

学习 > Cloud computing

概览

您需要做的准备工作

简介：两个考虑因素

Kubernetes StatefulSets

使用 IBM Cloud Kubernetes Service 部署 MongoDB 副本集

[了解 IBM Cloud Kubernetes Service 如何帮助中使用 StatefulSets](#) 和 Docker 容器完美结合使用

结束语



Peter Tuton

相关主题 2018 年 4 月 24 日发布 / 更新: 2018 年 5 月 30 日

评论

开发社区的一些人表达了对 Docker 容器和 Kubernetes 的担忧，尽管它们各自有着明显的优点。本文中，我将解释 Kubernetes 开发人员如何通过引入 **StatefulSets** 来应对这些担忧。在本教程中，我将在 Kubernetes 的 IBM Cloud Kubernetes Service 中部署一个 MongoDB 副本集。在此过程中，我将展示如何部署 Kubernetes StatefulSets。最后，我提供了一些关于在生产环境中使用这些工具

您需要做的准备工作

本文需要至少拥有以下工具的入门知识：

- [MongoDB](#)
- [Docker](#)
- [Kubernetes](#)
- 使用 [IBM Cloud Kubernetes Service](#) 创建一个 Kubernetes 集群
- 一个 [IBM Cloud 帐户](#)（至少应该是一个“即用即付”帐户；Lite 帐户无法使用 IBM Cloud File Storage 产品创建持久卷）

如果您对这些技术有更深入的了解，并且希望直接跳到部署指令部分，请转到“[在 IBM Cloud Kubernetes Service 上创建 MongoDB 副本集](#)”。

简介：两个考虑因素

如 Kubernetes) 来运行 MongoDB 的考虑因素。特别值得注意的是以下几点：

内容

- MongoDB 数据库节点是有状态的。

概览

- 一个副本集中的 MongoDB 数据库节点必须相互通信，包括在重新调度后（即创建新节点时

您需要做的准备工作

那些至少拥有 Docker 入门知识的人可能会认识到，第一个考虑因素与 Docker 容器的一条基本原则——“两个考虑因素”，而不是有状态。此外，试图向 Docker 容器添加状态很有挑战性。如果做的另一条设计原则——即“快速、一致地交付应用程序”。

在 IBM Cloud Kubernetes Service 中使用 StatefulSets

性。Kubernetes 的第五个考虑因素与容器网络相关，特别是主机名方面，因为容器的实例（“pod”）在用新主机名进行重建。因此，尽管 Kubernetes 的一些特性使其成为一个有吸引力的系统，可个性化的应用程序，但这可能会破坏容器间的通信。

相关主题

考虑到这些事实，许多架构师没有将 Docker 容器和 Kubernetes 视为提供 MongoDB 副本集的优势。为了解决这些问题，Kubernetes 社区引入了 **StatefulSets**。

Kubernetes StatefulSets

StatefulSets 是在 Kubernetes V1.5 中作为 beta 资源引入的，在 V1.9 中已稳定下来并广泛提及之类的有状态应用程序而设计的。

Kubernetes 文档表明：

就像部署一样，一个 StatefulSet 可以管理基于相同容器规格的 Pod。与部署不同的是，一个 StatefulSet 份。这些 pod 是根据同一种规格来创建的，但无法互换：每个 pod 都会在任何重新调度后保持着一

至关重要的是，StatefulSets 提供的（5 个）重要特性中的两个与通过 Docker 和 Kubernetes 考虑因素相关：

- 稳定、持久的存储
- 稳定、唯一的网络标识符

简言之，这意味着通过使用一个 MongoDB Docker 容器，可以在 Kubernetes 内定义一个能为附加持久存储的 StatefulSet 配置。无论新创建多少个节点，每个节点都可以使用一个唯一、无节点。

在 IBM Cloud Kubernetes Service 中使用 StatefulSets

至关重要。

内容

例如，许多应用程序都需要可预测的 IOPS。企业或政府要求还可能规定，应用程序存储需要在本地实现冗余（尽管 MongoDB 具有内在的复制特性）。

您需要做的准备工作

默认情况下，IBM Cloud Kubernetes Service 被配置为使用 IBM Cloud 的基于 NFS 的文件存储服务提供了 8 种预定义的存储类，所以集群管理员不需要创建任何这些类。

