

# 系统学习 Zabbix 系统监控（二）

## Linux 系统监控



# 目录

5 windows 系统硬件信息获取 .....	4
5.1 CPU 信息获取 .....	4
5.1.1 CPU 型号 .....	5
5.1.2 CPU 颗数 .....	6
5.1.3 CPU 核数 .....	6
5.1.4 CPU 逻辑核与线程 .....	7
5.1.5 监控结果 .....	8
5.2 内存信息获取 .....	8
5.2.1 创建监控项 .....	8
5.2.2 监控结果 .....	8
5.3 操作系统信息 .....	9
5.3.1 创建监控项 .....	9
5.3.2 监控结果 .....	10
5.4 服务器信息 .....	10
5.4.1 创建监控项 .....	10
5.4.2 监控结果 .....	11
6 Linux 系统状态监控 .....	11
6.1 Linux 部署 zabbix_agent .....	12
6.2 Linux 系统 CPU 监控 .....	12
6.3 Linux 系统内存监控 .....	13
6.4 Linux 磁盘使用监控 .....	14
6.5 Linux 磁盘读写监控 .....	15
6.5 Linux 网卡状态监控 .....	18
7 Linux 系统硬件信息获取 .....	18
7.1 CPU 信息获取 .....	19
7.1.1 CPU 型号 .....	19
7.1.2 CPU 颗数 .....	19
7.1.3 CPU 核数 .....	20
7.1.4 CPU 逻辑核与线程 .....	20
7.1.5 监控结果 .....	21
7.2 Memory 内存信息获取 .....	21
7.2.1 Memory 内存主板支持 .....	21
7.2.2 Memory 内存信息 .....	22

<b>7.3 Disk 硬盘信息获取</b> .....	22
<b>7.4 OS 操作系统信息获取</b> .....	23
<b>7.5 OS 服务器信息获取</b> .....	24



**【导读】** 本系列文章介绍了 zabbix 基本概念及其特点，阐述 zabbix 系统环境搭建与基础安装，如何实现对各类操作系统、硬件设备、应用软件监控告警。本文为系列第二篇，内容包括 Windows 系统硬件信息获取、Linux 系统监控、Linux 系统硬件信息获取。前文点此：[系统学习 Zabbix 系统监控（一）Windows 系统监控 | 运维进阶](#)

## 5 windows 系统硬件信息获取

本文通过在 OS 操作系统层面上，主要获取 windows 服务器下 CPU 信息、内存信息、硬盘信息、操作系统、服务器信息。信息获取的实现方式是通过在 windows 系统下部署自定义 bat 脚本，执行脚本获取数据，再将获取的信息发送给 zabbix 服务端，zabbix 界面创建相应的监控项，触发器等，最终将信息展示出来。监控项内容如下：

CPU 信息：型号、个数、内核数、逻辑核、CPU 健康状态，及状态告警。

内存信息：容量、个数、厂商、型号、序列号；主板支持内存最大容量和个数。

硬盘信息：厂商、个数、容量、序列号、接口类型、硬盘健康状态，及状态告警。

操作系统信息：主机名、操作系统版本、运行时长、系统线程数、系统时间。

服务器信息：品牌、型号、序列号。

说明：一些特殊数据需要实现监控，例如 CPU 温度、硬盘状态、Raid 卡状态、风扇转速等，windows 没有提供检测硬件温度组件，需要借助第三方工具如 IPMI tools，fan-speed 等，也可以使用 IPMI 协议等其他方法来丰富 windows 系统硬件监控项，对于虚拟机并不适用，此时，推荐使用服务器的管理口，如 HPE 服务器的 iLO、DEll 服务器的 iDRAC，联想服务器 XCC 等开启 snmp 功能，再进行 zabbix 配置，实现对服务器硬件全面监控。如果服务器未配置管理口，当然不能适用。

