

转Prometheus VS InfluxDB

2018年05月14日 09:49:35LarryHai6 阅读数：866 标签：monitoringtime series dbalert更多

前言

除了传统的监控系统如 Nagios, Zabbix, Sensu 以外，基于时间序列数据库的监控系统随着微服务的兴起越来越受欢迎，比如 Prometheus，比如 InfluxDB。gtt 也尝试了一下这两个系统，希望能找到两者的差别，为以后选型提供一些帮助。

首先，说道时间序列数据库不得不说老牌的 rrdtools 和 [graphite](#)，这些经典老系统工作的非常好，除了有人嫌弃它们在巨大规模情景下不 scale，嫌弃它们部署不方便外。于是有了 OpenTSDB, Prometheus, InfluxDB 等这些后起之秀。

监控系统

OpenTSDB

OpenTSDB：基于 Hadoop and HBase 的时间序列数据库，它最先提出了为 metric 增加 tag（key-value 键值对）的方法来实现更方便和强大的查询语法，InfluxDB 的设计和查询语法受它的启发很大。OpenTSDB 基于 Hadoop 和 HBase 的实现了变态的横向扩展能力，但是也因为这两个依赖，对于不熟悉 Hadoop 这套系统的团队来说，OpenTSDB 的维护成本很高，于是有人搞出了 InfluxDB。

InfluxDB

InfluxDB：InfluxData 公司使用 golang 实现的时间序列数据库，InfluxDB 的口号之一就是：From the ground up，没有任何外部依赖，就一个可执行文件，丢到服务器上就可以运行，对运维非常之友好。语法的设计很大程度受到 OpenTSDB 的启发。虽然项目初期标榜了自带集群功能，可以非常轻松地实现横向扩展，但是在 InfluxDB 1.0 之后集群功能被删除，取而代之的是通过 Relay 模式实现高可用，官方文档上挂出如下说明，[但是0.9版本的集群使用说明在官网上仍然能访问到，估计未来被删除的可能性非常大。](#)

Note: Clustering is now a commercial product called InfluxEnterprise. More information can be found here.

Prometheus

Prometheus：SoundCloud 开源的监控系统，已经提交给开源社区独立运营。并且和 k8s 一样都为[Cloud Native Computing Foundation](#) 的成员，虽然目前这个 Foundation 只有 k8s 和 Prometheus 两个项目。Prometheus 和上面两者最大的区别可以理解成：上面两者仅仅是数据库，而 Prometheus 是一个监控系统，它不仅包含了时间序列数据库，还有全套的抓取、检索、绘图、报警的功能。[官方也对这种区别做了详细的描述。](#)

它很大程度收到了 Google 内部的 Borgmon 系统的启发，基于拉（pull）模式实现的监控系统。在《Site Reliability Engineering》一书中有这句话提到 Prometheus，当然我不会告诉你原文其实还提到了 Bosun 和 Riemann 这两个监控系统，这是为什么这面这句话末尾有个省略号：

Even though Borgmon remains internal to Google, the idea of treating time-series data as a data source for generating alerts is now accessible to everyone through those open source tools like Prometheus [...]
— Site Reliability Engineering: How Google Runs Production Systems (O'Reilly Media)

不过 InfluxData 公司也推出了整套的围绕时间序列数据库的解决方案：TICK，功能覆盖了数据获取（Telegraf）、存储和查询（InfluxDB）、图表绘制（Chronograf）、报警（Kapacitor）。这套解决方案和 Elastic 公司的做法特别像：围绕着 ElasticSearch 核心功能，收购了 Logstash, Kibana，又搞出了 Beat、Watcher 等外围服务打造完整的功能完备的全文检索解决方案。

扯得有点远，回到文章的核心内容：InfluxDB 和 Prometheus 的区别是啥。目前主要区别在于：前者仅仅是一个数据库，它被动的接受客户的数据插入和查询请求。而后者是完整的监控系统，能抓取数据、查询数据、报警等功能。

