



链滴

Zabbix4.0 版本介绍与使用

作者: [GeekBoyDqz](#)

原文链接: <https://ld246.com/article/1567834562293>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<p></p>

<h2 id="一-监控系统简介">一、监控系统简介</h2>

<h3 id="1-监控系统介绍">1、监控系统介绍</h3>

<h4 id="1-1-什么是监控系统">1.1：什么是监控系统</h4>

监控系统性能资源的硬件或者软件

它是监控 系统资源以及性能的硬件或者软件

<h4 id="1-2-监控软件分类">1.2：监控软件分类</h4>

<h5 id="1-2-1-单一监控程序-">1.2.1：单一监控程序：</h5>

包含 win 系统的任务管理、Linux 的 top、vmstat、iostat 等

<h5 id="1-2-2-分布式监控程序-">1.2.2：分布式监控程序：</h5>

<p>Zabbix、Open-Falcon 属于分布式监控程序</p>

<p>通过这些监控系统，我们可以连接设备的繁忙程度；是否有异常进程占用资源。</p>

<p>监控硬件中通过传感器获得设备的监控信息</p>

<h4 id="1-3-为什么需要监控系统">1.3：为什么需要监控系统</h4>

在互联网时代，用户主要通过 APP、浏览器等方式来获取信息；

线上业务的稳定运行的依赖的因素很多；大到运营商，IDC，基础设施；小到 CPU，内存，应用代码；任何一个环节故障都可能影响业务稳定运行，严重的导致雪崩效应，从而给公司造成损失。

在故障发生前能及时告警并处理跟进，那么用户就不会告知到业务故障

<h4 id="1-4-监控系统功能">1.4：监控系统功能</h4>

<p>基础：数据收集

数据收集方式：客户端、snmp 协议、自定义插件</p>

<p>数据展示

图形化展示</p>

<p>告警

由告警策略与告警发送两个部分组成；通过策略监控数据，匹配规则触发告警发送操作；

告警可以通过短信、微信、邮件、电话、语音等方式发送</p>

<p>事件管理

统计事件告警信息</p>

<p>报表管理</p>

<p>认证权限</p>

<h4 id="1-5-开源监控系统">1.5: 开源监控系统</h4>

Nagios: 老牌监控系统; 1999 年发布初始版本, 可监控网络、主机等设备; 支持丰富的监控插件, 用户可根据自己实际环境定义监控。

Cacti: 老牌监控系统; 2001 年发布, 基于 SNMP 和 RRDTool 的网络流量监控分析系统; RRDTool 用来处理时间序列数据的套件, 是环形数据库

Grafana: 强大的展示页面和分析监控数据的工具, 现在 Grafana 可进行数据收集和告警设置

Ganglia: 分布式集群监控工具, 底层基于 RRDTool 实现

Prometheus: 基于 Go 语言开发的企业级监控、告警、存储的套件, 容器环境应用较多

Open-falcon: 小米公司开发的一款企业级、高可用、可扩展的开源监控解决方案

<h4 id="1-6-监控系统的趋势">1.6: 监控系统的趋势</h4>

<p>随着大数据和人工智能技术的不断普及, 会给我们的生活和工作带来新的挑战 and 机遇; </p>

智能监控:

异常检测、定位、故障预测、瓶颈分析等

<h4 id="1-7-如何选择监控系统">1.7: 如何选择监控系统</h4>

<h2 id="二-Zabbix发展介绍">二、Zabbix 发展介绍</h2>

<h3 id="1-Zabbix介绍">1、Zabbix 介绍</h3>

<h4 id="1-1-Zabbix创始人">1.1: Zabbix 创始人</h4>

1998 年 Alexei 开始着手写 zabbix;

2001 年以 GPL 协议发布了 Zabbix 1.0 alpha(阿尔法)

