

zabbix数据库表研究分析

@凡竹 (dingzfang@163.com)

1.监控获得的数据存放的表

zabbix中存储历史数据的表是以history开头的，在zabbix2.0.8中主要的表如下：

- (1)history表，该表主要存放监控到的数据类型为：数字型的浮点数据，即Numeric(float)
- (2)history_log表，该表主要存放监控到的数据类型为：日志类型，即log
- (3)history_str表，该表主要存放监控到的数据类型为：符号类型，即Character
- (4)history_text表，该表主要存放监控到的数据类型为：文字类型，即text
- (5)history_uint表，该表主要存放监控到的数据类型为：数字型的无符号类型，即Numeric(unsigned)
- (6)history_str_sync表，该表用于zabbix分布式部署时存放监控到的数据类型为：符号类型，即Character
- (7)history_sync表，该表用于zabbix分布式部署时存放监控到的数据类型为：数字型的浮点数据，即Numeric(float)
- (8)history_uint_sync表，该表用于zabbix分布式部署时存放监控到的数据类型为：数字型的无符号类型，即Numeric(unsigned)

这些表中公共的字段有：

itemid：表示监控项所对应的id

clock：表示取得监控项的值所对应的时间，注意clock为unix time

value：表示取得的监控项的值

不同类型的数据存放在不同的表中，可以在配置主机的监控项时指定对应的数据类型，进而该监控项的值就会存入相应的表中

 监测中
  资产记录
  报表
  配置
  管理

[主机组](#) | [模板](#) | [主机](#) | [维护](#) | [Web](#) | [动作](#) | [筛选](#) | [简报片显示](#) | [拓扑图](#) | [发现](#) | [IT服务](#)

[历史记录: 最新数据](#) » [历史记录](#) » [主机组的配置](#) » [主机的配置](#) » [项目的配置](#)

项目的配置

[主机清单](#)
主机: [hadoop](#)
受监测中




[应用集 \(1\)](#)
[项目 \(17\)](#)
[触发器 \(0\)](#)
[图形 \(0\)](#)
[发现规则 \(0\)](#)

项目

主机

hadoop

选择

名称

类型

Zabbix代理(被动式)

键值

选择

主机接口

没有找到接口

数据类型

数字的 (无正负)

数字的 (无正负)

数字的 (浮点)

符号

日志

文字

单位

使用自定义倍数

☐

1

数据更新间隔(秒)

30

弹性区间

间隔

期间

动作

并无定义的弹性区间

主机组 | 模板 | **主机** | 维护 | Web | 动作 | 筛选 | 简报片显示 | 拓扑图 | 发现 | IT服务

项目的配置

项目

主机 选择

名称

类型 ▼

键值 选择

主机接口 没有找到接口

数据类型 ▼

数据类型 ▼

单位 ▼

使用自定义倍数 ☐

数据更新间隔(秒)

弹性区间	间隔	期间	动作
并无定义的的弹性区间			

2.自动发现功能对应的表



监测中



资产记录



报表



配置



管理

主机组 | 模板 | 主机 | 维护 | Web | 动作 | 筛选 | 简报片显示 | 拓扑图 | 发现 | IT服务

历史记录: 历史记录 » 主机组的配置 » 主机的配置 » 项目的配置 » 发现的配置

发现规则的配置

发现规则

名称

由代理节点进行发现

没有代理节点

IP范围

192.168.0.1-255

延迟(秒计)

3600

检查

新的

设备唯一性标准

☒

IP地址

已启用

☒

存档

克隆

删除

取消

[主机组](#) | [模板](#) | [主机](#) | [维护](#) | [Web](#) | [动作](#) | [筛选](#) | [简报片显示](#) | [拓扑图](#) | [发现](#) | [IT服务](#)

发现规则的配置

名称	<input type="text"/>
由代理节点进行发现	<input type="checkbox"/> 没有代理节点 <input type="button" value="▼"/>
IP范围	<input type="text" value="192.168.0.1-255"/>
延迟(秒)	<input type="text" value="3600"/>
检查	<input type="button" value="新的"/>
设备唯一性标准	<input checked="" type="radio"/> IP地址
已启用	<input checked="" type="checkbox"/>