使用 StatefulSets 时，配置会根据应用程序或业务需求来选择一种现有的存储类；而且没有什么

在 IBM Cloud Kubernetes Service 中使用 StatefulSets

IBM Cloud File Storage 产品提供了众多优势，包括：

结束语

- 基于闪存的存储

相关主题

- 静止数据加密

评论

- 快照和副本

- 卷复制
- 可扩展的卷
- 可调节的 IOPS

备注：IBM Cloud Kubernetes Service 工作节点默认情况下使用加密的 SSD。

第 1 步.在 IBM Cloud Kubernetes Service 上创建 MongoDB 副本集

用于 IBM Cloud Kubernetes Service 的配置与其他 Kubernetes 提供者使用的配置几乎没有区别。选择 IBM Cloud Kubernetes Service 作为部署平台是众多原因之一。唯一的真正区别是 storageClassName 值。

备注：假设您已创建了一个 Container Service 集群。要获得关于如何创建 IBM Cloud Kubernetes Service 集群的信息，请参阅 IBM Cloud Kubernetes Service 文档中的“[设置集群](#)”。

备注：确保您的集群至少运行着 Kubernetes V1.9.2。为此，可以将 --kube-version 1.9.2 传递给部署脚本。有关更多信息，请参阅 [部署 Kubernetes 集群](#) 文档。

完成本节后，您会在 IBM Cloud Kubernetes Service 中运行的 Kubernetes 集群中创建一个 3 节点的副本集。IBM Cloud File Storage 产品来实现持久卷。

- 持久卷 (IBM Cloud File Storage)

图 1.3 节点 MongoDB 副本集的主要组件概览



备注：本文不会介绍 Edge Services、Security、LoadBalancer/Ingress Service 和 Application 集群中的详细信息，包括公开应用程序的网络选项，请参阅文档中的[“规划外部网络”](#)。

第 2 步.创建无头服务配置文件

Kubernetes 无头服务可以控制 pod 的域，不需要负载均衡器服务或像其他 Kubernetes 服务类与服务 selector 匹配的 pod 被分组到一起。

对于无头服务，请创建下面这个配置文件：

清单 1. mongo-headless-service.yaml

```

1  apiVersion: v1
2  kind: Service
3  metadata:
4
5
6
7  spec:
8    ports:
9      - name: mongo
10       port: 27017
11       targetPort: 27017
12     clusterIP: None
13     selector:
14       app: mongo

```

简介: 两个考虑因素

创建无头服务时将 clusterIP 值设置为 None (第 12 行)。

Kubernetes StatefulSets

确保对 targetPort 使用了合适的端口号 (第 11 行)。该端口号应该与下面第 3 步中的 StatefulSet (在 [IBM Cloud Kubernetes Service 中使用 StatefulSets](#) 行) 相匹配, 这是 mongod 服务使用的端口。默认情况下, 该端口为 27017。用于 port 参数指向连接该服务的端口。

相关主题

第 3 步.创建 StatefulSet 配置文件

对于 StatefulSet, 请创建下面这个配置文件:

清单 2. mongo-statefulset.yaml

```

1  apiVersion: apps/v1
2  kind: StatefulSet
3  metadata:
4    name: mongo
5  spec:
6    selector:
7      matchLabels:
8        app: mongo
9    serviceName: "mongo"
10   replicas: 3
11   podManagementPolicy: Parallel
12   template:
13     metadata:
14       labels:
15         app: mongo
16     spec:
17       terminationGracePeriodSeconds: 10
18       affinity:
19         podAntiAffinity:
20           requiredDuringSchedulingIgnoredDuringExecution:
21             - labelSelector:
22                 matchExpressions:
23                   - key: "app"
24                     operator: In
25                     values:
26                       - mongo
27             topologyKey: "kubernetes.io/hostname"
28   containers:

```

```

29     - name: mongo
30       image: mongo
31       command:
32
33
34
35     - rs0
36       ports:
37         - containerPort: 27017
38       volumeMounts:
39         - name: mongo-data
40           mountPath: /data/db
41     volumeClaimTemplates:
42     - metadata:
43         name: mongo-data
44       spec:
45         accessModes: [ "ReadWriteOnce" ]
46         storageClassName: ibmc-file-bronze
47       resources:
48         requests:
49           storage: 20Gi