2001 年 Alexei 成立了 Zabbix LLC 公司; 主要负责功能研发、技术支持、培训服务、为客户定制开发监控系统

每年大约 200W 的下载量, 被翻译为了 15 种语言

<h4 id="1-2-Zabbix特点">1.2: Zabbix 特点</h4>

1、监控任何东西, monitor anything; 监控任何的设备、系统、应用、服务和资源

2、企业级, 可以无限扩展, 分布式部署, 高可用, 安全

3、跨平台的软件, 可以在 windows、Linux、unix 等操作系统上部署

<h4 id="1-3-Zabbix组成">1.3: Zabbix 组成</h4>

Zabbix 由前端、服务端、代理端、客户端、Java 监控网关几个组件组成

前端由 PHP 语言编写

服务端、代理端、客户端由 C 语言编写

Java 监控网关 JAVA 开发

<h4 id="1-4-Zabbix版本">1.4: Zabbix 版本</h4>

<p>开源软件大神</p>

Linus Torvalds: Linux 内核和 Git 创始人

Richard Stallman: GNU 开源软件创始人

<p>开源许可协议

</p>

<h3 id="2-Zabbix4-0功能介绍">2、Zabbix4.0 功能介绍</h3>

<h4 id="2-1-Zabbix数据收集">2.1: Zabbix 数据收集</h4>

Zabbix 通过 agent 客户端收集数据，客户端支持多平台部署；比如 Windows、Linux、Unix、ac 等系统

Zabbix 客户端占用系统资源很少，可以获取 CPU、内存、网卡、磁盘、日志等信息

Zabbix 支持通过 SNMP(简单网络管理协议)获取监控数据，通过 SNMP 不仅仅可以监控网络设，可以监控打印机、UPS 等

可以通过 IPMI 获取硬件的温度、风扇的转速、硬盘状态、电源电压等;IPMI：只能平台管理接口开放的硬件管理接口标准；通过 IPMI 不仅可以获取监控数据，也可以管理硬件设备，重启，关机，取硬件日志等。

Zabbix 自带的检测工具进行监控，支持 TCP,ICMP,SSH,Telnet 检测方式；

Zabbix 支持自定义监控，支持 shell、，Python、ruby、perl、powershell 任何可执行的脚本行监控数据的收集

如对监控性能要求较高，可通过加载模块的方式来实现监控，这种方式只支持 Unix、Linux 系统

默认 Zabbix 提供 URL 监控

<h4 id="2-2-Zabbix数据展示">2.2: Zabbix 数据展示</h4>

<p>Zabbix 前端提供了丰富的展示功能；支持拓扑图展示业务架构、滚动图展示监控数据</p>

<h4 id="2-3-Zabbix故障检测">2.3: Zabbix 故障检测</h4>

<p>告警策略：</p>

Zabbix 支持 8 种运算符，29 个函数可以任意组合使用，完全可以满足我们的告警阈值设置需求

Zabbix 更注重故障预警，在故障发生前通知相关负责人或者进行相关操作

Zabbix 主要通过趋势分析来判断数据未来的走势，若趋势异常则会触发告警操作

Zabbix 支持和历史数据进行数据对比分析

Zabbix 支持自定义告警触发范围，若数据走势波动超出定义的范围则触发告警

<h4 id="2-4-Zabbix告警发送">2.4: Zabbix 告警发送</h4>

<p>Zabbix 可以通过邮件、电话、短信、微信、钉钉的方式将告警信息发送给相关负责人</p>

<p>Zabbix 可以在告警触发前执行定义好的故障处理的命令和脚本先进行故障的修复</p>

<p>Zabbix 可以实现告警场景升级，如 Nginx 服务宕机后将告警信息发送给了负责人并执行了相应脚本进行尝试恢复，但是过了 5 分钟后发现业务还是未恢复，那么这时候就可以进行告警升级，将告发送领导等</p>

<p>Zabbix 支持告警关联，定义服务与服务器的告警关联

2.5: Zabbix 安全和认证</p>

<p>Zabbix 支持多种认证方式，可以是本地用户，HTTP 基础认证，LDAP 认证</p>

<p>一般企业通过 LDAP 和域控进行结合统一管理用户和密码，不同用户登录设置不同的权限</p>

<p>安全方面，Zabbix 任意两个组件之间可通过 TLS 加密传输数据；适合场景：多地域、多机房</p>
>

<blockquote>
<p>SSL

SSL 是 TLS 的前身</p>
</blockquote>
<blockquote>
<p>TLS

TLS 全称是 Transport Layer Security 传输层安全性协议；SSL3.0 的基础上建立了 TLS 1.0</p>
</blockquote>
<h4 id="2-6-Zabbix自动化">2.6: Zabbix 自动化</h4>

