运维监控工具考察专题

运维行业有句话：“无监控、不运维”，是的，一点也不夸张，监控俗称“第三只眼”。没了监控，什么基础运维，业务运维都是“瞎子”。

所以说监控是运维这个职业的根本。尤其是在现在DevOps这么火的时候，用监控数据给自己撑腰，这显得更加必要。

有人说运维是背锅侠，那么，有了监控，有了充足的数据，一切以数据说话，运维还需要背锅吗，所以作为一个运维工程师，如何构建一套监控系统是你的第一件工作。

**1、常见的运维监控工具**

现在运维监控工具非常多，哪个好，哪个不好，哪个适合你，哪个不适合你，其实只有你了解了他们的特性后，才知道，所以从这里开始讲起。

* **Cacti**

Cacti是一套基于PHP,MySQL,SNMP及RRDTool开发的网络流量监测图形分析工具。

简单的说Cacti就是一个PHP程序。它通过使用SNMP协议获取远端网络设备和相关信息，（其实就是使用Net-SNMP 软件包的snmpget 和snmpwalk 命令获取）并通过RRDTOOL工具绘图，通过PHP程序展现出来。我们使用它可以展现出监控对象一段时间内的状态或者性能趋势图。

Cacti是很老的一款监控工具了，其实说它是一款流量监控工具更合适，对流量监控比较精准，但缺点很多，出图不好看，不支持分布式，也没有告警功能，所以使用的人会越来越少。

* **Nagios**

Nagios是一款开源的免费网络监视工具，能有效监控Windows、Linux和Unix的主机状态，交换机路由器等网络设置，打印机等。在系统或服务状态异常时发出邮件或短信报警第一时间通知网站运维人员，在状态恢复后发出正常的邮件或短信通知。

Nagios主要的特征是监控告警，最强大的就是告警功能，可支持多种告警方式，但缺点是没有强大的数据收集机制，并且数据出图也很简陋，当监控的主机越来越多时，添加主机也非常麻烦，配置文件都是基于文本配置的，不支持web方式管理和配置，这样很容易出错，不宜维护。

* **Zabbix**

Zabbix是一个基于WEB界面的提供分布式系统监视以及网络监视功能的企业级的开源解决方案。zabbix能监视各种网络参数，保证服务器系统的安全运营；并提供强大的通知机制以让系统运维人员快速定位/解决存在的各种问题。

Zabbix由2部分构成，zabbix server与可选组件zabbix agent。zabbix server可以通过SNMP，zabbix agent，ping，端口监视等方法提供对远程服务器/网络状态的监视，数据收集等功能，它可以运行在Linux, Solaris, HP-UX, AIX, Free BSD, Open BSD, OS X等平台上。

Zabbix解决了cacti没有告警的不足，也解决了nagios不能通过web配置的缺点，同时还支持分布式部署，这使得它迅速流行起来，zabbix也成为目前中小企业监控最流行的运维监控平台。

当然，Zabbix也有不足之处，它消耗的资源比较多，如果监控的主机非常多时，可能会出现监控超时、告警超时等现象，不过也有很多解决办法，比如提高硬件性能、改变zabbix监控模式等。

* **Ganglia**

Ganglia是一款为HPC（高性能计算）集群而设计的可扩展的分布式监控系统，它可以监视和显示集群中的节点的各种状态信息，它由运行在各个节点上的gmond守护进程来采集CPU 、内存、硬盘利用率、I/O负载、网络流量情况等方面的数据，然后汇总到gmetad守护进程下，使用rrdtool存储数据，最后将历史数据以曲线方式通过PHP页面呈现。

Ganglia监控系统有三部分组成，分别是gmond、gmetad、webfrontend。gmond安装在需要收集数据的客户端，gmetad是服务端，webfrontend是一个php的web ui界面，ganglia通过gmond收集数据，然后在webfrontend进行展示。

Ganglia的主要特征是收集数据，并集中展示数据，这是ganglia的优势和特色，ganglia可以将所有数据汇总到一个界面集中展示，并且支持多种数据接口，可以很方面的扩展监控，同时，最为重要的是，ganglia收集数据非常轻量级，客户端的gmond程序基本不耗费系统资源，而这个特点刚好弥补了zabbix消耗性能的不足。

最后，Ganglia在对大数据平台的监控更为智能，只需要一个配置文件，即可开通Ganglia对hadoop、spark的监控，监控指标有近千个，完全满足了对大数据平台的监控需求。