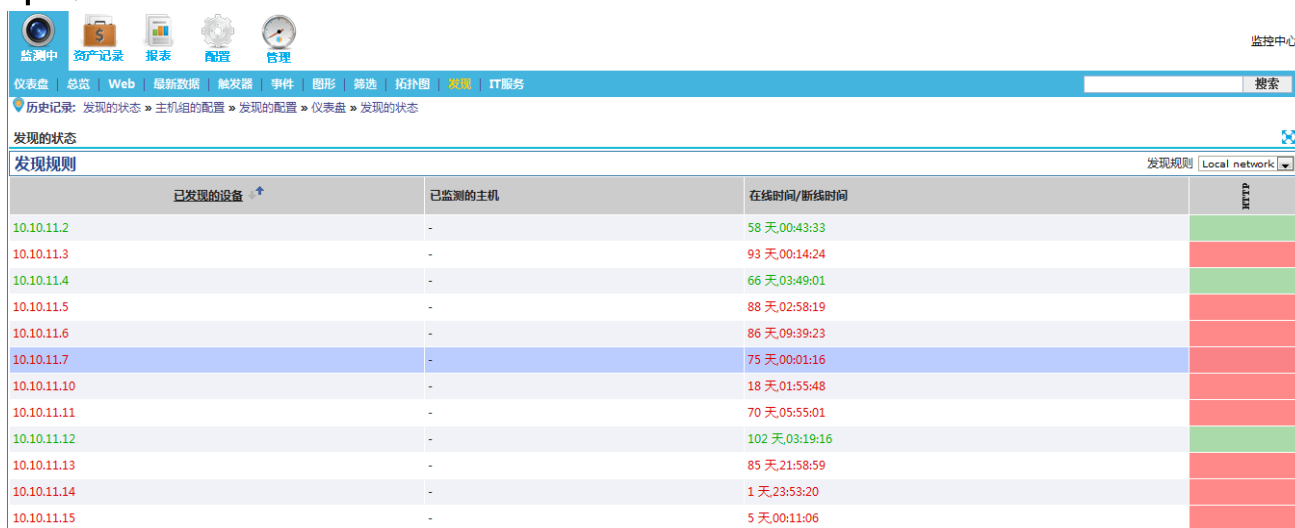
存档 克隆 删除 取消

(2)dchecks表：该表用于存放用户配置的检查方式，一个发现规则可以对应多个检查方式，如下：



(3)dhosts表：该表用于存放发现的主机及其状态，用于首页显示

(4)dservices表，该表用于存放发现的服务及其状态，界面如下：



已发现的设备	已监测的主机	在线时间/断线时间	FTP
10.10.11.2	-	58 天00:43:33	Green
10.10.11.3	-	93 天00:14:24	Red
10.10.11.4	-	66 天03:49:01	Green
10.10.11.5	-	88 天02:58:19	Red
10.10.11.6	-	86 天09:39:23	Red
10.10.11.7	-	75 天00:01:16	Red
10.10.11.10	-	18 天01:55:48	Red
10.10.11.11	-	70 天05:55:01	Red
10.10.11.12	-	102 天03:19:16	Green
10.10.11.13	-	85 天21:58:59	Red
10.10.11.14	-	1 天23:53:20	Red
10.10.11.15	-	5 天00:11:06	Red

3.web监测功能所对应的表

(1)

4.主机配置所对应的表

(1) hosts表：该表主要存放当前正在监控的主机服务器的基本配置信息，主要包括tab页的主机和IPMI信息。

- hostid:被监控服务器id

- host:被监控服务器名称
- ip:被监控服务器的ip
- status字段的取值有：0 (HOST_STATUS_MONITORED) 表示启用监控；1 (HOST_STATUS_NOT_MONITORED) 表示未启用监控；3 (HOST_STATUS_TEMPLATE) 表示模板；5 (HOST_STATUS_PROXY_ACTIVE) 表示proxy主动式监控；6 (HOST_STATUS_PROXY_PASSIVE) 表示proxy被动式监控