Zabbix 提供 2 个自定制化手段来提高技术人员高效的管理监控设备
网络扫描：通过扫描发现网段中满足条件的设备，如安装了 agent、打开了 SNMP、开启了特定服务等；满足条件的设备可以设置操作自动加入监控，应用对应的监控模板，或者执行某个处理脚本
Agent 自动注册：只要安装了 agent 的设备并且配置为主动上报模式，Zabbix 服务端就会根据户端传过来的信息执行监控相关操作
系统内部变化的元素如何进行监控？
如网卡流量、挂载的文件系统等信息，这些信息不同的设备如何实现监控？
使用 Zabbix 的 LLD 低级自动发现功能；
低级发现功能可自动发现并创建监控项，告警策略，图标

<h4 id="2-7-Zabbix-API">2.7: Zabbix API</h4>

通过 Zabbix API 可以获取监控数据，自动化配置管理，与第三方系统联动，开发自己的监控 API 等
Zabbix API 支持多种语言的调用，Shell, Python, Go 语言等

<h4 id="2-8-Zabbix4.0特性">2.8: Zabbix4.0 特性</h4>

性能提升 10%-20%
分部署网络带宽降低 80%
支持单点登入
支持 http agent

<h3 id="3-Zabbix架构组成">3、Zabbix 架构组成</h3>
<p>监控系统的组成模块包括：数据收集、汇聚、存储、展示、告警的分析和发送</p>
<h4 id="3-1-Zabbix架构组件介绍-">3.1: Zabbix 架构组件介绍：</h4>

Zabbix Agent: Zabbix 客户端，负责数据的收集上传
Zabbix Server: Zabbix 服务端，负责数据汇总、处理、告警发送等
Zabbix WEB: Zabbix 前端页面，提供了友好的展示和操作界面，负责数据的展示，监控系统的置管理，用户权限配置等
Databases: 数据和配置存储数据库，Zabbix 支持多种数据库，包括 MySQL、Oracle、DB2 等
Java GateWay: java 网关，负责通过 JVM 监控收集 Java 应用性能数据
Zabbix Proxy: Zabbix 代理，分布式部署架构会用到，主要收集设备的监控数据并将数据发送对应的 Zabbix Server

>3.2: 工作流程</h4>

<p>数据通过 Zabbix 客户端收集并发送给 Zabbix 服务端，Zabbix 服务端负责存储，数据分析，触告警等功能，用户或者管理员可以通过 Zabbix 前端页面进行数据展示和配置管理</p>

<p>大规模的业务场景</p>

设备规模大且分布在不同地域或多个不同机房，可以通过 Zabbix Proxy 实现分布式架构部署

>3.3: Zabbix Server 组成</h4>

<p>Zabbix Server 通过哪些进程实现的?

注：不同版本的 Zabbix 启动的进程名称可能不一样</p>

<p>数据收集:</p>

Poller: 主要负责收集 Server 主动拉取类型的监控数据

Trapper: 主要负责 Agent 主动上报的监控数据

Http Poller: 主要负责 URL 监控类型的数据收集

icmp pinger: 负责 ping 存活监控的数据收集(ping 存活检测)