* **Centreon**

Centreon是一款功能强大的分布式IT监控系统，它通过第三方组件可以实现对网络、操作系统和应用程序的监控：首先，它是开源的，我们可以免费使用它；其次，它的底层采用类似nagios的监控引擎作为监控软件，同时监控引擎通过ndoutil模块将监控到的数据定时写入数据库中，而Centreon实时从数据库读取该数据并通过Web界面展现监控数据；最后，我们可以通过Centreon web一键管理和配置主机，或者说Centreon就是nagios的一个管理配置工具，通过Centreon提供的Web配置界面，可以轻松完成nagios需要手工配置主机和服务的不足。

entreon的强项是一键配置和管理，并支持分布式监控，nagios能够完成的功能，通过centreon都能实现，同时，centreon还可以和ganglia进行集成，centreon将ganglia收集到的数据进行整合，可以实现主机自动加入监控以及自动告警的功能。

* **Prometheus**

Prometheus是一套开源的系统监控报警框架，它既适用于面向服务器等硬件指标的监控，也适用于高动态的面向服务架构的监控。对于现在流行的微服务，Prometheus的多维度数据收集和数据筛选查询语言也是非常的强大。Prometheus是为服务的可靠性而设计的，当服务出现故障时，它可以使你快速定位和诊断问题。

* **Grafana**

Grafana是一个开源的度量分析与可视化套件，通俗的说，Grafana就是一个图形可视化展示平台，它通过各种炫酷的界面效果展示我们的监控数据，

如果你觉得zabbix的出图界面不够好看，逼格不够高，就可以使用Grafana的可视化展示，同时，Grafana支持许多不同的数据源，Graphite，InfluxDB，OpenTSDB，Prometheus，Elasticsearch，CloudWatch和KairosDB都可以完美支持。



### 2、统一运维监控平台设计思路

运维监控平台不是简单的下载一个开源工具，然后搭建起来就行了，它需要根据监控的环境和特点进行各种整合和二次开发，以达到与自己的需求完全吻合的程度。那么下面就谈谈运维监控平台的设计思路。

构建一个智能的运维监控平台，必须以运行监控和故障报警这两个方面为重点，将所有业务系统中所涉及的网络资源、硬件资源、软件资源、数据库资源等纳入统一的运维监控平台中，并通过消除管理软件的差别。

数据采集手段的差别，对各种不同的数据来源实现统一管理、统一规范、统一处理、统一展现、统一用户登录、统一权限控制，最终实现运维规范化、自动化、智能化的大运维管理。

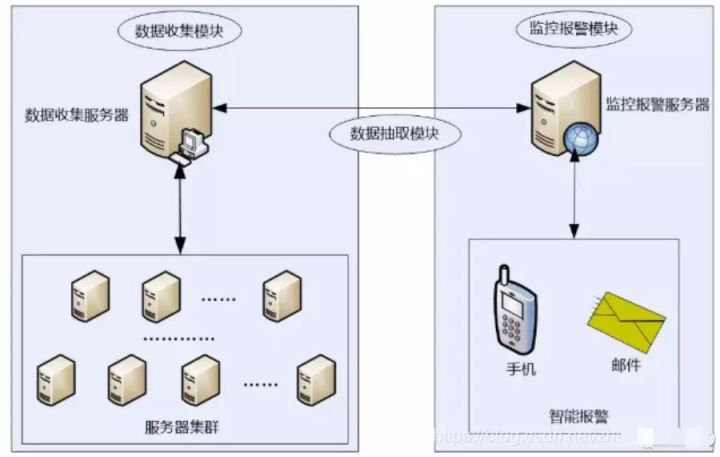
智能的运维监控平台，设计架构从低到高可以分为6层，三大模块，如下图：



* **数据收集层：**位于最底层，主要收集网络数据、业务系统数据、数据库数据、操作系统数据等，然后将收集到的数据进行规范化并进行存储。
* **数据展示层：**位于第二层，是一个Web展示界面，主要是将数据收集层获取到的数据进行统一展示，展示的方式可以是曲线图、柱状图、饼状态等，通过将数据图形化，可以帮助运维人员了解一段时间内主机或网络的运行状态和运行趋势，并作为运维人员排查问题或解决问题的依据。
* **数据提取层：**位于第三层，主要是对从数据收集层获取到的数据进行规格化和过滤处理，提取需要的数据到监控报警模块，这个部分是监控和报警两个模块的衔接点。
* **报警规则配置层：**位于第四层，主要是根据第三层获取到的数据进行报警规则设置、报警阀值设置、报警联系人设置和报警方式设置等。
* **报警事件生成层：**位于第五层，主要是对报警事件进行实时记录，将报警结果存入数据库以备调用，并将报警结果形成分析报表，以统计一段时间内的故障率和故障发生趋势。
* **用户展示管理层：**位于最顶层，是一个Web展示界面，主要是将监控统计结果、报警故障结果进行统一展示，并实现多用户、多权限管理，实现统一用户和统一权限控制。

