本小节将展示如何使用[Mesosphere DC/OS](https://mesosphere.com/)工具，来编排部署和管理一个由3个节点构成的以非安全模式启动的CockroachDB集群。

**WARNING:** 在实际生产业务中不推荐使用非安全模式的集群，后期若能以安全模式部署，我们将会更新此页面。

前言

需求

* 集群至少包含3个私有节点
* 若使用企业版的DC/OS，在安装CockroachDB之前使用superuser权限需要[配置一个服务账户](https://docs.mesosphere.com/1.9/security/ent/service-auth/custom-service-auth/)

| **安全模式** | **安全账户** |
| --- | --- |
| strict | 需要 |
| permissive | 可选 |
| disabled | 不需要 |

限制

CockroachDB在DC/OS中运行除了以下限制以外，跟其他环境一致：

* cockroachdbDC/OS 服务目前只在1.9和1.10版本上做过测试。
* 暂不支持在安全模式下运行
* 暂不支持在多数据中心部署集群
* 暂不支持移除集群中的节点
* 首次部署后不支持再次修改卷类型和卷大小
* 暂不支持机架放置和感知

Step 1: 安全和启动DC/OS

启动和运行DC/OS的最快方法是[在AWS CloudFormation上使用开源DC/OS模板](http://doc.cockroachchina.baidu.com/" \l "deploy/orchestrated-deployment/mesosphere-dc-os-deployment/open%20source%20DC/OS%20template%20on%20AWS%20CloudFormation)。此外，参照官方文档还可以使用其他开源或企业版的DC/OS安装方法：

* [Open Source DC/OS](https://docs.mesosphere.com/1.10/installing/oss/)
* [Enterprise DC/OS](https://docs.mesosphere.com/1.10/installing/ent/)

使用AWS CloudFormation，安装过程将耗时10-15分钟。在CloudFormation界面显示CREATE\_COMPLETE状态以后，用户需要[启动DC/OS](https://docs.mesosphere.com/1.10/installing/oss/cloud/aws/basic/" \l "launch-dcos)，并安装[DC/OS CLI](https://docs.mesosphere.com/1.10/cli/install/)。

Step 2: 启动CockroachDB

获取[CockroachDB配置文件](https://github.com/cockroachdb/dcos-cockroachdb-service" \l "node-settings)

按需调整。

dcos package describe --config cockroachdb

该配置文件默认创建了3个节点构成的CockroachDB集群，用户需要根据业务场景进行调整（可参考[Recommended Production Settings](http://doc.cockroachchina.baidu.com/" \l "deploy/production-checklist/)），新配置文件重命名为cockroach.json。

启动CockroachDB集群作为DC/OS服务

* 使用默认配置：

dcos package install cockroachdb

* 使用自定义cockroach.json配置：

dcos package install cockroachdb --options=cockroach.json

在DC/OS用户界面的**Services**标签页监控集群部署状况

**NOTE:** 用户也可以[在DC/OS用户界面安装CockroachDB](https://docs.mesosphere.com/1.10/deploying-services/install/" \l "installing-a-service-using-the-gui)

Step 3: 测试集群

获取CockroachDB集群的端点

dcos cockroachdb endpoints pg

{

"address": [

"10.0.0.212:26257",

"10.0.2.57:26257",

"10.0.3.81:26257"

],

"dns": [

"cockroachdb-0-node-init.cockroachdb.autoip.dcos.thisdcos.directory:26257",

"cockroachdb-1-node-join.cockroachdb.autoip.dcos.thisdcos.directory:26257",

"cockroachdb-2-node-join.cockroachdb.autoip.dcos.thisdcos.directory:26257"

],

"vip": "pg.cockroachdb.l4lb.thisdcos.directory:26257"

}

该端点包括：

* .mesos DC/OS集群中的每个机器的主机名。
* 如果.mesos提供的主机名无法解析，则需要每个机器的直接IP地址
* 一个vip地址，即一个允许HA的主机名，用于访问任何实例机器。在以下某些步骤中将会使用vip地址。

通常，.mesos端点只在同一个DC / OS集群中有效。 从集群外部，用户可以使用直接IP地址或设置作为CockroachDB实例机器网关的代理服务。 出于开发和测试目的，用户可以使用[DC/OS隧道](https://docs.mesosphere.com/1.10/developing-services/tunnel/)访问群集外部的服务，但此选项不适合生产系统使用。详见内容参考[监控集群](https://www.cockroachlabs.com/docs/stable/orchestrate-cockroachdb-with-mesosphere-insecure.html" \l "step-4-monitor-the-cluster)

SSH登陆DC/OS主节点

dcos node ssh --master-proxy --leader

启动临时容器运行SQL客户端

--host取值于vip端点：

docker run -it cockroachdb/cockroach:v2.0.5 sql --insecure --host=pg.cockroachdb.l4lb.thisdcos.directory