(2)hosts_templates表：该表主要存放当前正在监控的主机服务器与模板的关联关系，主要是tab页中的模板

(3)hostmacro表：该表主要存放当前正在监控的主机服务器所定义的一些宏，主要是tab页中的宏

(4)host_inventory表：该表主要存放当前正在监控的主机服务器所对应的资产记录，主要是tab页中的资产记录

(5)interface表：该表主要存放监控接口的信息，如代理接口，Jmx接口，SNMP接口等



5.监控项配置所对应的表

(1)items表：该表主要存放监控项的配置信息。

- Itemid:监控项的id
- hostid:该监控项监控的服务器id,对应于hosts表的hostid
- description:监控项的描述,如(Combined CPU Utilization %)
- status:0(Item_STATUS_ACTIVE),可用；
1 (ITEM_STATUS_DISABLED) ，不可用；

3 (ITEM_STATUS_NOTSUPPORTED) , 不支持

-
-
- define('ITEM_STATUS_ACTIVE',0);
- define('ITEM_STATUS_DISABLED',1);
- define('ITEM_STATUS_NOTSUPPORTED',3);
-
- type:
- define('ITEM_TYPE_ZABBIX',0);
- define('ITEM_TYPE_SNMPV1',1);
- define('ITEM_TYPE_TRAPPER',2);
- define('ITEM_TYPE_SIMPLE',3);
- define('ITEM_TYPE_SNMPV2C',4);
- define('ITEM_TYPE_INTERNAL',5);
- define('ITEM_TYPE_SNMPV3',6);
- define('ITEM_TYPE_ZABBIX_ACTIVE',7);
- define('ITEM_TYPE_AGGREGATE',8);
- define('ITEM_TYPE_HTTPTEST',9);
- define('ITEM_TYPE_EXTERNAL',10);
- define('ITEM_TYPE_DB_MONITOR',11);
- define('ITEM_TYPE_IPMI',12);
- define('ITEM_TYPE_SSH',13);
- define('ITEM_TYPE_TELNET',14);
- define('ITEM_TYPE_CALCULATED',15);
- define('ITEM_TYPE_JMX',16);

value_type:

```
define('ITEM_VALUE_TYPE_FLOAT',0);
define('ITEM_VALUE_TYPE_STR',1);
define('ITEM_VALUE_TYPE_LOG',2);
define('ITEM_VALUE_TYPE_UINT64',3);
define('ITEM_VALUE_TYPE_TEXT',4);
```

data_type:

```
define('ITEM_DATA_TYPE_DECIMAL',0);
define('ITEM_DATA_TYPE_OCTAL',1);
define('ITEM_DATA_TYPE_HEXADECIMAL',2);
define('ITEM_DATA_TYPE_BOOLEAN',3);
```

其对应的界面如下：



项目

主机: INSPUR-J5ZX-FK [选择]

名称: [输入框]

类型: Zabbix代理(被动式) [下拉菜单]

键值: [输入框] [选择]

主机接口: 10.10.12.106 : 10050 [下拉菜单]

数据类型: 数字的(无正负) [下拉菜单]

数据类型: 十进制数字 [下拉菜单]

单位: [输入框]

使用自定义倍数: ☐ [输入框: 1]

数据更新间隔(秒): 30 [输入框]

弹性区间

新的弹性区间	间隔(秒计)	期间	动作
	50	1-7,00:00-24:00	添加

保留历史(日计): 90 [输入框]

保留趋势(日计): 365 [输入框]

保存值: 不变 [下拉菜单]

显示值: 不变 [下拉菜单] [显示值对应]

新的应用集: [输入框]

应用集: 无 [下拉菜单]

(2)applications表：该表主要存放应用集的信息。其对应的界面如下：

应用集

主机: ubuntu1 [选择]

名称: [输入框]

[存档] [取消]