```

结束语

备注： Affinity 部分（第 18-27 行）可以确保没有安排某个 MongoDB 副本集 pod 在已运行所有 3 个节点都在单个工作节点上运行而工作节点发生故障，则这么做会引入潜在的故障点。

相关主题

创建配置后，是时候部署资源了。

第 4 步.部署该服务和 StatefulSet

要部署无头服务和 StatefulSet，只需使用 IBM Cloud CLI、Bluemix 或 bx，并使用 Kubernetes 命令。

1. 首先，登录到 IBM Cloud CLI:

```
$ bx login
```

2. 然后，在 CLI 中设置集群的上下文:

```
$ $(bx cs cluster-config sandbox-cluster | grep KUBECONFIG)
```

3. 接下来，执行以下命令（第 1 和第 4 行）。预期输出如下所示。

```

1 $ kubectl apply -f mongo-headless-service.yaml
2 service "mongo" created
3
4 $ kubectl apply -f mongo-statefulset.yaml
5 statefulset "mongo" created

```

4. 要进行确认，请使用以下命令（第 1 和第 5 行）获取每个对象。每个对象的预期输出如下所示。

```

1 $ kubectl get service mongo
2 NAME      TYPE          CLUSTER-IP    EXTERNAL-IP    PORT(S)        AGE
3 mongo     ClusterIP     None          <none>         27017/TCP      5s
4
5 $ kubectl get statefulset mongo
6 NAME      DESIRED    CURRENT    AGE
7 mongo     3          3          1m

```

Kub

在 IBM Cloud Kubernetes Service 中使用 StatefulSets

5. 部署 StatefulSet 后，Kubernetes 负责创建 pod，根据规范为 pod 命名（例如 mongo-0、mongo-1、mongo-2）。创建的 pod，请使用 pod 标签 app=mongo 来 get pod 对象，如下所示：

```

1 $ kubectl get pod -l app=mongo
2 NAME      READY    STATUS    RESTARTS    AGE
3 mongo-0   1/1      Running   0            30s
4 mongo-1   1/1      Running   0            30s
5 mongo-2   1/1      Running   0            30s

```

6. 此外，Kubernetes 负责创建持久卷声明 (pvc) 并将持久卷绑定到 pod。要列出持久卷声明，如下所示：

```

1 $ kubectl get pvc
2 NAME                                STATUS    VOLUME
3 mongo-data-mongo-0                  Bound    pvc-3ed73cf3-0940-11e8-ac45-925e6fdab1e7
4 mongo-data-mongo-1                  Bound    pvc-3ed82f17-0940-11e8-ac45-925e6fdab1e7
5 mongo-data-mongo-2                  Bound    pvc-3ed90d7e-0940-11e8-ac45-925e6fdab1e7

```

每个 MongoDB 副本集节点现在都已创建为可以（在内部）通过 DNS 寻址的 Kubernetes pod。

具体来讲，每个 pod 的内部 Kubernetes DNS 主机名都统一使用 StatefulSet 名称与附加的 \$(statefulset name)-\$(ordinal)

此外，通过使用无头服务，域管理的服务具有以下格式：

\$(servicename).\$(namespace).svc.cluster.local

— 目录 — 快速入门 — 部署副本集 — 配置副本集 — 启动副本集 — 验证副本集 — 故障排除

ψ (podname), ψ (service-name)

内容

概览 例如，使用所提供的配置时，可以使用以下 DNS 名称来引用每个 MongoDB 节点：
mongo-0.mongo, mongo-1.mongo, mongo-2.mongo
您需

简介：两个考虑因素

持久存储被配置为可通过每个 pod 内的 /data/db 挂载路径进行访问，并在默认情况下使用 IEFS。如果您删除了副本集，销毁了一个 pod，任何未来的同名 pod 会自动挂载到相同的持久存储上（除非该持久存储声明在 [IBM Cloud Kubernetes Service 中使用 StatefulSets](#)）。