在这6层中，从功能实现划分，又分为三个模块，分别是数据收集模块、数据提取模块和监控报警模块，每个模块完成的功能如下：

* **数据收集模块：**此模块主要完成基础数据的收集与图形展示。数据收集的方式有很多种，可以通过SNMP实现，也可以通过代理模块实现，还可以通过自定义脚本实现。常用的数据收集工具有Cacti、Ganglia等。
* **数据提取模块：**此模板主要完成数据的筛选过滤和采集，将需要的数据从数据收集模块提取到监控报警模块中。可以通过数据收集模块提供的接口或自定义脚本实现数据的提取。
* **监控报警模块：**此模块主要完成监控脚本的设置、报警规则设置，报警阀值设置、报警联系人设置等，并将报警结果进行集中展现和历史记录。常见的监控报警工具有Nagios、Centreon等。
* 在了解了运维监控平台的一般设计思路之后，接下来详细介绍下如何通过软件实现这样一个智能运维监控系统。
* 下图是根据上图的设计思路形成的一个运维监控平台实现拓扑图，从图中可以看出，主要有三大部分组成，分别是数据收集模块、监控报警模块和数据提取模块。
* 其中，数据提取模块用于其他两个模块之间的数据通信，而数据收集模块可以有一台或多台数据收集服务器组成，每个数据收集服务器可以直接从服务器群组收集各种数据指标，经过规范数据格式，最终将数据存储到数据收集服务器中。
* 监控报警模块通过数据抽取模块从数据收集服务器获取需要的数据，然后设置报警阀值、报警联系人等，最终实现实时报警。报警方式支持手机短信报警、邮件报警等，另外，也可以通过插件或者自定义脚本来扩展报警方式。这样一整套监控报警平台就基本实现了。



**3、企业运维监控平台选型**

* **中小企业监控平台选择Zabbix**

Zabbix是一款综合了数据收集、数据展示、数据提取、监控报警配置、用户展示等方面的一款综合运维监控平台。

Zabbix学习入门较快，功能也很强大，是一个可以迅速用起来的监控软件，能够满足中小企业的监控报警需求，因此是中小型企业运维监控的首选平台。

但是，Zabbix当监控服务器数量较多时，会产生很多问题，如监控数据不准确、报警超时等等问题，这是因为Zabbix对服务器性能要求较高，当监控的服务器数量超过500台后，监控性能急剧下降，此时需要进行分布式监控部署，并且需要提升监控服务器的性能。

安全性方面，Zabbix客户端的agent如果故障，收集到的数据将丢失，同时Zabbix Server也是单点，可能还需要对Zabbix Server做HA保证数据的安全和监控的高可用。

* **互联网大企业监控平台选择Ganglia+Centreon**

开源监控软件组合应用+二次开发是大型互联网企业构建监控平台的一个基本策略，对于有海量服务器、多业务系统的复杂监控，没有哪个软件能独立完成企业的所有监控需求，因此，多种开源监控软件组合应用+二次开发才是监控平台的最终方向。

推荐ganglia是因为ganglia客户端软件对服务资源占用非常低，并且扩展插件非常多，监控扩展也非常容易，同时结合专业的web监控平台centreon，可以实现在数据收集、数据展示、数据提取、监控报警配置、用户展示等方面的完美配合，因此这里对海量服务器进行监控我们推荐ganglia+centreon组合。

**4、运维监控平台的历程**

这是一个经验和总结，我结合这么多年我们监控平台的演变，总结了一下不同阶段、不同机器数量，监控平台需要的构建思路和策略。

* **机器数量小于100台的阶段**

这个时期由于机器数量较少，因此，对监控的需求也很简单，监控的用途可能主要用于通知问题、快速定位与解决问题，大致总结一下，此阶段监控平台的特点如下：

1>部署简单，上手易用

2>稳定运行，不出故障

3>可进行报警，以邮件、短信等形式

基于以上特点和需求，可以使用比较流行开源的监控软件Nagios，Cacti，Zabbix，Ganglia等等。流行的开源产品文档很多，可快速上手，并且有大量的前人使用经验，遇到问题也很容易解决。