Push vs Pull

到此，我们知道 Prometheus 是基于 pull 的，InfluxDB 是基于 push 的。关于 push 和 pull 之前写过 [ansible 和 puppet 的对比](#)，但是在监控系统上，又有了微妙的差别。

首先，Push 和 Pull 描述的是数据传输的方式，它不影响传输的内容。换言之，只要是 push 能够携带的信息，pull 肯定也能携带同样的信息，比如 "CPU 利用率 30%" 这样的监控数据，不管是 pull 还是 push，传输的内容还是这些，不会因为传输模式改变导致消息体积暴长，因此两种方式消耗的网络带宽不会差别很大。

gtt 认为 Push 和 Pull 的主要区别在：

发起者不同

pull 的发起人是监控系统，它依次轮询被监控目标，所以如果目标在防火墙内或者 NAT 之后，则 Pull 方式行不通。并且，对于批处理（batch）类型的任务，因为可能整个处理时间小于轮询的间隔时间，因此监控系统会捕捉不到这类任务的数据。

为了解决这两个问题 Prometheus 提供了 pushgateway_exporter 组件来支持 push 模式的监控需求。

push 要求发起人是被监控目标，所以它可以突破防火墙限制，即使目标躲在 NAT 之后，仍然能顺利将数据推送出来，对于批处理类型的任务也能比较从容的发出数据。

这里 gtt 不同意，因为 pull 解决单点故障的方法是增加另外一个监控系统，本质上是是通过数据冗余提高可靠性，那么 push 为什么不能推送到两个监控系统上呢，这样也能做到数据冗余。

对于性能瓶颈，这点更不成立，因为不管是 push 还是 pull，影响的只是传输方式，对传输的数据内容没有影响，占用的带宽是一样的。那么唯一区别是并发度可能不一样，push 模式下，目标服务可能在某个时间内集中向监控系统推送数据，导致瞬间并发请求很大，类似 DDos 攻击。相反地，pull 以此轮询目标服务，能够按照自己能够承受的并发度处理监控数据，避免了监控数据短时间内爆发的情况。但解决办法也有，在 push 模式下，给监控服务加上请求处理队列，超过监控系统负载的请求暂存在队列中，这样监控系统就能按照自己的节奏来处理数据，防止被队友给DDoS。

所以单点问题、性能问题不是两种模式的本质区别。

逻辑架构不同

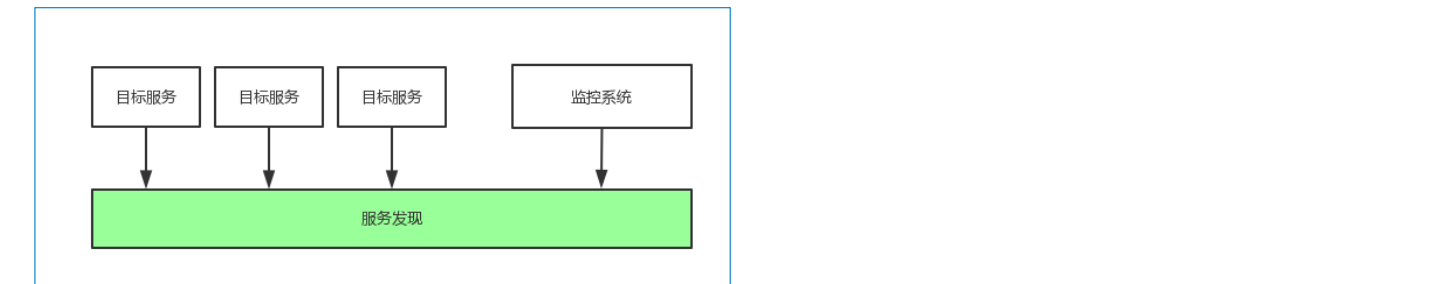
push 要求被监控目标知道监控系统的地址（IP或者域名），所以这部分信息需要设置在目标服务中，换言之，目标服务依赖监控系统。监控系统如果地址改变，所有目标服务都需要做相应的改动。而一旦产生依赖意味着监控系统故障，可能会影响到目标服务正常运行，当然在编程时可以做一些规避，但是逻辑上仍然是目标服务依赖监控系统。架构图下图所示：



pull 要求监控系统知道所有目标服务的地址，目标服务对监控系统是不知情的。所以监控系统依赖目标服务，每次新增加一个目标服务，对监控系统做配置修改。从这点区别上看，pull 模式更加符合逻辑架构。为了自动化处理目标服务的增加和删除，Prometheus 支持从服务发现系统中动态获取目标服务的地址，省去了大规模微服务部署情况下复杂的配置需求。逻辑架构如下图所示：



