**Zabbix监控知识点总结**

**目录：**

**[Zabbix简介](#Zabbix简介)**

**[Zabbix运行条件](#Zabbix运行条件)**

**[Zabbix功能](#Zabbix功能)**

**[优劣势](#优劣势)**

**[组件说明](#组件说明)**

**[Zabixx的主动模式和被动模式的工作流程](#Zabixx的主动模式和被动模式的工作流程)**

**[客户端守护进程](#客户端守护进程)**

**[扩展：zabbix的监控架构](#扩展：zabbix的监控架构)**

**重点掌握：**

1. **两种工作模式**
2. **三种架构**
3. **Zabbix的一些优点和劣势**
4. **应用环境需要lamp（web环境）**

**Zabbix默认端口：10051 服务端；10050 客户端**

**Zabix自身提供了众多监控模板（比如：对http服务监控；对系统级的监控项操作（Linux，windows）对硬件进行监控（内存使用量，cpu负载）；对网络流量的监控；对磁盘的使用量监控）。**

C/S（客户端/服务端）：收集数据。

B/S（浏览器/服务器）：以web的形式在网页中显示收集数据的结果。

**一. Zabbix简介**

**Zabbix是一个基于WEB界面企业级开源分布式监控解决方案**，由C语言编写而成的底层架构（server端和agent端）。

Zabbix通过C/S模式采集数据，通过B/S模式在web端展示和配置。

**被监控端：主机通过安装agent方式采集数据，网络设备（对路由器，交换机进行监控）通过SNMP（简单网络管理协议）方式采集数据**

**Server端：通过收集SNMP和agent发送的数据，写入MySQL数据库，再通过php+apache在web前端展示**。

**二.** **Zabbix运行条件：**

Server：

**Zabbix Server需运行在LAMP**（Linux+Apache+Mysql+PHP）环境下，对硬件要求低

Agent：

目前已有的**agent基本支持市面常见的OS操作系统**，包含Linux、HPUX、Solaris、Sun、windows等

**SNMP：**

**支持各类常见的网络设备（比如：路由器，交换机）**

**三.** **Zabbix功能**

具备常见的商业监控软件所具备的功能（主机的性能监控、网络设备性能监控、数据库性能**监控**、FTP等通用协议监控、多种**告警**方式（微信报警，短信报警，邮件报警，钉钉报警）、详细的报表**图表绘制**）

支持自动发现网络设备和服务器，**支持分布式**，能集中展示、管理分布式的监控点，**扩展性强**，server提供**通用接口**，可以自己开发完善各类监控项。

**四.** **优劣势**

**优点：**

开源，无软件成本投入；

Server对设备性能要求低（实际测试环境：虚拟机Redhat EL AS5，2GCPU 1G内存，监控5台设备，CPU使用率基本保持在10%以下，内存剩余400M以上）；

支持设备多；

支持分布式集中管理；

开放式接口，扩展性强；

当监控的**item（监控项）**比较多服务器队列比较大时可以采用**被动状态**，被监控客户端主动从server端去下载需要监控的item然后取数据上传到server端。这种方式对服务器的负载比较小。

**缺点：**

**无厂家支持，出现问题解决比较麻烦**

**需在被监控主机上安装agent，所有数据都存在数据库里，产生的数据量很大,瓶颈主要在数据库**。

**四．****组件说明：**

1）**zabbix server**：负责接收agent发送的报告信息的核心组件，所有配置、统计数据及操作**数据**都由它组织进行；

2）**database storage：专用于存储所有配置信息**，以及由zabbix收集的数据；

3）web interface：zabbix的GUI接口；

4）**proxy：**可选组件，常**用于**监控节点很多的**分布式环境中**，代理server收集部分数据转发到server，可以**减轻server的压力**；