(3)items_applications表：该表主要存放监控项与应用集的关联关系

(4)triggers表:该表主要存放触发器的配置信息

(5)trigger_depends表：该表主要存放触发器的依赖关系

触发器的配置

主机: hadoop 受监测中 [图标] [图标] [图标] 应用集 (1) 项目 (17) 触发器 (0) 图形 (0) 发现规则 (0)

触发器 [切换] 相依关系 [切换]

名称: [输入框]

表达式: [输入框] [添加]

表达式构造 [链接]

多重问题事件产生: ☐

描述: [输入框]

URL: [输入框]

严重性: 未分类 [切换] 消息 [切换] 警告 [切换] 一般严重 [切换] 严重 [切换] 灾难 [切换]

已启用: ☒

(6)graphs表：该表主要存放监控图形的配置信息

- graphid:监控图的id
- name:监控图的名称,描述

- graphtype:0对应曲线图,2对应饼图
- flags字段的取值有：
 - 0 (ZBX_FLAG_DISCOVERY_NORMAL)，表示一般的item；
 - 1 (ZBX_FLAG_DISCOVERY)，表示一个低级的发现规则；
 - 2 (ZBX_FLAG_DISCOVERY_CHILD)，表示item的原型；
 - 4 (ZBX_FLAG_DISCOVERY_CREATED)，表示由发现规则创建的项

(7)graphs_items表：该表主要存放监控图形与监控项的关联关系

- graphid:监控图id
- itemid:监控图对应的监控项id
- drawtype:

(8)发现的信息也存放在items表中，由字段flags来区分，其中 flags=0表示监控项， flags=1表示发现 discovery， flags=2,flags=4表示发现的原型项目

名称	项目	触发器	图形	键值	间隔	类型	状态	错误
<input type="checkbox"/> Mounted filesystem discovery	项目原型 (5)	触发器类型 (2)	图形原型 (1)	vfs.fs.discovery	3600	Zabbix代理(被动式)	已启用	✓
<input type="checkbox"/> Network interface discovery	项目原型 (2)	触发器类型 (0)	图形原型 (1)	net.if.discovery	3600	Zabbix代理(被动式)	已启用	✓

(9)item_discovery表：

6.模板配置所对应的表

(1)模板的信息也存放在hosts表中，由字段status来区分， status=3表示模板， status=0表示启用对主机的监控， status=1表示停止对主机的监控

7.主机组

(1)groups表：该表主要存放主机组的基本配置信息

(2)hosts_groups表：该表主要存放主机组与主机(包括模板)之间的关联

8.维护配置所对应的表

(1)maintenances表：该表主要存放维护的配置信息

(2)timeperiods表：该表主要存放期间的配置信息，注意该表的字段month，dayofweek存放的值是二进制值转换为数字后的值，如下代码(maintenance.php)：

```
// days of week
if (!isset($new_timeperiod['dayofweek'])) {
$dayofweek = (!isset($new_timeperiod['dayofweek_su'])) ?
'0' : '1';
$dayofweek .= (!isset($new_timeperiod['dayofweek_sa'])) ?
'0' : '1';
$dayofweek .= (!isset($new_timeperiod['dayofweek_fr'])) ?
'0' : '1';
$dayofweek .= (!isset($new_timeperiod['dayofweek_th'])) ?
'0' : '1';
$dayofweek .= (!isset($new_timeperiod['dayofweek_we']))
? '0' : '1';
$dayofweek .= (!isset($new_timeperiod['dayofweek_tu'])) ?
'0' : '1';
$dayofweek .= (!isset($new_timeperiod['dayofweek_mo']))
? '0' : '1';
$new_timeperiod['dayofweek'] = bindec($dayofweek);
}
```

```
// months
if (!isset($new_timeperiod['month'])) {
$month = (!isset($new_timeperiod['month_dec'])) ? '0' :
'1';
$month .= (!isset($new_timeperiod['month_nov'])) ? '0' :
'1';
$month .= (!isset($new_timeperiod['month_oct'])) ? '0' : '1';
```



```

$month .= (!isset($new_timeperiod['month_sep'])) ? '0' : '1';
$month .= (!isset($new_timeperiod['month_aug'])) ? '0' :
'1';
$month .= (!isset($new_timeperiod['month_jul'])) ? '0' : '1';
$month .= (!isset($new_timeperiod['month_jun'])) ? '0' : '1';
$month .= (!isset($new_timeperiod['month_may'])) ? '0' :
'1';
$month .= (!isset($new_timeperiod['month_apr'])) ? '0' : '1';
$month .= (!isset($new_timeperiod['month_mar'])) ? '0' :
'1';
$month .= (!isset($new_timeperiod['month_feb'])) ? '0' : '1';
$month .= (!isset($new_timeperiod['month_jan'])) ? '0' : '1';
$new_timeperiod['month'] = bindec($month);
}