鸡贼的人应该发现了，那岂不是服务发现系统被所有目标服务依赖了？是的，服务发现系统和监控系统在逻辑架构上处于不同地位，在有服务发现的架构中，如果目标服务没有被“发现”，它实际上是是不能正常提供服务的，所以必须依赖服务发现系统，而相反，目标服务在没有监控系统的情况下仍然可以正常运行。逻辑架构如下图所示：



因为不依赖监控系统，即使没有部署监控服务，人工判断目标服务是否正常也非常容易，只要模拟监控系统访问目标服务的某个接口即可，所以 pull 模式下的监控更向白盒，你可以很轻松的获取到所有信息。相反的，push 模式依赖于一个成型的监控服务，没有监控服务就完全不知道目标服务运行状况如何，这点比较让人难以接受。

查询语法

到此，我们知道 Prometheus 是基于 pull 模式获取数据，InfluxDB 是基于 push 模式获取数据。现在关注两者在数据查询上的区别。

比如获取磁盘 IO 时间的数据：

```
时间戳      metric: 值  tag
1475216224  disk_io_time:10  type="sda"
1475216224  disk_io_time: 30  type="sdb"
1475216224  disk_io_time: 11  type="sdc"
1475216224  disk_io_time: 18  type="sde"
```

作为基本的时间序列数据库，两者对数据的基本获取都很简单。

InfluxDB:

```
SELECT mean("value") FROM "disk_io_time" WHERE $timeFilter GROUP BY time($interval), "instance" fill(null)
```

Prometheus:

```
disk_io_time
```

基本的算数计算

两者的差别也不大：

InfluxDB:

```
SELECT mean("value") *1024 FROM "disk_io_time" WHERE $timeFilter GROUP BY time($interval), "instance" fill(null)
```

Prometheus:

```
disk_io_time*1024
```

计算速度

InfluxDB:

```
SELECT derivative(mean("value"), 10s) *1024 FROM "disk_io_time" WHERE $timeFilter GROUP BY time($interval), "instance" fill(null)
```

Prometheus:

```
rate(disk_io_time)*1024
```

维度之间的计算

这点是目前为止 gtt 发现的两者最大区别。比如我需要 sda 和 sdc 的 io 时间相加，InfluxDB 还不支持这样的语法，不过社区已经在讨论相关的实现了：[\[feature request\] Mathematics across measurements #3552](#)。

而 Prometheus 能够完成这个任务：

```
rate(disk_io_time{type="sda"}) + rate(disk_io_time{type="sdc"})
```

总结

整体比较下来，Prometheus 是一个靠谱的监控系统，它的设计深受到 Google 内部 Borgmon 系统的启发，并且有着优雅的查询语法，不过是基于拉（pull）模式的，需要在具体业务中做抉择。而 InfluxDB 仅仅是时间序列数据库，没有其他监控相关的功能，不过 InfluxData 公司还提供了配套的其他组件可供选择。于 Prometheus 相比，它的查询语法更加复杂，并且不支持维度之间的计算。

基于Prometheus做多维度的容器监控 - 一云博客

什么是prometheus? prometheus从官方介绍来说，他是一个开源的系统监控和报警工具，最初由SoundCloud推出。自2012成立以来，许多公司和组织都采用了prometheus，项目有...

 想对作者说点什么

- prometheus丢数据调试与处理 - 辉哥

prometheus在大数据下处理丢数据

1451

来自： 辉哥

Prometheus+ Grafana 微服务系统监控方案搭建 - zjh_746140129的博客

Prometheus是由 SoundCloud 开发的开源监控报警系统和时序序列数据库(TSDB).自2012年起,许多公司...