最初我们选择了nagios，因为这款软件是最早流行的，后来因为主机和服务添加不方便，切换到了zabbix上了，此阶段，zabbix应该是最好的选择。

* **机器数量200到1000的阶段**

这个阶段，由于机器数量变多，监控需求也开始变得复杂，不过主要还是用于通知、告警，发现问题，并避免同样的问题再次发生，根据这个阶段的特点，我们在这个时期主要对监控平台做了以下工作：

1>监控内容分类：由于要监控的机器很多，监控内容也随之增多，于是我们将监控根据用途不同，进行了分类，主要分为系统基础监控数据、网络监控数据和业务监控数据。

2>全覆盖式监控：将所有机器均纳入监控中，主要包含软件监控和硬件监控，硬件监控主要是监控硬件性能和故障，软件监控除了第一步提到的各种基础监控数据外，还增加了业务逻辑监控，尽可能的覆盖业务流程，通过大量自定义监控减少和去除重复的问题，保障业务稳定运行。

3>多种告警方式，确保无漏报：将所有监控根据重要程度、紧急程度进行分类，分别用邮件，微信，短信，电话等不同级别的方式进行通知，每个监控对应到不同的人，确保每个监控都有人处理，并且对于重要的业务采用持续通知的方式，不处理就一直通知。

这个阶段的难点是对告警信息的处理，由于机器越来越多，需要监控的服务也越来越多，告警信息就出现了爆发式增长，每天收到上千封报警邮件是经常的事情。 过多的邮件出现，其实就失去了告警的意义，因为我们不可能去查看每一封邮件，而这么多告警邮件中，很多都是非必要的告警，例如系统负载偶尔增高一下，就发了告警邮件，这完全是不需要的。

因此，这个阶段，主要是对监控告警策略进行配置和优化，尽量减少不必要的告警邮件，例如，对系统负载的监控，可以选择连续几次负载超过阀值，然后持续多久之后才进行告警操作，通过对告警策略的优化，告警信息大大减少，每天最多几十封，这样的话，就不会错过任何告警信息了。

* **机器数量超过1000台的阶段**

由于业务持续增长，对服务器需求越来越多，当我们的服务器超过1000台以后，监控的情况发生了变化，或者说监控出现了很多奇怪的问题，主要有如下一些：

1>告警不及时

当我们服务器超过1000台以后，我们的zabbix就经常罢工，有时候监控数据不能及时显示，有时候告警迟迟不来，特别是告警延时，这个是最恐怖的事情，线上业务7\*24小时不能出现故障，虽然监控到了异常，但是通过监控系统发出来已经是1个或者几个小时之后了，那监控还有什么意义呢，及时性是监控系统的第一要求，这个是必须要解决的问题。

如何解决这个问题呢，除了对监控进行优化，例如分布式proxy方式部署，开启zabbix主动模式，还对数据收集进行了扩展和优化，我们对基础数据的收集，抛弃了zabbix来实现，而采用ganglia，而对业务数据部分实现仍然采用zabbix完成，通过将收集数据的负载进行分担，大大减低了zabbix的负载，数据收集的准确性，及时性又恢复正常了。

2>告警系统出现了单点故障

由于服务器众多，收集的数据也飞速增长，曾经有一次，监控服务器突然意外宕机了，等系统恢复启动起来，已经是一个小时以后了，这一个小时运维就变成了睁眼瞎了，多可怕的事情。

自从发生监控系统宕机事故后，我们对监控服务器进行了分布式高可用部署，以避免单点故障，同时对监控到的数据进行远程异地备份，当监控服务器故障后，会自动切换到备用监控系统上，并且监控数据自动保存同步。

3>告警需求监控系统无法满足

业务的增加，客户对业务稳定性要求变得更加苛刻，为了保证业务系统稳定运行，业务逻辑监控需求被提出来了，业务逻辑监控就是对业务系统的运行逻辑进行监控，当业务运行逻辑故障时候，也需要进行告警，很显然，对业务逻辑的监控，没有现成的工具和代码，只能根据业务逻辑自行开发，通过提高业务逻辑接口，汇报数据等方式，我们对zabbix进行了多项二次开发，以满足对业务逻辑的监控。

最后，运维监控平台是运维工作中不可或缺的一部分，如何构建适合自己的运维监控平台，每个公司的需求不一样，每个运维面对的痛点也不尽相同，但，不管有什么需求，多少需求，万变不离其宗，有了机器上的各种监控数据，运维就能做很多事情。运维监控的路上，我们一起前行。

作者：一尘在心,来源：CSDN

<https://blog.csdn.net/zhanaolu4821/article/details/83898152>

https://blog.51cto.com/cloumn/detail/33