*# Welcome to the cockroach SQL interface.*

*# All statements must be terminated by a semicolon.*

*# To exit: CTRL + D.*

root@pg.cockroachdb.l4lb.thisdcos.directory:26257/>

执行SQL

**CREATE** **DATABASE** bank;

**CREATE** **TABLE** bank.accounts (id INT **PRIMARY** **KEY**, balance DECIMAL);

**INSERT** **INTO** bank.accounts **VALUES** (1, 1000.50);

**SELECT** \* **FROM** bank.accounts;

+*----+---------+*

| id | balance |

+*----+---------+*

| 1 | 1000.5 |

+*----+---------+*

(1 row)

退出SQL shell并删除临时pod。

Step 4: 监控集群

使用DC/OS隧道运行HTTP代理，并用于访问集群的Admin界面。 - 安装DC/OS隧道包

dcos package install tunnel-cli --cli

* 启动DC/OS隧道

sudo dcos tunnel http

* 访问https://www.cockroachlabs.com/docs/stable/http.cockroachdb.l4lb.thisdcos.directory.mydcos.directory

Step 5: 拓展集群

默认的cockroachdb服务创建一个3个节点构成的CockroachDB集群。通过更新调度程序进程，可以在启动服务后向集群添加节点：

* 在DC/OS用户界面，切换到**Services**表单
* 选择**cockroachdb**服务
* 点击**Edit**.
* 选择**Environment**
* 更新NODE\_COUNT变量，对应集群需要调整到的CockroachDB节点数量。
* 点击**Review & Run** ，再**Run Service**

调度进程将使用新配置重新启动，并验证检测到的配置更改。 返回Admin界面，查看**Node List**，检查新增节点是否成功添加到群集中。

此外，用户可以[SSH登陆到DC/OS的主节点](https://docs.mesosphere.com/1.10/administering-clusters/sshcluster/)，执行[cockroach node status](https://www.cockroachlabs.com/docs/stable/view-node-details.html)命令创建临时容器，--host取值于vip端点。

dcos node ssh --master-proxy --leader

docker run -it cockroachdb/cockroach:v2.0.5 node status --insecure --host=pg.cockroachdb.l4lb.thisdcos.directory

+----+--------------------------------------------------------------------------+--------+---------------------+---------------------+------------+-----------+-------------+--------------+--------------+------------------+-----------------------+--------+--------------------+------------------------+

| id | address | build | updated\_at | started\_at | live\_bytes | key\_bytes | value\_bytes | intent\_bytes | system\_bytes | replicas\_leaders | replicas\_leaseholders | ranges | ranges\_unavailable | ranges\_underreplicated |

+----+--------------------------------------------------------------------------+--------+---------------------+---------------------+------------+-----------+-------------+--------------+--------------+------------------+-----------------------+--------+--------------------+------------------------+

| 1 | cockroachdb-0-node-init.cockroachdb.autoip.dcos.thisdcos.directory:26257 | v1.1.3 | 2017-12-11 20:59:12 | 2017-12-11 19:14:42 | 41183973 | 1769 | 41187432 | 0 | 6018 | 2 | 2 | 2 | 0 | 0 |

| 2 | cockroachdb-1-node-join.cockroachdb.autoip.dcos.thisdcos.directory:26257 | v1.1.3 | 2017-12-11 20:59:12 | 2017-12-11 19:14:52 | 115448 | 71037 | 209282 | 0 | 6218 | 4 | 4 | 4 | 0 | 0 |

| 3 | cockroachdb-2-node-join.cockroachdb.autoip.dcos.thisdcos.directory:26257 | v1.1.3 | 2017-12-11 20:59:03 | 2017-12-11 19:14:53 | 120325 | 72652 | 217422 | 0 | 6732 | 4 | 3 | 4 | 0 | 0 |

| 4 | cockroachdb-3-node-join.cockroachdb.autoip.dcos.thisdcos.directory:26257 | v1.1.3 | 2017-12-11 20:59:03 | 2017-12-11 20:21:43 | 41248030 | 79147 | 41338632 | 0 | 6569 | 1 | 1 | 1 | 0 | 0 |

| 5 | cockroachdb-4-node-join.cockroachdb.autoip.dcos.thisdcos.directory:26257 | v1.1.3 | 2017-12-11 20:59:04 | 2017-12-11 20:56:54 | 41211967 | 30550 | 41181417 | 0 | 6854 | 1 | 1 | 1 | 0 | 0 |

+----+--------------------------------------------------------------------------+--------+---------------------+---------------------+------------+-----------+-------------+--------------+--------------+------------------+-----------------------+--------+--------------------+------------------------+

(5 rows)

Step 6: 集群维护

更新配置

除了添加节点，用户还可以[修改节点的CPU和内存要求](https://github.com/cockroachdb/dcos-cockroachdb-service" \l "resizing-a-node)，更新[放置约束](https://github.com/cockroachdb/dcos-cockroachdb-service" \l "resizing-a-node)以及更改其他[服务设置](https://github.com/cockroachdb/dcos-cockroachdb-service" \l "service-settings)。参照**Step 5**操作，需要更新的环境变量。

