
Scheduling Calls

This topic provides instructions for managing outbound HTTP calls in Scheduler for Pivotal Cloud Foundry (PCF).

Manage Calls

You can use Scheduler for PCF to schedule execution of HTTP calls to external HTTP services. See the following sections to learn more about creating, running, and scheduling calls and viewing call history.

Note: If you want to use the Cloud Foundry Command Line Interface (cf CLI) for managing calls, you must install the Scheduler for PCF CLI plugin on your local machine. This plugin is packaged with the Scheduler for PCF tile on Pivotal Network cf. For more information, see the Prerequisites section of the Using Scheduler for PCF topic.

Create a Call

You can create a call by running the `cf create-call APP-NAME CALL-NAME URL` command, where:

- `APP-NAME` is the app you want to create a call for.
- `CALL-NAME` is the name for your call.
- `URL` is the URL to execute a HTTP POST call against.

Execute a Call

You can execute a call manually by running the `cf run-call CALL-NAME` command. This is often useful to test the configuration of a call prior to scheduling it for recurring execution.

See the following example:

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

Schedule a Call

You can schedule a call 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 call at noon every day, run the following command:

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

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

View Calls

You can use the cf CLI to list all calls in a space by running `cf calls`. See the following example:
View Schedules for Calls
You can review schedules for all calls in a space by running `cf call-schedules`. See the following example:

```
$ cf call-schedules
Getting scheduled calls for org my-org / space my-space as user@example.com...
App Name: my-app
my-call 2b69ec0c-9664-46bb-4817-54afcedb6b6d 0 12 * * OK
```

View Call History
You can review call history by running `cf call-history CALL_NAME`. See the following example:

```
$ cf call-history my-call
Getting scheduled call history for my-call in org my-org / space my-space as user@example.com...
1 - 1 of 1 Results
GUID Execution State Scheduled Time Execution Start Time Execution End Time Exit Message
```

Delete a Call
You can delete a call by running `cf delete-call CALL_NAME`. See the following example:

```
$ cf delete-call my-call
Really delete the call my-call with url https://example.com and all associated schedules and history? [y/N]: Y
OK
```

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

```
$ cf delete-call-schedule 2b69ec0c-9664-46bb-4817-54afcedb6b6d
Really delete the schedule 2b69ec0c-9664-46bb-4817-54afcedb6b6d / 0 12 * * and all associated history? [y/N]: Y
OK
```