

# 基于 Prometheus 的容器云监控平台应用研究<sup>①</sup>

谢超群

(福建中医药大学现代教育技术中心 福建福州 350000)

**摘要** 在高校数据中心信息化不断深入的背景下, 高校数据中心的各种应用服务逐步向微服务架构转变, 向容器云上进行迁移, 高校数据中心的传统监控模式已不适合对容器云进行监控。本文结合高校数据中心容器云监控存在的问题, 将开源的监控系统 Prometheus 作为 Kubernetes 容器云监控和智能告警的平台, 并对接开源可视化组件 Grafana 构建了前端可视化监控界面。该监控平台可以实现对数据中心 Kubernetes 容器云的各种资源对象的监控和智能告警, 解决了高校数据中心容器云监控的难题。

**关键词** Prometheus; 容器云; Kubernetes; Grafana; 数据中心

## Application Research of Container Cloud Monitoring Platform Based on Prometheus

Xie Chaoqun

(Modern Educational Technology Center, Fujian University of Traditional Chinese Medicine, Fuzhou, Fujian, 350000, China)

**Abstract** Under the background of deepening informatization of university data center, all kinds of application services of university data center are gradually transforming to micro service architecture and migrating to container cloud. The traditional monitoring mode of university data center is not suitable for monitoring container cloud. Combined with the existing problems of container cloud monitoring in university data center, this paper uses the open source monitoring system Prometheus as the platform of Kubernetes container cloud monitoring and intelligent alarm, and constructs the front-end visual monitoring interface with the open source visual component Grafana. The monitoring platform can realize the monitoring and intelligent alarm of various resource objects of Kubernetes container cloud in data center, and solve the problem of container cloud monitoring in university data center.

**Keywords** Prometheus; Container Cloud; Kubernetes; Grafana; Data Center

<sup>①</sup>本文系 2020 年福建省中青年教师科研基金项目“基于 Zabbix 高校数据中心运维监控平台的构建研究”(项目编号: JAT200213)的研究成果之一。

## 1 引言

近年来,随着高校信息化的不断深入,高校数据中心的许多应用服务已经逐渐从采用微服务架构转变到使用容器部署的方式。目前高校数据中心主要利用Docker容器技术和Kubernetes容器编排管理技术来构建学校的容器云平台,但传统的监控模式不具备完善的监控容器云的功能,导致高校数据中心在容器云平台的监控方面遇到了以下问题:由于Kubernetes管理的应用服务具备弹性扩展的特性,容器云中的应用容器会动态创建和销毁,传统监控模式无法自动发现和监控这些容器;容器云可能需要对应用运行的指标进行监控,但传统的监控模式无法对应用容器中的应用运行指标进行监控。

Prometheus是一款开源的监控系统,具备对Kubernetes管理的容器云的监控功能,可对容器云上的Node物理节点、应用容器、Kubernetes集群资源、容器应用运行状态进行动态发现、监控和智能告警。结合开源可视化组件Grafana对接Prometheus监控数据,作为容器云的监控前端,构建高校数据中心容器云监控平台,可解决目前高校数据中心容器云监控面临的难题。

## 2 Prometheus 简介

Prometheus是基于Google公司的Borgmon监控系统的设计理念,由SoundCloud公司采用Go语言开发的一款开源监控系统。与Google的Borgmon功能类似,Prometheus主要用于数据中心各种服务器、容器云平台和应用程序等的监控,具备强大的云原生监控功能,于2016年加入云原生计算基金会(CNCF),并成为继Kubernetes之后第二个CNCF认可的云原生项目<sup>[1]</sup>。