```

(3)maintenances_hosts表：该表主要存放维护与主机的关联关系

(4)maintenances_groups表：该表主要存放维护与主机组的关联关系

(5)maintenances_windows表：该表主要存放维护与期间的关联关系

9.Web监控所对应的表

(1)httpstest表：该表主要存放web监控方案的配置信息

(2)httpstep表：该表主要存放web监控步骤的配置信息

对应的界面如下：

WEB监控的配置

方案 步骤

应用集

名称

认证

用户

密码

数据更新间隔(秒)

代理

变数

主动式 ☒

方案的步骤

名称

URL

贴上

超时

要求的字串

要求的状态码

(3)httpstestitem表：该表主要存放web监控方案所对应的监控项，该表的type字段是指监控项的数据类型，可以根据该字段的值将监控到的数据存放在不同的以history开头的表中

(4)httpstepitem表：该表主要存放web监控方案的步骤所对应的监控项，该表的type字段是指监控项的数据类型，可以根据该字段的值将监控到的数据存放在不同的以history开头的表中

10.动作action的配置所对应的表

(1)actions表：该表主要存放动作action的配置信息，注意该表的字段evaltype的取值来自条件的配置信息中的计算方式：

计算方式 (A) 和 (B)

条件	标示	名称	动作
(A)		维护状态 不在 "维护"	移除
(B)		触发器值 = "问题"	移除

新的触发条件

[添加](#)

(2)conditions表：该表主要存放条件的配置信息

(3)operations表：该表主要存放操作的配置信息，从***到***，

持续时间，操作类型和计算方式等

(4)opmessage_grp表：该表主要存放操作类型为送出消息时送出消息所关联到的用户组

(5)opmessage_usr表：该表主要存放操作类型为送出消息时送出消息所关联到的用户

(6)opmessage表：该表主要存放操作类型为送出消息时送出消息的主题，内容，以及发送消息的媒介

(7)opconditions表：该表主要存放操作执行的条件

(8)opcommand表：该表主要存放当操作类型为远程命令时存放执行命令的配置信息

(9)opcommand_grp表：该表主要存放当操作类型为远程命令时远程执行命令的主机组

(10)opcommand_hst表：该表主要存放当操作类型为远程命令时远程执行命令的主机

(11)opgroup表：该表主要存放当操作类型为添加到主机组时（事件源为发现）所关联到的主机组

(12)optemplate表：该表主要存放当操作类型为与模板关联时（事件源为发现）所关联到的模板

动作的配置

动作	条件	操作
名称	<input type="text"/>	
默认操作步骤停留时间	3600 (最少60秒)	
默认标题	{TRIGGER.STATUS}: {TRIGGER.NAME}	
默认消息	<div>Trigger: {TRIGGER.NAME} Trigger status: {TRIGGER.STATUS} Trigger severity: {TRIGGER.SEVERITY} Trigger URL: {TRIGGER.URL}</div> <div>Item values: 1 {ITEM.NAME1} / {HOST.NAME1}</div>	
恢复消息	<input type="checkbox"/>	
已启用	<input checked="" type="checkbox"/>	
<div>存档</div> <div>取消</div>		

11.报表功能所对应的表：

(1)trends表：该表主要存储了汇总后的监控数据,以对HTTP GET REQUEST/S的监控项,每条数据汇总了一小时的监控数据计算得出,对应于history表

