

Prometheus

第一章 简介

Prometheus 受启发于 Google 的 Borgmon 监控系统（相似的 Kubernetes 是从 Google 的 Borg 系统演变而来），从 2012 年开始由前 Google 工程师在 Soundcloud 以开源软件的形式进行研发，并且于 2015 年早期对外发布早期版本。

2016 年 5 月继 Kubernetes 之后成为第二个正式加入 CNCF 基金会的项目，同年 6 月正式发布 1.0 版本。

2017 年底发布了基于全新存储层的 2.0 版本，能更好地与容器平台、云平台配合。Prometheus 作为新一代的云原生监控系统，目前已经有超过 650+位贡献者参与到 Prometheus 的研发工作上，并且超过 120+项的第三方集成。

特点

Prometheus 是一个开源的完整监控解决方案，其对传统监控系统的测试和告警模型进行了彻底的颠覆，形成了基于中央化的规则计算、统一分析和告警的新模型。相比于传统监控系统，Prometheus 具有以下优点（随便列举几个）：

易于管理

Prometheus 核心部分只有一个单独的二进制文件，不存在任何的第三方依赖(数据库，缓存等等)。唯一需要的就是本地磁盘，因此不会有潜在级联故障的风险。

Prometheus 基于 Pull 模型的架构方式，可以在任何地方（本地电脑，开发环境，测试环境）搭建我们的监控系统。

对于一些复杂的情况，还可以使用 Prometheus 服务发现(Service Discovery)的能力 动态管理监控目标。

监控服务的内部运行状态

Prometheus 鼓励用户监控服务的内部状态，基于 Prometheus 丰富的 Client 库，用户可以轻松的在应用程序中添加对 Prometheus 的支持，从而让用户可以获取服务和应用内部真正的运行状态。

强大的数据模型

所有采集的监控数据均以指标(metric)的形式保存在内置的时间序列数据库当中 (TSDB)。所有的样本除了基本的指标名称以外，还包含一组用于描述该样本特征的标签。

强大的查询语言

Prometheus 内置了一个强大的数据查询语言 PromQL。通过 PromQL 可以实现对监控数据的查询、聚合。同时 PromQL 也被应用于数据可视化(如 Grafana)以及告警当中。通过 PromQL 可以轻松回答类似于以下问题：

在过去一段时间中 95%应用延迟时间的分布范围？

预测在 4 小时后，磁盘空间占用大致会是什么情况？

CPU 占用率前 5 位的服务有哪些？(过滤)

构架

生态圈组件

Prometheus Server：主服务器，负责收集和存储时间序列数据

client libraies：应用程序代码插桩，将监控指标嵌入到被监控应用程序中

Pushgateway：推送网关，为支持 short-lived 作业提供一个推送网关

exporter：专门为一些应用开发的数据摄取组件-exporter，例如：HAProxy、StatsD、Graphite 等等。

Alertmanager：专门用于处理 alert 的组件

构架理解

Prometheus 既然设计为一个维度存储模型，可以把它理解为一个 OLAP 系统。

存储计算层

Prometheus Server，里面包含了存储引擎和计算引擎。

Retrieval 组件为取数组件，它会主动从 Pushgateway 或者 Exporter 拉取指标数据。

Service discovery，可以动态发现要监控的目标。

TSDB，数据核心存储与查询。

HTTP server，对外提供 HTTP 服务。

采集层

采集层分为两类，一类是生命周期较短的作业，还有一类是生命周期较长的作业。

短作业：直接通过 API，在退出时间指标推送给 Pushgateway。

长作业：Retrieval 组件直接从 Job 或者 Exporter 拉取数据。

应用层

应用层主要分为两种，一种是 AlertManager，另一种是数据可视化。

第二章 PromQL

Prometheus 通过指标名称 (metrics name) 以及对应的一组标签 (labelset) 唯一 定义一条时间序列。指标名称反映了监控样本的基本标识，而 label 则在这个基本特征上为 采集到的数据提供了多种特征维度。用户可以基于这些特征维度过滤，聚合，统计从而产生 新的计算后的一条时间序列。PromQL 是 Prometheus 内置的数据查询语言，其提供对时 间序列数据丰富的查询，聚合以及逻辑运算能力的支持。并且被广泛应用在 Prometheus 的日常应用当中，包括对数据查询、可视化、告警处理当中。可以这么说，PromQL 是 Prometheus 所有应用场景的基础，理解和掌握 PromQL 是 Prometheus 入门的第一课。

简单用法

- 1.PromQL支持使用 = 和 != 两种完全匹配模式。
- 2.可以通过[]来选择时间范围。PromQL的时间范围选择器支持的时间单位：s、m、h、d、w、y。
- 3.支持数学运算符。
- 4.支持布尔运算。

5.支持集合运算。

6.支持聚合操作：sum、min、max、avg、stdvar、stddev、count、count_values、bottomk、topk、quantile

第三章 集成Flink

拷贝新的flink目录，叫做flink-prometheus（在此不演示）

拷贝flink的jar包

```
1 cp /opt/module/flink-prometheus/plugins/metrics-prometheus/flink-metrics-prometheus-1.12.0.jar  
   /opt/module/flink-prometheus/lib/
```

进入到flink的conf目录，修改flink-conf.yaml

添加如下配置

```
1 ##### 与 Prometheus 集成配置 #####  
2 metrics.reporter.promgateway.class:  
3   org.apache.flink.metrics.prometheus.PrometheusPushGatewayReporter  
4   # PushGateway 的主机名与端口号  
5   metrics.reporter.promgateway.host: hadoop202  
6   metrics.reporter.promgateway.port: 9091  
7   # Flink metric 在前端展示的标签（前缀）与随机后缀  
8   metrics.reporter.promgateway.jobName: flink-metrics-ppg  
9   metrics.reporter.promgateway.randomJobNameSuffix: true  
10  metrics.reporter.promgateway.deleteOnShutdown: false  
11  metrics.reporter.promgateway.interval: 30 SECONDS
```

测试一下，启动网猫！

```
1 nc -lk 9999
```

提交任务

```
1 [atguigu@hadoop202 flink-prometheus]$ bin/flink run -t yarn-per-job -c  
2 com.atguigu.flink.chapter02.Flink03_WordCount_UnboundStream ./flink-base-1.0-SNAPSHOT-jar-with-  
dependencies.jar
```