### 5.1 CPU 信息获取

应用集：CPU 硬件

cpu 信息主要有：CPU 型号、CPU 颗数、CPU 核数、CPU 逻辑核与线程（超线程，一般是核心数的 2 倍）

监控项配置：

名称 ▲	触发器	键值
CPU 型号		cpu_hardware_model
CPU 核数		cpu_hardware_core
CPU 硬件状态	触发器 1	cpu_status
CPU 逻辑核		system.cpu.num[]
CPU 颗数		cpu_hardware_number

### 5.1.1 CPU 型号

需要编写程序对 CPU 型号进行提取，相关配置如下：

1、在 C:\zabbixconf\zabbix\_agentd.win.conf 文件中：自定义程序开关设置为开启，UnsafeUserParameters=1 并添加监控项：

# CPU 型号

UserParameter=cpu\_hardware\_model,C:\zabbixscriptcpu\_hardware\_model.bat

Zabbix 界面添加监控项：

监控 key 值：

cpu\_hardware\_model

2、程序目录为

C:\zabbixscriptcpu\_hardware\_model.bat

```

1. @echo off
2.
3. ::
4.
5. wmic cpu get name|C:\zabbix\gawk-3.1.6-1-bin\bin\awk.exe "NR==2 { print }" > C:\zabbix\script\cpu_har
   hardware_model.txt
6.
7. type C:\zabbix\script\cpu_hardware_model.txt

```

3、监控项配置：

名称

类型

键值

信息类型

## 5.1.2 CPU 颗数

需要编写程序对 CPU 型号进行提取，相关配置如下：

1、在 C:\zabbixconf\zabbix\_agentd.win.conf 文件中：自定义程序开关设置为开启，UnsafeUserParameters=1 并添加监控项：

# CPU 型号

UserParameter=cpu\_hardware\_number,C:\zabbixscript\cpu\_hardware\_number.bat

2、程序目录为：

C:\zabbixscript\cpu\_hardware\_number.bat

```
1. @echo off
2.
3. ::
4.
5. wmic cpu get DeviceID | findstr /V "DeviceID" | findstr /V "^$" | C:\zabbix\UnxUtils\usr\local\wbin\grep.exe
   "CPU" | C:\zabbix\UnxUtils\usr\local\wbin\wc.exe -l | C:\zabbix\UnxUtils\usr\local\wbin\sed.exe "s/ //g" >
   C:\zabbix\script\cpu_hardware_number.txt
6.
7. type C:\zabbix\script\cpu_hardware_number.txt
```

3、监控项配置：

The screenshot shows the Zabbix monitoring item configuration interface. The configuration is as follows:

- 名称 (Name): CPU 颗数
- 类型 (Type): Zabbix 客户端
- 键值 (Key): cpu\_hardware\_number
- 信息类型 (Info type): 字符
- 更新间隔 (Update interval): 1h
- 自定义时间间隔 (Custom time interval):

类型	间隔	期间	动作
灵活	调度	50s	1-7,00:00-24:00
- 历史数据保留时长 (History retention): 30d
- 查看值 (View value): 不变
- 新的应用集 (New application set): (empty)
- 应用集 (Application set): -无-, CPU 状态, CPU 硬件

## 5.1.3 CPU 核数

需要编写程序对 CPU 核数进行提取，相关配置如下：

1、在 C:\zabbixconf\zabbix\_agentd.win.conf 文件中：自定义程序开关设置为开启，UnsafeUserParameters=1 并添加监控项：

# CPU 核数 一颗 CPU 的核心数

UserParameter=cpu\_hardware\_core,C:\zabbixscript\cpu\_hardware\_core.bat

2、程序目录为：

C:\zabbixscript\cpu\_hardware\_core.bat

```
1. @echo off
2.
3. wmic cpu get NumberOfCores | C:\zabbix\gawk-3.1.6-1-bin\bin\awk.exe "NR==2 { print }" > C:\zabbix\script\cpu_hardware_core.txt
4.
5. type C:\zabbix\script\cpu_hardware_core.txt
```