Prometheus主要由Prometheus Server、Exporter、Push Gateway、Alertmanager、Web UI等几大模块构成,Prometheus Server是Prometheus的核心组件,主要负责对监控对象和监控数据的采集、存储和查询,Prometheus Server采用轮询Pull的方式,从监控对象开放的接口中拉取监控数据。Prometheus Server内置高效的TSDB数据库,其抓取的监控数据按照时间序列存储在该数据库中,并使用自定义的PromQL,对外提供监控数据的查询和分析功能。Prometheus Server除了可以采用静态配置方式监控目标对象之外,还可以利用Service Discovery自动发现方式动态监控目标对象,比如自动感知Kubernetes动态创建的应用容器,并将新的应用容器加入监控列表<sup>[2]</sup>。Exporter安装在监控对象上,主要负责将监控数据转化为Prometheus Server能理解的数据形式,通过HTTP的End Point端点开放在网络上。Prometheus Server通过访问Exporter暴露的End Point端点来采集监控数据。Exporter分为两种类型:一类Exporter可以直接支持Prometheus的数据采集,其内置了Prometheus Server的HTTP的End Point端点,Prometheus Server可直接获取该类型Exporter的监控数据;另一类Exporter由于监控目标不支持Prometheus的数据采集,需采用Prometheus的Client Library来编写监控目标的监控采集程序Exporter。Push Gateway作为中间网关,主要负责对短期监控任务的数据采集,短期监控任务往往在Prometheus Server抓取监控数据之前就已经结束,因此需要Push Gateway来采集短期监控任务推送过来的监控数据,避免监控数据的丢失。Alertmanager主要负责管理来自Prometheus

Server推送过来的告警信息，具备告警去重、分组的功能，并可以自定义告警方式，可以支持邮件、微信、钉钉等主流媒介告警推送方式。Prometheus Server 可以利用PromQL自定义告警的规则，当PromQL创建的规则被触发时，Prometheus Server将产生告警并推

送到Alertmanager<sup>[3]</sup>。Web UI是Prometheus内置的Web图形界面，可以对Prometheus Server抓取的数据在浏览器中利用PromQL实现监控数据的查询以及可视化。Prometheus的体系架构图如图1所示。