现在只剩下使用每个节点来初始化副本集了。
结束语

相关主题

第 5 步.启动并配置 MongoDB 副本集
评论

创建每个 MongoDB 副本集节点后，副本集本身需要启动和配置。为此，访问第一个节点并运行以下命令：

备注：可以将这些命令编写为 CI/CD 流程的一部分，也可以增加或删除新节点。

1. 首先，在第一个副本集节点上执行 MongoDB shell：

```
$ kubectl exec -it mongo-0 -- mongo
```

2. 现在，启动 MongoDB 副本集：

```
> rs.initiate()
```

3. 为副本集的第一个成员重新配置正确的 DNS 名称：

```
> var cfg = rs.conf();cfg.members[0].host="mongo-0.mongo:27017";rs.reconfig(cfg)
```

4. 添加剩余的副本集节点：

```
1 | rs.add("mongo-1.mongo:27017")
```



```
2 | rs.add("mongo-2.mongo:27017")
```

```
> rs.status()
```

内容

概览

新配置会显示 3 个成员，它们的主机名有一个对应的“_id”值。

您需要做的准备工作

MongoDB 副本集现已启动和配置。此刻，MongoDB 副本集已基本准备好供在同一个 Kubernetes StatefulSets 应用程序访问。

Kubernetes StatefulSets

备注：要从 Kubernetes 集群外访问 MongoDB 副本集，需要一个 Kubernetes “NodePort”或“LoadBalancer”类型的 Service。在 [IBM Cloud Kubernetes Service 中使用 StatefulSets](#) 文档中，需要关于如何规划外部网络的更多细节，请参阅 [IBM Cloud Kubernetes Service 文档](#)。

结束语

使用本文中提供的配置时，MongoDB 副本集 URI 连接字符串为：

相关主题

mongodb://mongo-0.mongo:27017,mongo-1.mongo:27017,mongo-2.mongo/myproject?replicaSe

生产考虑因素

应该将本文视为入门指南。在生产中使用 Kubernetes 运行 MongoDB 时，有更多考虑因素，其介绍过。未来的文章将提供特定于 IBM Cloud Kubernetes Service 产品的操作说明。

例如，建议您在生产过程中使用所提供的配置时考虑以下情况（参见下面的“相关主题”了解这

- 身份验证 (mongoDb)
- 不以根用户身份运行
- 使用 MongoDB Enterprise Edition (mongoDb)
- 对集群分区 (mongoDb)
- 解决大量的成员 (mongoDb)
- 管理计算资源 (Kubernetes)
- 使用 IBM Container Registry (IBM Cloud) 在一个高度可用的、可扩展的架构中安全地存储

结束语

本文演示了在基于 Kubernetes 的 IBM Cloud Kubernetes Service 中如何部署一个 MongoDB 副本集。

此外，我还谈到了使用 IBM Cloud Kubernetes Service 的优势，特别是重点介绍了支撑该服务内容。

最后，按照本文所提供的操作说明，您可以在几分钟内在 IBM Cloud Kubernetes Service 上创建和测试环境奠定基础。在考虑更多因素之后，也可以实现一个生产环境。
您需要做的准备工作

简介：两个考虑因素

相关主题

Kubernetes StatefulSets

[创建一个 IBM Cloud 帐户](#)
[在 IBM Cloud Kubernetes Service 中使用 StatefulSets](#)

[IBM Cloud Kubernetes Service 文档](#)

结束语

[MongoDB 文档](#)

相关主题

[Kubernetes 文档](#)

评论

[Docker 文档](#)

[“通过 Docker 和 Kubernetes 以微服务形式运行 MongoDB” \(Andrew Morgan\)](#)

评论

添加或订阅评论，请先[登录](#)或[注册](#)。

☐ 有新评论时提醒我

developerWorks

站点反馈

我要投稿

投稿指南

报告滥用

第三方提示

关注微博

[加入](#)[ISV 资源 \(英语\)](#)

选择语言

[English](#)[中文](#)[日本語](#)[Русский](#)[Português \(Brasil\)](#)[Español](#)[한글](#)[技术文档库](#)[dW 中国时事通讯](#)[博客](#)[活动](#)[社区](#)[开发者中心](#)[视频](#)[订阅源](#)[软件下载](#)[Code patterns](#)

[联系 IBM](#) [隐私条约](#) [使用条款](#) [信息无障碍选项](#) [反馈](#) [Cookie 首选项](#)

内容

概览

您需要做的准备工作

简介：两个考虑因素

Kubernetes StatefulSets

在 IBM Cloud Kubernetes Service 中使用 StatefulSets

结束语

相关主题

评论