3、监控项配置：

名称	<input type="text" value="CPU 核数"/>
类型	<input type="text" value="Zabbix 客户端"/>
键值	<input type="text" value="cpu_hardware_core"/> <input type="button" value="选择"/>
信息类型	<input type="text" value="字符"/>

### 5.1.4 CPU 逻辑核与线程

使用 zabbix 自带 key，监控 key 值：system.cpu.num[]

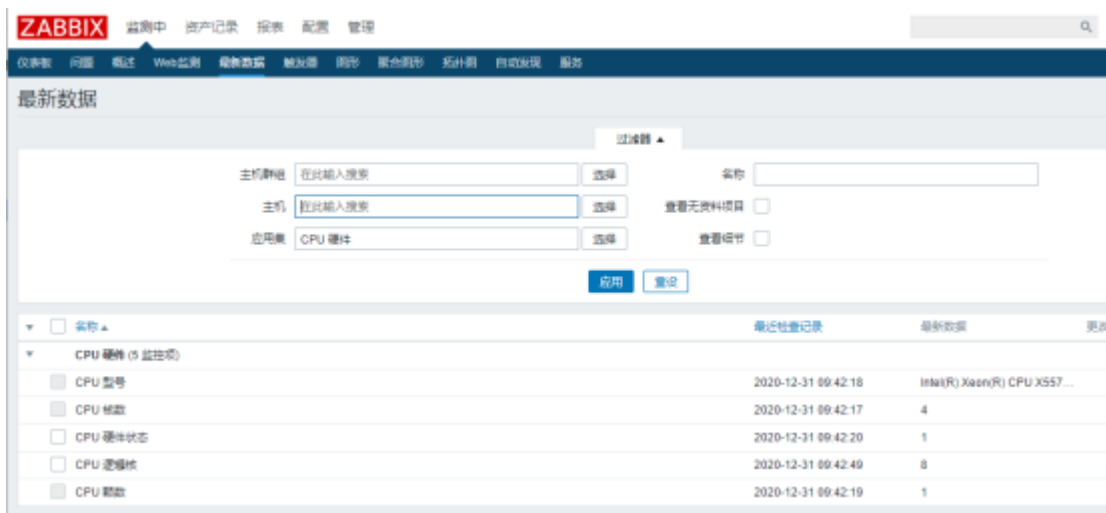
说明：type 可用值，online (默认值), max 范例：system.cpu.num

经实践检查此处的 key 值为逻辑核心，CPU 逻辑核心、线程（超线程，一般是核心数的 2 倍）windows 系统下管理处理器，看到的数量。

监控项配置：

名称	<input type="text" value="CPU 逻辑核"/>
类型	<input type="text" value="Zabbix 客户端"/>
键值	<input type="text" value="system.cpu.num[online]"/> <input type="button" value="选择"/>
信息类型	<input type="text" value="文本"/>

## 5.1.5 监控结果



## 5.2 内存信息获取

memory 内存信息：包括序号、制造商、容量、序列号、型号、速率

memory 内存主板支持：最大容量，最大槽位数

### 5.2.1 创建监控项

名称	触发器	键值
Memory 内存主板支持		memory_biso_support_info
memory 内存信息		memory_hardware_info

创建：memory 内存信息、memory 内存主板支持

键值：memory\_biso\_support\_info、memory\_hardware\_info

应用集：Memory 内存硬件

### 5.2.2 监控结果

1、监测中 > 最新数据 > Memory 内存硬件 > memory 内存信息



时间戳	值					
2020-12-31 09:42:39	BankLabel	Capacity	Manufacturer	PartNumber	SerialNumber	Speed
	BANK02	8589934592	Micron	36KSF1G72PDZ-1G1M1	B063F6B1	800
	BANK03	8589934592	Micron	36KSF1G72PDZ-1G1M1	BA63F6B1	800
	BANK05	4294967296	Hynix	HMT151R7TFR4C-H9	39104623	800
	BANK06	8589934592	Micron	36KSF1G72PDZ-1G1M1	4163F6B1	800
	BANK07	4294967296	Hynix	HMT151R7TFR4C-H9	60101623	800
	BANK08	8589934592	Micron	36KSF1G72PDZ-1G1M1	7B63F6B1	800