java gateway: 负责 java 和 java gateway 通信处理的数据

java poller: 负责拉取 JMX 类型的数据

ipmo poller: 负责 IPMI 类型的监控数据

timer: 负责处理和时间有关的监控数据已经告警维护等

vmware collector: 负责收集 vmware 虚拟化环境的监控数据

unreacheable poller: 负责处理无法到达类的监控数据

<p>数据处理:</p>

preprocessing: 对监控数据进行预处理

history syncer: 负责将数据分析并保存至数据库中

housekeeper: 负责定期清理历史数据

<p>告警实现:</p>

通过 alerter 进程处理和发送，而 alerter manager 负责管理 alerter 进程

escalator: 负责处理告警过程中的各个步骤，如告警升级之类

<p>分布式通信:</p>

由 proxy poller 进行负责实现

<p>自动发现:</p>

由 discovery process 负责设备的自动发现功能

>4、Zabbix 基本术语</h3>

>4.1: 组件功能</h4>

<p></p>

>4.2: 监控收集</h4>

host: 主机，任何被监控的设备都叫主机，服务器，交换机，存储，打印机这些在 Zabbix 中都称为主机

host group: 主机组，主机的逻辑分组，如同一个机房的主机分为一组，同一个办公区的打印机

为一组

- item: 监控项, 可以理解为监控的一个指标, 如 CPU 的使用率, 负载, 网卡接收流量等
- value preprocessing: 监控项的数据预处理, 就是在存入数据库之前按照指定的规则进行预处理; 比如处理成数据的变化量, 数据的每秒变化速率, 或者是单位的转换, 从 ms 换算成 s 等
- application: 应用, 一组监控项的逻辑分组, 比如 Nginx 的监控项统一分到 Nginx 应用中
- template: 模板, 就是可以应用多个监控设备的监控集合, 包含监控项, 触发器, 图形, LLD, 报警等; 同一类的监控可以整理成模板, 从而可以重复利用, 大幅度提高监控的效率
- web scenario: web 场景, 是监控 Web 的一个或者多个 HTTP 请求, 一个场景中可以是单个 URL, 也可以是多个 URL; 比如可以将用户登入, 搜索, 点击商品详情, 加入购物车等操作放入一个场景进行监控, 当所有的步骤都成功, 该场景的监控才是成功的
- Macros: 宏, 可以认为是一个变量, 可以应用在告警, 模板等功能中

4.3: 数据展示

- graph: 图表, 可以将一个或者多个监控项的监控数据放入同一张图表中, 比如将 CPU 的用户使用率, 系统使用率, 空闲率都放入 CPU 监控图表中
- screen: 聚合图表, 就是将多个 graph 聚在一张监控大屏中, 比如将 CPU, 网卡, 内存, IO 这图表放在一起, 就组成了主机的聚合图表
- maps: 网络拓扑图, Zabbix 还支持拓扑图展示监控, 并在拓扑图中加入对应的监控指标, 当监控项异常时, 拓扑图也会显示异常, 很方便定位问题
- Slide shows: 幻灯片播放, 每隔一段时间轮流播放多个聚合图表, 可以实现将主机, 网络, 存储, 缓存, 数据库的各个聚合图表进行轮流播放

4.4: 告警相关

- Trigger: 触发器, 是告警策略的设置, 可以分别设置正状态和异常状态的触发器; 比如 CPU 持续 5 分钟使用率超过 80% 就触发告警
- event: 事件, 比如告警状态的变化, 自动化发现策略生效, 客户端注册成功等等这些都是事件比如当 CPU 5 分钟使用率为 85% 就会触发上面的策略, 触发器的状态会从 OK 变成 Problem, 这就是一个事件
- problem: 异常状态
- OK: 正常状态
- action: 操作, 是根据事件以及条件定义的一系列动作, 当 CPU 告警发生时, 可以触发一个操作, 这个操作是发送告警信息给管理员, 让他及时处理
- escalation: 升级, 是在一个动作内执行的操作, 告警升级操作就是通过这个实现, 比如上面的告警信息发生就是也给 escalation
- media: 媒介, 指告警通知的方式, 包括短信, 邮件, 微信等
- notification: 通知, 是关于事件的消息, 通过指定的媒介发送给用户
- remote command: 远程命令, 指预先定义的, 在指定的条件下会被执行的命令; CPU 告警产生时除了发送信息外, 也可以执行命令, 比如将这台设备上的服务自动重启下
- Maintenance: 维护模式, 就是升级系统或者维护的时间段, 该时间段可以不发送告警, 或者发送告警, 但是不计入服务的可用性; 如上面是因为系统维护或者发布业务发送的 CPU 告警, 那么告警操作可以不发送