833

来自： zjh_746140129的博客

prometheus学习笔记（一） - 想飞天的菜鸟

prometheus： 1.prometheus是一套开源的监控&报警&时间序列数据库的组合； 2.prometheus特性： ...

1884

来自： 想飞天的菜鸟

不用离家，[区块链](#)开发八周学会！

- [telegraf](#)将数据写入[influxdb](#) - 约会远行的专栏

[开发者调查](#)
[Python学习路线!](#)
[会员任意学](#)
[中国大数据技术大会](#)
[什么是大数据](#)
[什么是服务器](#)
[广告](#)
[登录](#)
[注册](#)
[×](#)

- 记一次**InfluxDB**故障 - 柳清风的专栏

1183

在一个测试环境中，InfluxDB已经运行了一年多，今天突然发现不能保存监控数据了。我们先回顾一...

来自： [柳清风的专栏](#)
- 时序列数据库武斗大会之 **TSDB** 名录 **Part 1** - wangpeng198688的专栏

1393

通过上一章《时序列数据库武斗大会之什么是TSDB》的介绍，相信大家已经知道了什么是时序列数据...

来自： [wangpeng198688的...](#)
- docker**：快速构建容器监控系统**cAdvisor+InfluxDB+Grafana** - To be a better man~

476

容器监控的方案有很多，有docker自身的docker stats命令、有Scout、有Data Dog等等，本文主要和...

来自： [To be a better man~](#)



假蜜大行其道，真蜜无人问津！蜂农辛酸泪流！这才是正宗的蜂蜜！

智数互动 · 顶新

prometheus+grafana构建应用监控（一） - haoziwlh的博客

1775

监控目标 基础监控 服务器基础指标（cpu load、内存、文件句柄数、用户线程数、端口连接数、存储...

来自： [haoziwlh的博客](#)

时间序列数据库**InfluxDB** OpenTSDB - fanren224的博客

665

doing

来自： [fanren224的博客](#)

- 饿了么 **Influxdb** 实践之路 - GitChat

258

目前饿了么基础设施的监控存储使用的是时序数据库 influxdb，监控规模达到 2 万台机器以上，每秒...

来自： [GitChat](#)
- 搭建 **RabbitMQ Server** 高可用集群 - 大JAVA解决方案

96

搭建 RabbitMQ Server 高可用集群阅读目录：准备工作搭建 RabbitMQ Server 单机版RabbitMQ Serv...

来自： [大JAVA解决方案](#)
- 时间序列数据的存储和计算 - 开源时序数据库解析 - zhoushuntian的博客

279

摘要： Prometheus 开源时序数据库解析的系列文章在之前已经完成了几篇，对比分析了Hbase系的O...

来自： [zhoushuntian的博客](#)



我们对天发誓：这正版传奇爆率9.8，送VIP，卸载算我输！

贪玩游戏 · 顶新

Prometheus监控的最佳实践——关于监控的3项关键指标 -

663

本文来自Weaveworks的工程师Anita Burhle在Rancher Labs与Weaveworks联合举办的Online Meetu...

来自： [RancherLabs的博客](#)

InfluxDB介绍 - vtnews的专栏

398

转白：http://www.cnblogs.com/iiiiher/p/8046600.html [svc]influxdb最佳实战-监控...

来自： [vtnews的专栏](#)

- 解惑|你是否为容器监控操碎了心？ - shurenyun的博客

1501

导读：容器对于物理机和虚拟机，单从监控上看就不是一个数量级的，但监控又是至关重要的，没有...

来自： [shurenyun的博客](#)
- 关于**prometheus**设计的一些思考 - 柳清风的专栏

5871

其实设计一套完整的监控系统是挺复杂的，既要考虑的通用性，也要考虑的对各种指标和监控对象的...

来自： [柳清风的专栏](#)
- InfluxDB+Grafana**整合及**metrics**没有显示问题 - 坤哥的博客

4901

InfluxDB与Grafana整合很简单，但是里面也有一些坑，这里简单说下我的整合过程。InfluxDB我之...