- itemid:监控项的id,对应于items表的itemid
- clock:该条汇总后的监控信息的开始时间
- value_min:汇总数据最小值
- value_avg:汇总数据平均值
- value_max:汇总数据最大值

(2)trends_uint表：该表主要存储汇总后的无符号型数据，对应于history_uint表

12.事件与告警所对应的表

(1)events表：当服务器的状况达到预定义的通知或报警条件时,会增加一条事件,写入events表

- eventid:事件的id
- clock:事件发生的时间戳
- source表示事件来源，其值有三个：
define('EVENT_SOURCE_TRIGGERS',0);define('EVENT_SOURCE_DISCOVERY',1);define('EVENT_SOURCE_AUTO_REGISTRATION',2);
- object表示对象的类型，其值有0,,1,2,3.define('EVENT_OBJECT_TRIGGER',0);define('EVENT_OBJECT_DHOST',1);define('EVENT_OBJECT_DSERVICE',2);
- value :
-
- define('TRIGGER_VALUE_FALSE',0);
- define('TRIGGER_VALUE_TRUE',1);
- define('TRIGGER_VALUE_UNKNOWN', 2); // only in "events" table
- value_changed:
-
- define('TRIGGER_VALUE_CHANGED_NO',0);
- define('TRIGGER_VALUE_CHANGED_YES', 1);
-
- objectid表示对象id，其值有三个：分别对应不同的来源事件的id值，如事件来源为trigger，则该objectid值为triggerid

(2>alerts表：主要用于存放告警信息

- alertid:报警的id
- eventid:报警对应的事件id
- clock:报警发生的时间戳
- subject:报警主题
- message:报警内容

13.筛选的配置

(1)screens表：主要存放筛选的配置信息

筛选的配置

筛选

名称

列数

1

行

1

存档

取消

(2)screens_items表：主要存放筛选的每个筛选配置项

筛选格子配置

资源

图形

图形名称

选择

宽

500

高

100

横向对齐

左侧

中

右

纵向对齐

顶部

居中

底部

列跨幅

1

行宽度

1

动态监控项

☐

存档

删除

取消

14.简报的配置

(1)slideshows表：主要存放简报的基本配置

名称

默认延迟(秒)

30

简报片

筛选

延迟

动作

添加

存档

取消

(2)slides表：主要存放存放简报片的配置，一个简报包含多个简报片

名称	<input type="text"/>		
默认延迟(秒计)	<input type="text" value="30"/>		
简报片	<div> <div> <div>筛选</div> <div>延迟</div> <div>动作</div> </div> <div> <div>1</div> <div>ubuntu-server-cpu</div> <div>15</div> <div>移除</div> </div> </div>		
	<div>添加</div>		

15.拓扑图的配置

(1)sysmaps表：主要存放拓扑图的基本配置

名称

Local network

宽

1000

高

1000

背景图片

没有图片

自动的图示对应

<手册>

显示显示对应

图标高亮

☒

触发器状态上的标记组件改变

☒

展开单一问题

☒

进阶标示

☐

图标标签类型

组件名称

图标标签位置

左侧

问题显示

仅未了解的

URLs

名称	URL	组件
<input type="text"/>	<input type="text"/>	主机

添加

保存

克隆

删除

取消

(2) sysmap_url表：主要存放拓扑图中每个组件所关联到的url

(3)sysmaps_elements表：主要存放拓扑图的每个元素

编辑拓扑图组件

类型

主机

标示

{HOST.NAME}

{HOST.CONN}

标示位置

底部

主机

ubuntusto1

选择

图标

自动的图示选择

默认

Zabbix_server_3D_(128)

问题

默认

维护

停用的

默认

默认

坐标

X: 586 Y: 136

URLs

名称	URL
<input type="text"/>	<input type="text"/>

添加

应用

移除

关闭

其中字段elementtype的取值有：

- 0 (SYSMAP_ELEMENT_TYPE_HOST) ；
- 1 (SYSMAP_ELEMENT_TYPE_MAP) ；
- 2 (SYSMAP_ELEMENT_TYPE_TRIGGER) ；
- 3 (SYSMAP_ELEMENT_TYPE_HOST_GROUP) ；
- 4 (SYSMAP_ELEMENT_TYPE_IMAGE) ；

