CockroachDB为集群中的每个节点生成详细的时间序列度量标准。 本小节将展示将这些指标导入[Prometheus](https://prometheus.io/)中的过程，后者是一个用于存储、聚合和查询时间序列数据的开源工具。同时，还将展示了如何将[Grafana](https://grafana.com/)和[Alertmanager](https://prometheus.io/docs/alerting/alertmanager/)连接到Prometheus以获得多样的数据可视化和通知功能。

**TIPS:** 查看更多监控选项，可以阅读[Monitoring and Alerting](http://doc.cockroachchina.baidu.com/" \l "deploy/monitoring-and-alerting/overview/)。

前言

* 在[本地](https://www.cockroachlabs.com/docs/stable/start-a-local-cluster.html)或是[生产环境](http://doc.cockroachchina.baidu.com/" \l "deploy/manual-deployment/overview/)中部署了运行一个CockroachDB集群。
* 本小节的所有文件都存放在CockroachDB repository的[monitoring](https://github.com/cockroachdb/cockroach/tree/master/monitoring)文件夹。

Step 1: 安装Prometheus

* 下载跟操作系统对应的[2.x Prometheus tarball](https://prometheus.io/download/)
* 解压出可执行文件，添加到PATH，方便在命令行中运行Prometheus
* 确认Prometheus安装成功

prometheus --version

prometheus, version 2.2.1 (branch: HEAD, revision: bc6058c81272a8d938c05e75607371284236aadc)

build user: root@149e5b3f0829

build date: 20180314-14:21:40

go version: go1.10

Step 2: 配置Prometheus

获取CockroachDB对应的[Prometheus配置文件](https://github.com/cockroachdb/cockroach/blob/master/monitoring/prometheus.yml)

wget https://raw.githubusercontent.com/cockroachdb/cockroach/master/monitoring/prometheus.yml -O prometheus.yml

检查配置文件时可看到文件设置了每10秒采集一次单个非安全的本地节点的时间序列度量标准数据：

* scrape\_interval: 10s 代表采集周期
* metrics\_path: '/\_status/vars'代表CockroachDB面向Prometheus开放的用于获取时间序列指标的端点
* scheme: 'http'代表作为采集数据源的集群是不安全的的。
* targets: ['localhost:8080']代表CockroachDB节点用于收集时间序列指标数据的主机名和端口。

自定义配置文件

| **场景** | **需要修改的配置** |
| --- | --- |
| 本地多节点集群 | 将每个节点的地址以'localhost:<http-port>'格式添加到targets配置项中。 |
| 生产环境集群 | 配置每个节点的地址，以'localhost:<http-port>'格式添加到targets配置项中。同时确保机器的网络配置允许TCP连接到监视端点的端口。 |
| 安全集群 | 将scheme: 'http'替换为scheme: 'https'。 |

获取配置规则

创建rules文件夹，下载[聚合规则](https://github.com/cockroachdb/cockroach/blob/master/monitoring/rules/aggregation.rules.yml)和[报警规则](https://github.com/cockroachdb/cockroach/blob/master/monitoring/rules/alerts.rules.yml)，存放到该文件夹中。

mkdir rules

cd rules

wget -P rules https://raw.githubusercontent.com/cockroachdb/cockroach/master/monitoring/rules/aggregation.rules.yml

wget -P rules https://raw.githubusercontent.com/cockroachdb/cockroach/master/monitoring/rules/alerts.rules.yml

启动Prometheus

* 使用--config.file标签指定配置文件，启动Prometheus服务

prometheus --config.file=prometheus.yml

INFO[0000] Starting prometheus (version=1.4.1, branch=master, revision=2a89e8733f240d3cd57a6520b52c36ac4744ce12) source=main.go:77

INFO[0000] Build context (go=go1.7.3, user=root@e685d23d8809, date=20161128-10:02:41) source=main.go:78

INFO[0000] Loading configuration file prometheus.yml source=main.go:250

INFO[0000] Loading series map and head chunks... source=storage.go:354

INFO[0000] 0 series loaded. source=storage.go:359

INFO[0000] Listening on :9090 source=web.go:248

INFO[0000] Starting target manager... source=targetmanager.go:63

* 访问http://<hostname of machine running prometheus>:9090，使用Prometheus界面查询、聚合、显示CockroachDB时间序列指标。
* Prometheus自动完成CockroachDB时间序列指标，若需要更详尽的带描述的列表，访问http://<hostname of a CockroachDB node>:8080/\_status/vars。
* 更多Prometheus界面的使用细节，可以查看[官方文档](https://prometheus.io/docs/introduction/getting_started/)。

Step 4: 向Alertmanager发送通知

积极的监控可以帮助用户更早地发现问题，同时在发生干预和调查的事件时将情况及时通知用户，是十分有必要的。在**Step 2**下载了CockroachDB相关的报警规则，接下来需要下载、配置和启动[Alertmanager](https://prometheus.io/docs/alerting/alertmanager/)。

* 下载操作系统对应的最新版本[Alertmanager tarball](https://prometheus.io/download/" \l "alertmanager)
* 解压出可执行文件并添加到PATH，方便在命令行中运行Alertmanager

alertmanager --version

alertmanager, version 0.15.0-rc.1 (branch: HEAD, revision: acb111e812530bec1ac6d908bc14725793e07cf3)

build user: root@f278953f13ef

build date: 20180323-13:07:06

go version: go1.10

* 编辑解压出的配置文件，配置用户需要的通知接收项。
* 使用--config.file指定配置文件并启动Alertmanager服务

alertmanager --config.file=simple.yml

* 访问http://<hostname of machine running alertmanager>:9093查看Alertmanager运行情况。用户也可以在界面定义[消息警报（silencing alerts）](https://prometheus.io/docs/alerting/alertmanager/" \l "silences)。

Step 5: 可视化Grafana指标

Prometheus能够简单地绘制指标，而Grafana是另一个功能更强大的可视化工具，可以方便地与Prometheus集成。

* [下载、安装用户操作系统对应的Grafana](https://grafana.com/grafana/download)
* 访问http://<hostname of machine running grafana>:3000，使用默认的用户名/密码admin/admin来登陆Grafana界面，并创建自己的账户。
* [将Prometheus配置为数据源](http://docs.grafana.org/datasources/prometheus/)

| **配置项** | **配置值** |
| --- | --- |
| Name | Prometheus |
| Default | True |
| Type | Prometheus |
| Url | http://:9090 |
| Access | Direct |

* 下载CockroachDB对应的[Grafana仪表盘](https://github.com/cockroachdb/cockroach/tree/master/monitoring/grafana-dashboards)

*# runtime dashboard: node status, including uptime, memory, and cpu.*

wget https://raw.githubusercontent.com/cockroachdb/cockroach/master/monitoring/grafana-dashboards/runtime.json

*# storage dashboard: storage availability.*

wget https://raw.githubusercontent.com/cockroachdb/cockroach/master/monitoring/grafana-dashboards/storage.json

*# sql dashboard: sql queries/transactions.*

wget https://raw.githubusercontent.com/cockroachdb/cockroach/master/monitoring/grafana-dashboards/sql.json

*# replicas dashboard: replica information and operations.*

wget https://raw.githubusercontent.com/cockroachdb/cockroach/master/monitoring/grafana-dashboards/replicas.json

* [为Grafana添加仪表盘](http://docs.grafana.org/reference/export_import/#importing-a-dashboard)