4.5: 认证和权限

- User: 用户, 可以是内部用户也可以是 LDAP 用户
- User Group: 用户组, 多个用户组成的逻辑组, 比如运维组, 研发组, 产品组
- permission: 权限, 用户或者用户组对监控设备的访问权限是不同的, 可以是读写权限, 只读权限, 或者没有任何权限
- User Type: 用户类型, Zabbix 中提供了三种用户类型, 普通用户, 管理员, 超级管理员;
- 普通用户: 只能查看对应的监控设备
- 管理员: 可以编辑有权限的访问的设备
- 超级管理员: 可以管理监控系统中的所有配置, 一般只有监控系统管理员是超级用户

<h2 id="三-Zabbix安装部署">三、Zabbix 安装部署</h2>

<h3 id="1-Zabbix二进制安装">1、Zabbix 二进制安装</h3>

<h4 id="1-1-资源要求">1.1：资源要求</h4>

<p></p>

<h4 id="1-2-环境准备">1.2：环境准备</h4>

<p>VM ware 虚拟机

CentOS 7.5

Zabbix 4.0</p>

<h5 id="1-2-1-环境配置">1.2.1：环境配置</h5>

<pre><code class="highlight-chroma">设置阿里镜像源

[root@zabbix-server ~]# wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-7.repo

[root@zabbix-server ~]# yum install epel-release

关闭防火墙

[root@zabbix-server ~]# sed -ri 's/SELINUX=enforcing/SELINUX=disabled/' /etc/selinux/config

[root@zabbix-server ~]# setenforce 0

[root@zabbix-server ~]# getenforce

Permissive

[root@zabbix-server ~]# systemctl stop firewalld;systemctl disable firewalld

安装常用开发组件

[root@zabbix-server ~]# yum groups install "Development Tools"

[root@zabbix-server ~]# yum groups info "Development Tools"

增加Zabbix镜像源

[root@zabbix-server ~]# rpm -ivh https://repo.zabbix.com/zabbix/4.3/rhel/7/x86_64/zabbix-release-4.3-3.el7.noarch.rpm

安装命令补齐工具

[root@zabbix-agent ~]# yum -y install bash-completion

[root@zabbix-agent ~]# yum -y install bash-completion-extras

</code></pre>

<h5 id="1-2-2-安装Zabbix-Server和Frontend">1.2.2：安装 Zabbix Server 和 Frontend</h5>

<pre><code class="highlight-chroma">搜索包名

[root@zabbix-ser


```
er ~]# yum search all zabbix
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">安装zabbix
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# yum install zabbix-server-mysql
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# yum install zabbix-web-mysql
</span></span></code></pre>
<h5 id="1-2-3-安装配置MySQL">1.2.3: 安装配置 MySQL</h5>
<ul>
<li>这里我安装最新 MySQL57 版本</li>
</ul>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">① 下载 MySQL 官方的 Yum Repository
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
</span></span><span class="highlight-line"><span class="highlight-cl">② 安装 yum Repos
tory
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# rpm -ivh mysql57-community-release-el7-11.noarch.rpm
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">③ 安装 MySQL
务端
</span></span><span class="highlight-line"><span class="highlight-cl">查看 mysql server
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# yum repolist enabled | grep "mysql.*-community.*"
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# yum repolist all | grep mysql
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">安装
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# yum install mysql-community-server
</span></span><span class="highlight-line"><span class="highlight-cl">④ 启动 mysql 服务
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# systemctl start mysqld.service
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# systemctl status mysqld.service
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">⑤ 数据库密码配置
</span></span><span class="highlight-line"><span class="highlight-cl">查看登陆密码
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# cat /var/log/mysqld.log | grep 'password'
</span></span><span class="highlight-line"><span class="highlight-cl">登陆
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# mysql -uroot -p
</span></span><span class="highlight-line"><span class="highlight-cl">到这里 MySQL 就
全安装完成了
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">重设密码
</span></span><span class="highlight-line"><span class="highlight-cl">解决密码过短问题
</span></span><span class="highlight-line"><span class="highlight-cl">mysql> set glo
al validate_password_policy=0;
</span></span><span class="highlight-line"><span class="highlight-cl">再次设置密码
</span></span><span class="highlight-line"><span class="highlight-cl">mysql> ALTER
```

USER 'root'@'localhost' IDENTIFIED BY '123';

⑥ 数据导入

创建数据库

[root@zabbix-server ~]# mysql -uroot -p

mysql> create database zabbix character set utf8 collate utf8_bin;

mysql> grant all privileges on zabbix.* to zabbix@localhost identified by '123456';