来自： [坤哥的博客](#)



卖疯了！德国智能扫地机器人,仅售348元！

查准教育 · 顶新

InfluxDB源码编译、安装、配置及主从同步实现 - 無痕剑的博客

712

InfluxDB使用源码编译 公司有个项目要求使用InfluxDB时序数据库存储点东西。第一次听说还有这种...

来自： [無痕剑的博客](#)

influxdb介绍 - gongpulin的博客

315

InfluxDB 是一个时间序列数据库，用于处理海量写入与负载查询。InfluxDB旨在用作涉及大量时间戳...

来自： [gongpulin的博客](#)

kubernetes学习记录（10）——建立**Heapster+Influxdb+Grafana**集群性能监控平台 - 个人...

1332

采用的是Heapster+Influxdb+Grafana建立集群性能监控平台。据说Heapster需要与Kubernetes Maste...

来自： [个人学习记录](#)

- Grafana+Prometheus**打造全方位立体监控系统 - kwame211的博客

4187


摘要： 前言 本文主要介绍如何使用Grafana和Prometheus以及node_exporter对Linux服务器性能进行...

来自： [kwame211的博客](#)
- 性能工具之**JMeter+InfluxDB+Grafana**打造压测可视化实时监控 - Mo小泽的技术博客

228

概述 本文将介绍如何使用JMeter+InfluxDB+Grafana打造压测可视化实时监控。引言 我们很多时...

来自： [Mo小泽的技术博客](#)



卖疯了！德国智能扫地机器人,仅售348元！

查准教育 · 顶新

开发者调查

Python学习路线！

会员任意学

中国大数据技术大会

什么是大数据

什么是服务器

广告

登录

注册

×

- 基于InfluxDB&Grafana的JMeter实时性能测试数据的监控和展示 - RickyOne_RR的博客 7971
- 本文主要讲述如何利用JMeter监听器Backend Listener, 配合使用InfluxDB+Grafana展示实时性能测... 来自: RickyOne_RR的博客

<https://blog.csdn.net/u011537073/article/details/80305804> 6/8

logstash 学习小记

个人分类

IT-编程语言-Perl

1篇

IT-编程语言-Java

44篇

IT-应用框架-Struts

5篇

IT-应用框架-Spring

36篇

IT-应用框架-Hibernate

4篇

展开

归档

2018年9月

4篇

2018年8月

2篇

2018年7月

6篇

2018年6月

3篇

2018年5月

14篇

展开

热门文章

基于Token的WEB后台认证机制

阅读量： 34887

iOS开发入门教程

阅读量： 31184

Oauth2.0 用Spring-security-oauth2 非常简单

阅读量： 29034

用于HTML5移动开发的10大移动APP开发框架

阅读量： 27257

h5做app和原生app有什么区别？

阅读量： 21786

最新评论

Vue 爬坑之路（三）—— 使用 ...

u012605477： 百度云链接失效了楼主，能重新再发布一个嘛？

Vue 爬坑之路（三）—— 使用 ...

qq_38476773： 这么写，怎么看懂 草

推荐一款基于Java的音视频处理开...

cheng_sl1： 请问，楼主在转码视频之后，没有出现音频消失的情况吗？

在Docker环境下部署Kafka

u011762977： kafka启动了,但是报错 missing mandatory config: KAFKA_ZOO...

前端框架大比拼：Angularjs...

fantian001： 你确定你是微软工作的？ 还 hello world ？ vue-cli 脚手架不能一键 hello ...

入台证要多久

联系我们

 QQ客服

 kefu@csdn.net

 客服论坛

 400-660-0108

工作时间 8:00-22:00

关于我们 | 招聘 | 广告服务 | 网站地图

 百度提供站内搜索 京ICP证09002463号

©1999-2018 江苏乐知网络技术有限公司

江苏知之为计算机有限公司 北京创新乐知

信息技术有限公司版权所有

网络110报警服务

经营性网站备案信息

北京互联网违法和不良信息举报中心

中国互联网举报中心



VIP
免广告