5）**agent：**部署在被监控的主机上，**负责收集主机本地数据**如cpu、内存、数据库等数据发往server端或proxy端；

**五．****Zabixx的主动模式和被动模式的工作流程；**

agentd需要安装到被监控的主机上，它负责定期收集各项数据，并发送到zabbix server端，zabbix server将数据存储到数据库中，zabbix web根据数据在前端进行展现和绘图。这里agentd收集数据分为主动和被动两种模式：

1. 代理主动和代理被动模式是争对客户端的。
2. 代理主动模式：每隔一定时间（时间可以自定义，默认30秒），客户端会主动向服务端发起连接请求，服务端收到请求后，会将监控项发给客户端程序，客户端程序收到后，根据监控项内容，收集相关的监控数据，并发回给服务端，服务端会将数据写入到数据库，并通过web界面展示。（客户端开启主动模式时，客户端没有端口号）
3. 代理被动模式：每隔一定时间，服务器主动将监控项发给客户端，客户端收集好数据后，发回给服务端，服务端会将数据写入到数据库，并通过web界面展示。（客户端端口号：10050）

**主动模式的工作效率要高。**

**六．****客户端守护进程：**

**此进程收集客户端数据**，例如cpu负载、内存、硬盘使用情况等。

**zabbix\_get**

zabbix工具，单独使用的命令，通常在server或者proxy端执行获取远程客户端信息的命令。通常用户排错。例如在server端获取不到客户端的内存数据，我们可以使用**zabbix\_get获取客户端的内容的方式来做故障排查。**

**zabbix\_sender**

zabbix工具，用于**发送数据给server或者proxy，通常用于耗时比较长的检查**。很多检查非常耗时间，导致zabbix超时。于是我们在脚本执行完毕之后，使用sender主动提交数据。

**zabbix\_server**

zabbix**服务端守护进程**。zabbix\_agentd、zabbix\_get、zabbix\_sender、zabbix\_proxy、zabbix\_java\_gateway的数据最终都是提交到server

备注：**当然不是数据都是主动提交给zabbix\_server,也有的是server主动去取数据。**

**zabbix\_proxy**

zabbix代理守护进程。功能类似server，唯一不同的是它**只是一个中转站**，它需要**把收集到的数据提交/被提交到server里。**

**zabbix\_java\_gateway**

zabbix2.0之后引入的一个功能。顾名思义：Java网关，类似agentd，但是只用于Java方面。需要特别注意的是，它**只能主动去获取数据**，而不能被动获取数据。它的数据最终会**给到server或者proxy。**

**七．扩展：zabbix的监控架构**

在实际监控架构中，zabbix根据网络环境、监控规模等 分了三种架构： server-client 、master-node-client、server-proxy-client三种 。

**1、server-client架构（适合小型公司）**

也是zabbix的最简单的架构，**监控机和被监控机**之间不经过任何代理 ，直接由zabbix server和zabbix agentd之间**进行数据交互。适用于网络比较简单，设备比较少的监控环境 。**

****2、server-proxy-client架构（中型公司）****

其中proxy是server、client之间沟通的一个桥梁，**proxy（是代理程序。没有存数据的数据库；在内存中暂时存数据）**本身没有前端，而且其本身并不存放数据，**只是将agentd发来的数据暂时存放，而后再提交给server 。**该架构经常是和master-node-client架构做比较的架构 ，**一般适用于跨机房、跨网络的中型网络架构的监控。**

**3、master-node-client架构（大型公司；最庞大、最稳定、最麻烦的一种架构）**

该架构是zabbix最复杂的监控架构，**适用于跨网络、跨机房、设备较多的大型**环境 。**每个node同时也是一个server端**，node（是服务端程序）下面可以接proxy，也可以直接接client 。**node有自已的配置文件和数据库**，其要做的是将配置信息和监控数据向master同步，master的故障或损坏对node其下架构的完整性无影响。

**Zabbix\_agent：客户端的守护进程**

**八．Grafana简介**

**Grafana是一个可视化面板（dashboard），有着非常漂亮的图形和布局展示**

**Grafana主要特性：灵活丰富的图形化选项；可以混合多种风格；支持白天和夜间模式；多个数据源。**