导入数据结构

[root@zabbix-server ~]# zcat /usr/share/doc/zabbix-server-mysql*/create.sql.gz | mysql -uzabbix -p zabbix

</code></pre>

<h5 id="1-2-4-配置Zabbix-Serve">1.2.4: 配置 Zabbix Serve</h5>

修改数据库相关的信息

<pre><code class="highlight-chroma">[root@zabbix-server ~]# vim /etc/zabbix/zabbix_server.conf

92:DBHost=localhost

101:DBName=zabbix //数据库名称

117:DBUser=zabbix //用户

126:DBPassword=23456 //用户密码

</code></pre>

启动 Zabbix Server

<pre><code class="highlight-chroma">[root@zabbix-server ~]# systemctl start zabbix-server.service

[root@zabbix-server ~]# systemctl status zabbix-server.service

</code></pre>

检查日志是否有报错

<pre><code class="highlight-chroma">[root@zabbix-server ~]# more /var/log/zabbix/zabbix_server.log

</code></pre>

配置 Zabbix 前端

<pre><code class="highlight-chroma">修改PHP相关参数

[root@zabbix-server ~]# vim /etc/php.ini

max_execution_time = 300 //默认30,Zabbix要求为300

memory_limit = 128M //内存限制

```

</span></span><span class="highlight-line"><span class="highlight-cl">post_max_size =
6M      //默认8M改为16M
</span></span><span class="highlight-line"><span class="highlight-cl">upload_max_filesi
e = 2M
</span></span><span class="highlight-line"><span class="highlight-cl">max_input_time =
300      //默认60
</span></span><span class="highlight-line"><span class="highlight-cl">max_input_vars =
0000      //去除注释，默认1000改为10000
</span></span><span class="highlight-line"><span class="highlight-cl">always_populate_r
w_post_data = -1      //去除注释，默认On改为-1
</span></span><span class="highlight-line"><span class="highlight-cl">date.timezone =
sia/Shanghai //去除注释，默认空，更为上海时区
</span></span></code></pre>
<ul>
<li>启动 httpd</li>
</ul>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">[root@zabbix-server ~]# systemctl start httpd.service
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# systemctl enable httpd.service
</span></span><span class="highlight-line"><span class="highlight-cl">[root@zabbix-ser
er ~]# systemctl status httpd.service
</span></span></code></pre>
<ul>
<li>浏览器访问</li>
</ul>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">http://192.168.66.40/zabbix/
</span></span></code></pre>
<p><b
>
<br>
<br>
<br>
<br>
</p>
<ul>
<li>
<p>Zabbix 前端准备好了! 默认用户名为 Admin, 密码为 zabbix.<br>
</p>
</li>
<li>
<p>本机安装 Zabbix-agent</p>
</li>
</ul>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">[root@zabbix-server ~]# yum install zabbix-agent

```

[root@zabbix-server ~]# systemctl start zabbix-agent.service
</code></pre>

这样就可以监控本机的情况了

查看监控数据

<h2 id="四-问题总结">四、问题总结</h2>
<h3 id="1-Web前端图表中文乱码解决">1、Web 前端图表中文乱码解决</h3>
<h4 id="方式1-">方式 1: </h4>

<p>问题原因

zabbix 前端默认使用的字体跟中文不兼容导致

</p>

<p>解决方案

从 Windows 字体当中拷贝字体到 zabbix-server 端的 web 前端的/usr/share/zabbix/assets/fonts 目录下, 然后修改 zabbix 前端字体配置就可以了

</p>

<pre><code class="highlight-chroma">[root@zabbix-server ~]# vim /usr/share/zabbix/include/defines.inc.php
</code></pre>
<p>

</p>
<h4 id="方式2-yum安装字体">方式 2: yum 安装字体</h4>
<pre><code class="highlight-chroma">[root@zabbix-server ~]# yum -y install wqy-microhei-fonts
[root@zabbix-server ~]# cp /usr/share/fonts/wqy-microhei/wqy-microhei.ttc /usr/share/fonts/dejavu/DejaVuSans.ttf
</code></pre>

浏览器检查