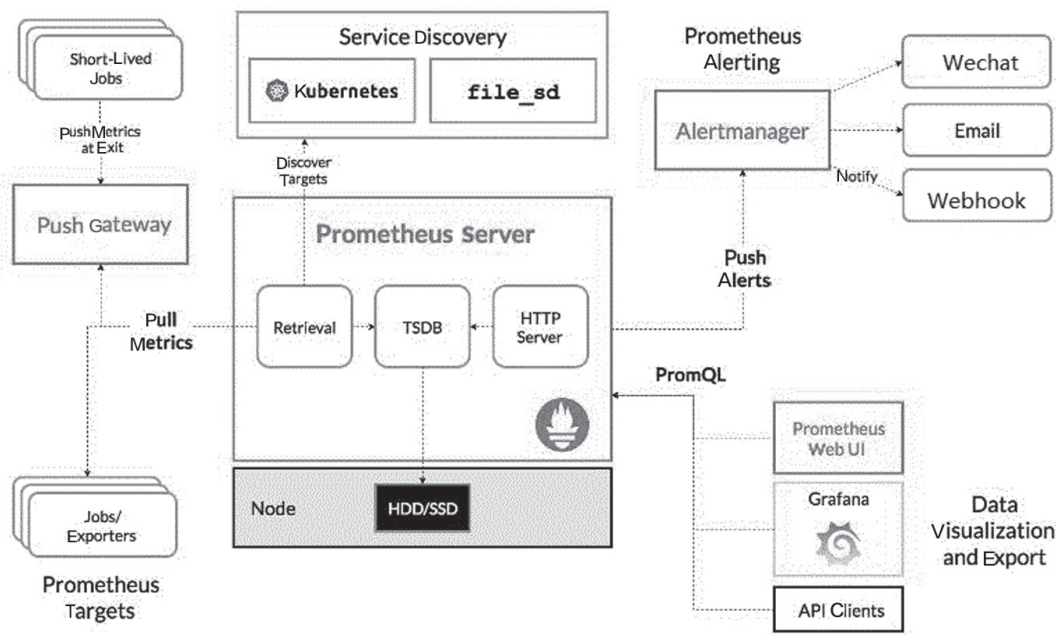


图 1 Prometheus 体系架构图

### 3 Kubernetes 容器云监控策略研究

Kubernetes是一款开源的容器编排工具，现已成为高校数据中心容器云管理的常见平台。从架构层次上分析，Kubernetes容器云的监控可以分为四个层次。第一层次是对容器云底层基础实施的监控，主要是对容器云的各种集群节点如物理机、虚拟机、云主机的监控，包含对集群节点的主机负载、CPU使用率、内存使用率、存储空间和网络IO等指标的监控，这些指标直接影响着整个容器云的运行状态。第二个层次是对 Kubernetes内部系统组件的监控，kube-apiserver是Kubernetes

提供的API服务接口，是Kubernetes所有服务的入口，Kubernetes容器云中的内部组件和外部客户端都需要与kube-apiserver通讯，才能完成与Kubernetes容器云的交互，因此需要监控kube-apiserver的可用性和并发数等指标，这些指标直接影响着Kubernetes容器云的正常运行。kube-scheduler负责完成容器POD在集群上调度工作，kubelet完成POD的创建和启动工作。这两个Kubernetes组件的运行状态直接影响着POD容器创建和启动的速度，因此需要监控kube-scheduler和kubelet的运行状态。第三个层次是对容器和POD性能指标的监

控。容器和POD是承载应用运行的系统资源，容器和POD占用的内存、CPU、网络IO等性能指标，直接影响着容器中应用的运行状态<sup>[4]</sup>。第四个层次是对容器中运行的应用服务本身的监控，比如容器中运行的Web服务和中间件服务的可用性和并发数等指标，这些是数据中心管理员都需要关注的应用指标，通过这些指标能随时掌握数据中心容器云中应用服务的动态情况。

#### 4 容器云监控平台应用研究

随着越来越多高校数据中心的应用服务采用微服务的架构，数据中心的应用服务从传统服务器的部署模式渐渐向容器部

署的模式转移，导致数据中心管理的应用容器增长迅速。数据中心的应用容器主要采用Kubernetes容器编排管理工具进行管理，Prometheus支持监控Kubernetes管理的容器云。通过Prometheus可以监控Kubernetes集群节点，Kubernetes的内部系统组件kube-apiserver、kube-scheduler、kubelet的运行情况，容器、POD相关的性能指标数据，Kubernetes资源对象的状态信息，应用运行状态等。使用Prometheus基本可以完成对Kubernetes管理的容器云各个层面的监控，高校数据中心容器云监控平台架构图如图2所示。