可以看到，序列依次为：内存序号、容量、制造商、型号、序列号、速率。

2、监测中 > 最新数据 > Memory 内存硬件 > memery 内存主板支持

时间戳	值	
2020-12-31 09:42:38	MaxCapacity	MemoryDevices
	201326592	18

第一列为主板支持最大容量，第二列为主板支持最大槽位数。

## 5.3 操作系统信息

OS 操作系统信息：主机名、操作系统版本、运行时长、统线程数、系统时间。

其中操作系统版本是自定义程序获取，主机名、运行时长、统线程数、系统时间是 zabbix 自带监控模板，自带模板直接套用。

### 5.3.1 创建监控项

名称	触发器	键值
系统时间		system.localtime[local]
操作系统版本		os_version
绝对秒	触发器 1	system.localtime[]
系统线程数		perf_counter[\2\250]
系统运行时长	触发器 1	system.uptime
主机名	触发器 1	system.hostname[]

创建：系统时间、操作系统版本、绝对秒、系统线程数、系统运行时长、主机名

键值：system.localtime[local]、os\_version、system.localtime[]、perf\_counter[2250]、system.uptime、system.hostname[]

应用集：OS 操作系统

### 5.3.2 监控结果

监测中 > 最新数据 > OS 操作系统

OS 操作系统 (5 监控项)		
<input type="checkbox"/> 主机名	2020-12-31 11:42:50	NETWORKER
<input checked="" type="checkbox"/> 操作系统版本	2020-12-31 11:42:43	Microsoft Windows...
<input type="checkbox"/> 系统时间	2020-12-31 11:46:52	2020-12-31,11:46:...
<input type="checkbox"/> 系统运行时长	2020-12-31 11:46:56	15 天, 01:16:24
<input type="checkbox"/> 绝对秒	2020-12-31 11:42:51	1609386151

## 5.4 服务器信息

监控服务器信息：品牌、型号、序列号。

### 5.4.1 创建监控项

名称	触发器	键值
OS 服务器 序列号		os_device_serialnumber
OS 服务器 型号		os_device_mode
OS 服务器 品牌		os_device_manufacturer

创建：OS 服务器序列号、OS 服务器型号、OS 服务器品牌

键值：os\_device\_serialnumber、os\_device\_mode、os\_device\_manufacturer

应用集：OS 服务器信息

## 5.4.2 监控结果

监测中 > 最新数据 > OS 服务器信息

The screenshot shows a web interface for monitoring OS server information. At the top, there is a search bar with 'OS 服务器信息' and a '选择' button. Below it are '应用' and '默认' buttons. The main content is a table with columns: '名称', '最近检查记录', '最新数据', and '更改'. The table lists three items under 'OS 服务器信息 (3 监控项)': 'OS 服务器 品牌' (IBM), 'OS 服务器 型号' (System x3650 M2 -...), and 'OS 服务器 序列号' (99F8756).

名称	最近检查记录	最新数据	更改
OS 服务器 品牌	2020-12-31 10:42:40	IBM	
OS 服务器 型号	2020-12-31 10:42:41	System x3650 M2 -...	
OS 服务器 序列号	2020-12-31 10:42:42	99F8756	

# 6 Linux 系统状态监控

Linux 系统监控，监控项原则上能利用 zabbix 提供模板就尽量使用，zabbix 提供不了的就编写 shell 脚本，这样就省去大部分代码编写时间，减少工作量。状态信息原则上已使用率为主要观察点，不需要再监控剩余率，Linux 系统监控模板，与 window 系统监控应用集、监控项命名保持统一，监控模板如下所示：

应用集：CPU 状态、CPU 硬件、Disk 硬盘、Disk 磁盘状态、Memory 内存状态、Memory 内存硬件、OS 操作系统、OS 服务器信息、agent 模板链接。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/745340240341011314>