* 进行更新操作后，调度程序将重新启动并自动将检测到的任何变更依次部署到服务。例如，修改的变更将首先应用于cockroachdb-0，然后应用于cockroachdb-1，依此类推。
* 节点配置有“准备检查（Readiness check）”，以确保在将指定变更应用于更新序列中的下一个节点之前，基础服务基本处于正常状态。 但是，这种基本检查并非万无一失，应采取合理措施确保给定的配置更改不会对服务行为产生负面影响。

重启节点

用户可以在同一位置重新启动节点，使用同样的持久卷数据。区别于系统重新启动进程，该操作不会删除不在持久卷上的任何数据。

* 获取重启的节点的pod名称：

dcos cockroachdb pod list

[

"cockroachdb-0",

"cockroachdb-1",

"cockroachdb-2",

"cockroachdb-3",

"cockroachdb-4",

"metrics-0"

]

* 重启相关的pod

dcos cockroachdb pods restart cockroachdb-<NUM>

替换节点

用户可以将节点移动到新系统，并丢弃先前系统使用的持久卷，以便在新系统中重建。节点不会自动移动，因此必须在合适时机执行节点移动，例如在系统脱机或系统已脱机之前。

* 获取重启节点的pod名称：

dcos cockroachdb pod list

[

"cockroachdb-0",

"cockroachdb-1",

"cockroachdb-2",

"cockroachdb-3",

"cockroachdb-4",

"metrics-0"

]

* 停止后在DC/OS集群新位置上重启pod

dcos cockroachdb pods replace cockroachdb-<NUM>

访问日志进行问题定位

可以从DC/OS Web界面查看调度程序和服务（即CockroachDB）的日志。

* 调度程序的日志能够在调度程序的范围内，有效帮助确定节点未启动的原因。
* 节点的日志帮助定位检查服务本身的问题，例如CockroachDB的日志。

日志通常通过管道传输到stdout和/或stderr命名的文件。 查看给定节点的日志：

* 在DC/OS界面，进入**Service**表单
* 选择**cockroachdb**服务
* 在服务列表中，选择需要检查的任务，包括节点相关的cockroachdb-0-node-init或cockroachdb-#-node-join 。
* 关于任务的细节，可查看**Logs**标签页。

备份和恢复

cockroachdb DC/OS服务能够方便地使用CockroachDB的开源命令cockroach dump将基于每个数据库的数据备份到S3存储桶并从此类备份恢复数据。目前尚不支持使用S3以外的数据存储。

**TIPS:** 用户若需要S3以外备份和恢复功能，或是需要针对超大数据库进行快速备份、增量备份、快速分布式恢复，可以考虑获取License使用CockroachDB提供的企业版功能。

备份

执行以下命令备份数据库的表

dcos cockroachdb backup [<flags>] <database> <s3-bucket>

| **flags** | **简介** |
| --- | --- |
| --aws-access-key | AWS访问密钥 |
| --aws-secret-key | AWS Secret密钥 |
| --s3-dir | AWS S3目标文件夹 |
| --s3-backup-dir | s3-dir里的目标文件夹 |
| --region | AWS区域 |

默认情况下，AWS访问密钥和Secret密钥可以分别从AWS\_ACCESS\_KEY\_ID和AWS\_SECRET\_ACCESS\_KEY环境变量中提取。用户必须定义这些环境变量或指定备份的标签才能生效。

考虑到备份在上载到S3之前会存储在本地硬盘上，占用的空间与表中当前的数据一样多，用户需要确保为节点配置足够的磁盘空间以执行备份操作。

恢复

可以执行以下命令进行恢复操作：

dcos cockroachdb restore [<flags>] <database> <s3-bucket> <s3-backup-dir>

| **flags** | **简介** |
| --- | --- |
| --aws-access-key | AWS访问密钥 |
| --aws-secret-key | AWS Secret密钥 |
| --s3-dir | AWS S3目标文件夹 |
| --s3-backup-dir | s3-dir里的目标文件夹 |
| --region | AWS区域 |

默认情况下，AWS访问密钥和Secret密钥可以分别从AWS\_ACCESS\_KEY\_ID和AWS\_SECRET\_ACCESS\_KEY环境变量中提取。用户必须定义这些环境变量或指定备份的标签才能生效。

Step 7: 停止集群

* 卸载cockroachdb服务

MY\_SERVICE\_NAME=cockroachdb

dcos package uninstall --**app**-id=$MY\_SERVICE\_NAME $MY\_SERVICE\_NAME

* 对于使用1.9版本的DC/OS，使用[janitor.py](https://docs.mesosphere.com/1.9/deploying-services/uninstall/" \l "framework-cleaner)脚本清理残留的资源。而对于1.10版本的DC/OS，则不需要此步骤。

dcos node ssh --master-proxy --leader "docker run mesosphere/janitor /janitor.py \

-r $MY\_SERVICE\_NAME-role \

-p $MY\_SERVICE\_NAME-principal \

-z dcos-service-$MY\_SERVICE\_NAME"

* 卸载DC/OS。
* 如果使用AWS CloudFormation，需要[Uninstalling DC/OS on AWS EC2](https://docs.mesosphere.com/1.10/installing/oss/cloud/aws/removeaws/)。