字段elementssubtype的取值有：

- 0 (SYSMAP_ELEMENT_SUBTYPE_HOST_GROUP) ；
- 1 (SYSMAP_ELEMENT_SUBTYPE_HOST_GROUP_ELEMENTS) ；

- (4)sysmap_element_url表：主要存放元素所关联到的url
- (5)sysmaps_links表：主要存放元素之间的链接
- (6)sysmaps_link_triggers表：主要存放当某个触发器被触发时，元素之间链接的显示

16.IT服务的配置

(1)services表：主要存放服务的配置信息

IT服务的配置

服务	相依关系	时间
名称	ubuntu	
上层服务	root	更改
状态计算演算法	问题, 如果至少一个下层有一个问题	
计算SLA, 可接受的SAL(%计)	<input checked="" type="checkbox"/> 99.9000	
触发器		选择
排序次(0->999)	0	
存档 取消		

(2)services_links表：主要存放服务之间的依赖关系

服务	相依关系	时间																				
依赖于	<table border="1"> <thead> <tr> <th>服务</th> <th>软</th> <th>触发器</th> <th>动作</th> </tr> </thead> <tbody> <tr> <td>zabbix-server</td> <td><input type="checkbox"/></td> <td>Lack of available memory on server Zabbix server</td> <td>移除</td> </tr> <tr> <td>ubuntusto1</td> <td><input type="checkbox"/></td> <td>Free disk space is less than 20% on volume /</td> <td>移除</td> </tr> <tr> <td>ssw</td> <td><input checked="" type="checkbox"/></td> <td>web-baidu-trigger-fail</td> <td>移除</td> </tr> <tr> <td colspan="4">添加</td> </tr> </tbody> </table>	服务	软	触发器	动作	zabbix-server	<input type="checkbox"/>	Lack of available memory on server Zabbix server	移除	ubuntusto1	<input type="checkbox"/>	Free disk space is less than 20% on volume /	移除	ssw	<input checked="" type="checkbox"/>	web-baidu-trigger-fail	移除	添加				
服务	软	触发器	动作																			
zabbix-server	<input type="checkbox"/>	Lack of available memory on server Zabbix server	移除																			
ubuntusto1	<input type="checkbox"/>	Free disk space is less than 20% on volume /	移除																			
ssw	<input checked="" type="checkbox"/>	web-baidu-trigger-fail	移除																			
添加																						
存档 取消																						

(3)services_times表：主要存放服务时间的配置

服务	相依关系	时间								
服务次数	<table border="1"> <thead> <tr> <th>类型</th> <th>间隔</th> <th>备注</th> <th>动作</th> </tr> </thead> <tbody> <tr> <td>在线时间</td> <td>日 22:30 - 周六 22:30</td> <td></td> <td>移除</td> </tr> </tbody> </table>	类型	间隔	备注	动作	在线时间	日 22:30 - 周六 22:30		移除	
类型	间隔	备注	动作							
在线时间	日 22:30 - 周六 22:30		移除							
新的服务时间	<table border="1"> <tbody> <tr> <td>在线时间</td> <td><input type="text"/></td> </tr> <tr> <td>自从</td> <td>日 <input type="text"/> 时间 <input type="text"/></td> </tr> <tr> <td>直到</td> <td>日 <input type="text"/> 时间 <input type="text"/></td> </tr> <tr> <td colspan="2">添加</td> </tr> </tbody> </table>	在线时间	<input type="text"/>	自从	日 <input type="text"/> 时间 <input type="text"/>	直到	日 <input type="text"/> 时间 <input type="text"/>	添加		
在线时间	<input type="text"/>									
自从	日 <input type="text"/> 时间 <input type="text"/>									
直到	日 <input type="text"/> 时间 <input type="text"/>									
添加										

