|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 版本 | 修订者 | 修订描述 |
| 2018.11.28 | v1 | 林 清 | 初稿 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

[1. 介绍 2](#_Toc531184230)

[2. QuickStart 2](#_Toc531184231)

[2.1 安装Operator 2](#_Toc531184232)

[2.2 部署Spark任务 2](#_Toc531184233)

[3. 使用说明 3](#_Toc531184234)

[4. 工作原理分析 3](#_Toc531184235)

# 介绍

[Spark Operator](https://github.com/GoogleCloudPlatform/spark-on-k8s-operator)是GoogleCloudPlatform 官方GitHub上的一个开源项目，通过Spark Operator可以让用户在Kubernetes集群中快速部署Spark任务，并且提供基于prometheus的监控信息采集等功能。

Spark Operator基于Spark 2.3.x和Kubernetes 1.9以及以上版本开发，并在持续开发。

# QuickStart

## 安装Operator

下载下面的Helm包上传到Kubernetes环境上安装：



helm install sparkoperator -n spark-op --namespace spark-operator --set enableWebhook=false --set enableMetrics=false

PS：Spark-Operator 的Helm包可以在[Helm Chart官方](https://github.com/helm/charts)GitHub下载到，安装时需要Pull 镜像gcr.io/spark-operator/spark-operator:v2.3.1-v1alpha1-latest请搭梯子！

## 部署Spark任务

安装Spark-Operators之后，Kubernetes中会多出两个CRD（自定义资源）：



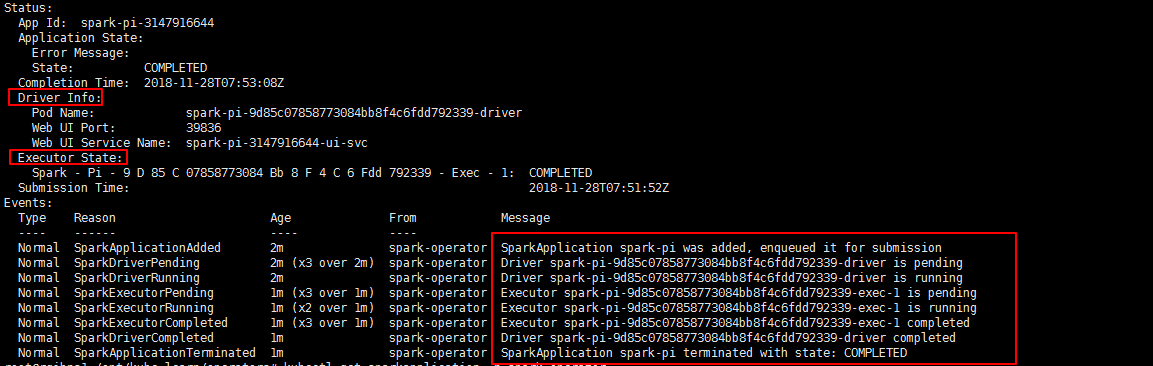
使用sparkapplications可以部署Spark任务，参考下面的Yaml文件部署spark-pi任务。



安装上面的yaml后，在spark-operator命名空间下会生成sparkapplication对象，该对象控制Spark 任务中Driver Pod和Executor 对应Pod的创建。



使用kubectl describe sparkapplication spark-pi -n spark-operator命令可以查看任务的调度状态，以及配置信息。



用户通过kubectl log命令可以查看driver和executor的运行日志。

# 使用说明

## API参考

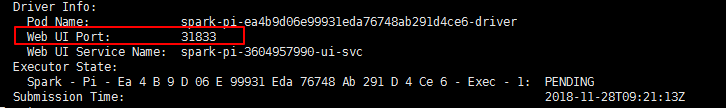
Spark Operator定义了SparkApplication和ScheduledSparkApplication两个CRD对象，前者代表一个Spark任务，后者代表定时调度的Spark任务。用户只要定义SparkApplication对象的Yaml文件，即定义了任务。官方给出[API详细说明](https://github.com/GoogleCloudPlatform/spark-on-k8s-operator/blob/master/docs/api.md)。

参考Spark-Pi的yaml文件定义：

**apiVersion:** "sparkoperator.k8s.io/v1alpha1"  
**kind:** SparkApplication  
**metadata:  
 name:** spark-pi  
 **namespace:** spark-operator  
**spec:  
 type:** Scala  
 **mode:** cluster  
 **image:** "gcr.io/spark-operator/spark:v2.3.1"  
 **imagePullPolicy:** IfNotPresent  
 **mainClass:** org.apache.spark.examples.SparkPi  
 **mainApplicationFile:** "local:///opt/spark/examples/jars/spark-examples\_2.11-2.3.1.jar"  
 **restartPolicy:** Never  
 **driver:  
 cores:** 0.1  
 **coreLimit:** "200m"  
 **memory:** "512m"  
 **serviceAccount:** spark-op-sparkoperator  
 **executor:  
 cores:** 1  
 **instances:** 1  
 **memory:** "512m"

## 访问WebUI

当用户提交SparkApplication后，Operator会自动创建一个NodePort。用户通过Kubernetes集群的任意节点都能访问到WebUI。端口号信息可以通过kubectl describe sparkapplication <app name>查询，如下图。



## 配置定时任务

通过ScheduledSparkApplication可以配置定时调度的spark任务。参考下面的例子，配置了一个5min调度的spark-pi任务。

**apiVersion:** "sparkoperator.k8s.io/v1alpha1"  
**kind:** ScheduledSparkApplication  
**metadata:  
 name:** spark-pi-scheduled  
 **namespace:** spark-operator  
**spec:  
 schedule:** "@every 5m"  
 **concurrencyPolicy:** Allow  
 **template:  
 type:** Scala  
 **mode:** cluster  
 **image:** "gcr.io/spark-operator/spark:v2.3.1"  
 **imagePullPolicy:** Always  
 **mainClass:** org.apache.spark.examples.SparkPi  
 **mainApplicationFile:** "local:///opt/spark/examples/jars/spark-examples\_2.11-2.3.1.jar"  
 **restartPolicy:** Never  
 **driver:  
 cores:** 0.1  
 **coreLimit:** "200m"  
 **memory:** "512m"  
 **serviceAccount:** spark-op-sparkoperator  
 **executor:  
 cores:** 1  
 **instances:** 1  
 **memory:** "512m"

## 制作业务镜像

用户可以基于Google提供的Spark镜像作为业务的基础镜像（只需将Jar包添加到Image中即可）。当Google提供spark 2.3.1和spark 2.4.0两个版本的基础镜像。

当前不太清楚Spark-Operator是否需要使用Google配套的Spark基础镜像（需要测试）。

## Metrics

## Mutating Admission Webhook

# 工作原理分析