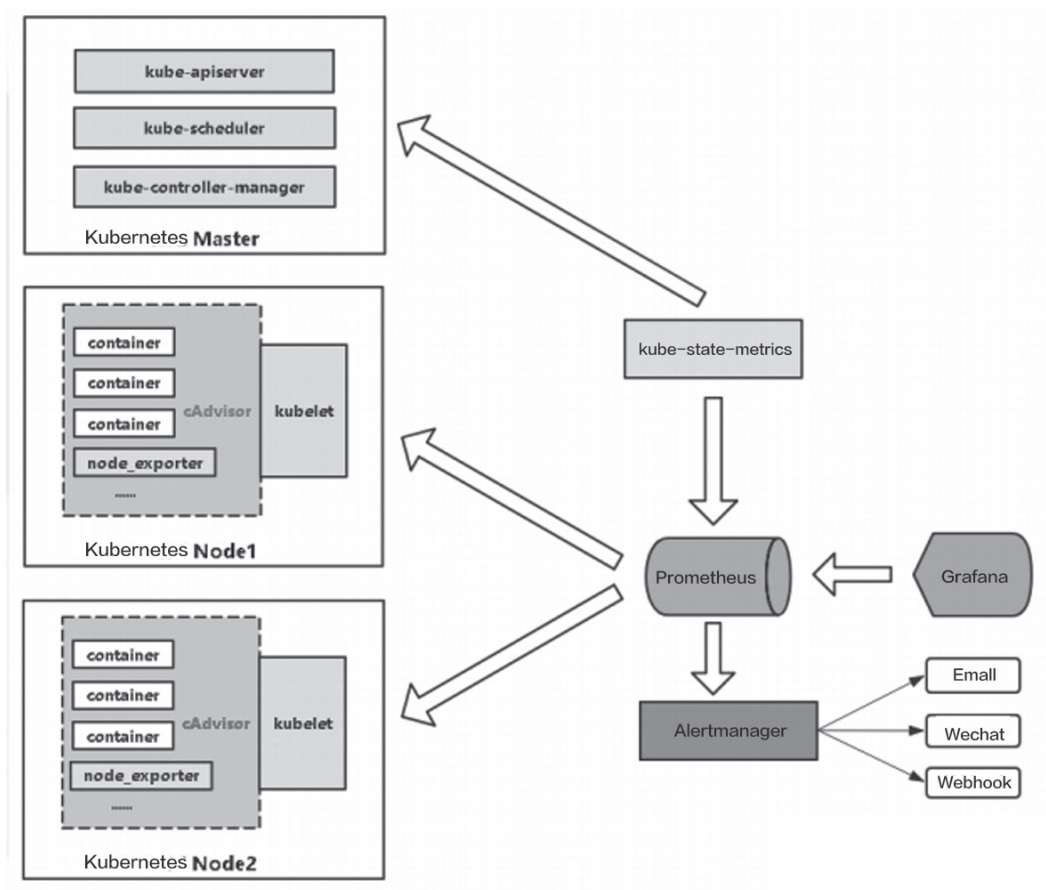


图2 容器云监控平台架构图



采用Prometheus监控Kubernetes容器云的具体实现思路如下：通过在每台Kubernetes集群节点部署node\_exporter，采集Kubernetes容器云节点资源使用的监控数据。node\_exporter的部署方式采用DeamonSet控制器来部署node\_exporter，保证容器云中每个节点上都会运行一个node\_exporter的POD实例，并会随着节点数的增减而动态调整node\_exporter的POD实例的数量。通过kube-apiserver、kube-scheduler、kubelet等Kubernetes内部组件开放的接口，采集Kubernetes容器云内部系统组件的监控数据。kube-apiserver、kube-scheduler、kubelet默认都有开放HTTP接口，Prometheus可以通过该接口来采集指标数据。cAdvisor是内置在Kubernetes的kubelet组件中的容器监控工具，默认部署在每个容器云节点之上，Prometheus通过查询cAdvisor暴露的HTTP接口，采集应用容器的相关性能指标，包含容器的内存、CPU、网络IO、资源IO等资源使用情况，并可以通过容器的监控数据进一步计算POD的性能指标<sup>[5]</sup>。Kube-state-metrics组件是Kubernetes官方提供的监控容器云资源对象的组件，采用Deployment部署的方式运行在Kubernetes容器云中，通过Kubernetes提供的API获取容器云中Node、POD、DeamonSet、Deployment、ingress、End Point等资源的状态信息，并将这些信息以HTTP接口的形式提供给Prometheus进行采集<sup>[6]</sup>。容器云上部署的应用服务的性能监控，可以在应用程序开发时利用Prometheus的Client Library开发应用的HTTP监控接口，将应用程序的运行相关监控数据暴露给外界，Prometheus可以通过该接口获取相应的应用运行监控数据。随着Kubernetes容器云

上的容器数量增多，Prometheus需要监控的目标对象越来越多。因此，Prometheus采用Deployment的方式部署在Kubernetes容器云上，可以保证Prometheus的POD随着监控目标的数量增多，进行相应的弹性扩容。Prometheus是一个实时的时间序列采集平台，默认只能存储近一段时间的数据，可以通过为Prometheus挂载远端存储（如OpenTSDB）来实现对监控数据的长期保存。

Kubernetes容器云监控的前端可以采用开源可视化组件Grafana对接Prometheus的监控数据来实现，具体思路如下：以Deployment的方式在容器云上部署开源可视化组件Grafana，安装Grafana的Prometheus数据源插件，配置Prometheus数据源插件，连接到Prometheus读取采集到的监控数据<sup>[7]</sup>。通过Grafana获取到Prometheus的监控数据后，针对采集到的监控数据，利用Grafana丰富的内置可视化组件和PromQL定制开发相应的Grafana 监控数据仪表盘。

Kubernetes容器云监控的智能告警可以采用Prometheus的Alertmanager告警管理组件来实现，具体思路如下：采用Deployment的方式在Kubernetes容器云上部署Alertmanager，接受来自Prometheus的各种告警。通过配置Prometheus的规则文件来定义各种告警触发规则，同时用Alertmanager 的配置文件来定义各种信息媒介的推送方式<sup>[8]</sup>。微信告警的推送方式如下：可以先申请一个微信公众号，并配置微信相关的wechat\_api连接信息和微信推送信息模板来实现。由于Alertmanager默认不支持钉钉告警，需要在容器云上先部署一个prometheus-webhook-dingtalk的Deployment，该Deployment已经默认

配置好了钉钉机器人信息推送接口URL。然后通过配置Alertmanager的Webhook, 对接到prometheus-webhook-dingtalk容器服务上, 完成Alertmanager与钉钉机器人Webhook信息推送接口的对接。

通过上述基于Prometheus的 Kubernetes 容器云监控平台思路的实践, 基本可以实现高校数据中心容器云上各种资源对象的可视化监控和智能告警, 解决高校数据中心容器云监控的难题, 提高高校数据中心容器云的运维监控水平。

## 5 结语

随着微服务架构在高校数据中心的逐渐落地, 高校数据中心面临着高效监控微服务容器云的难题。Prometheus具备强大的云原生监控功能, 可对数据中心Kubernetes容器云的集群节点、内部系统组件、容器以及POD和应用服务进行监控。利用可视化对接组件Grafana对接 Prometheus监控数据源, 构建Kubernetes容器云可视化监控平台, 并通过Prometheus的Alertmanager告警管理组件进行监控平台的智能告警, 可以实现对高校数据中心容器云的各种资源对象的监控和智能告警, 解决高校数据中心容器云监控的难题。

## 参考文献

- [1] The Linux Foundation. Prometheus Documentation. [EB/OL]. [2021-05-21]. <https://prometheus.io/docs/introduction/overview/>.
- [2] 布莱希. Prometheus学习手册[M]. 宋佳洋, 薛锦, 译. 北京: 中国电力出版社, 2020: 16—19.
- [3] 陈金窗, 刘政委, 张其栋, 等. Prometheus监控技术与实践[M]. 北京: 机械工业出版社, 2020: 116—120.
- [4] 特恩布尔. Prometheus监控实战[M]. 史天, 张媛, 肖力, 译. 北京: 机械工业出版社, 2019: 53—55.
- [5] 陈晓宇. 深入浅出Prometheus[M]. 北京: 电子工业出版社, 2019: 171—174.
- [6] GitHub. Kube-State-Metrics Documentation. [EB/OL]. [2019-01-11]. <https://github.com/kubernetes/kube-state-metrics/>.
- [7] 朱政科. Prometheus云原生监控: 运维与开发实战[M]. 北京: 机械工业出版社, 2020: 62—67.
- [8] The Linux Foundation. Alertmanager Documentation. [EB/OL]. [2021-06-11]. <https://prometheus.io/docs/alerting/latest/alertmanager/>.