(4)service_alarms表：服务在某个时间的告警值

17.管理->一般的配置

- (1)config表：存放图形化使用者接口的配置，触发器严重性，触发器显示选项，其他参数，工作时间，清理器
- (2)images表：存放图片的配置信息
- (3)valuemap表与mappings表：存放值对应的配置
- (4)globalmacro表：存放宏的配置
- (5)globalnars表：
- (6)regexps表和expressions表：存放正则表达式的配置
- (7)icon_map表和icon_mapping表：存放图标对应关系

18.管理->节点管理

(1)nodes表与node_cksum表

19.管理->认证

(1)config表

20.管理->用户

(1)usrgrp表：存放用户组的配置

(2)users表：存放用户的配置

(3)users_groups表：存放用户与用户组的关联

(4)user_history表：存放用户的访问历史记录

(5)media表：存放用户关联的告警媒介



新的示警媒体

类型: Email

收件人:

当活动时: 1-7,00:00-24:00

用此如果示警度

- ☒ 未分类
- ☒ 消息
- ☒ 警告
- ☒ 一般严重
- ☒ 严重
- ☒ 灾难

状态: 已启用

添加 取消

21.管理->告警媒体类型

(1)media_type表：存放告警媒体的类型

示警媒体

描述:

类型: 电子邮件

SMTP服务器: localhost

SMTP HELO: localhost

SMTP电邮: zabbix@localhost

已启用: ☒

存档 取消

22.管理->脚本

(1)scripts表：存放脚本的配置信息

23.管理->审计

(1)auditlog表：

(2)auditlog_details表：

24管理->告警

(1>alerts表：存放告警的记录

status字段的取值为:0(ALERT_STATUS_NOT_SENT),表示未发送；1 (ALERT_STATUS_SENT)，表示已经发送，且发送成

功；2（ALERT_STATUS_FAILED），表示发送失败
alerttype字段的取值为：0（ALERT_TYPE_MESSAGE），表示发送消息；1（ALERT_TYPE_COMMAND），表示调用命令
retries字段表示失败重试次数

25其他

(1)acknowledges表：存放对触发器和事件的确认消息

警报了解: BULK ACKNOWLEDGE



(2)autoreg_host表：存放自动注册主机的信息

(3)escalations表：

(4)functions表：存放函数的信息

(5)graph_discovery

(6)graph_theme

(7)help_items:

(8)housekeeper:

(9)ids表：最大号表

(10)item_discovery

(11)profiles表：存放用户的一些个性化的配置信息，如首页上的常用图形，常用筛选，常用拓扑图等

```
$db_profiles = DBselect(  
'SELECT p.value_id,p.source'.  
' FROM profiles p'.  
' WHERE p.userid='.CWebUser::$data['userid'].  
' AND p.idx='.zbx_dbstr($idx).  
' ORDER BY p.profileid'  
);
```

type字段的取值有：0（PROFILE_TYPE_UNKNOWN），表示不知道的类型；1（PROFILE_TYPE_ID），表示ID类型；

2（PROFILE_TYPE_INT），表示int类型；

3（PROFILE_TYPE_STR），表示str类型；

4（PROFILE_TYPE_ARRAY_ID），表示array id类型；

5（PROFILE_TYPE_ARRAY_INT），表示array int类型；

6 (PROFILE_TYPE_ARRAY_STR) , 表示array str类型

(12)proxy_autoreg_host:

(13)proxy_dhistory:

(14)proxy_history

(15)rights:存放用户组对应的权限, 如图



rightid:表示唯一权限ID

groupid:表示用户组ID

permission:表示权限, 取值为0表示拒绝访问, 1表示拥有写权限 (暂时保留), 2表示拥有读权限, 3表示读写权限

id:存放主机组的id

(16)sessions:存放用户的会话信息

(17)trigger_discovery

events表:

(1)value字段的取值为: 0 (TRIGGER_VALUE_FALSE), 表示触发器没有被触发, 数据正常; 1 (TRIGGER_VALUE_TRUE), 表示触发器被触发, 数据异常, 此时异常可分等级 (triggers表中的等级字段为priority);

2 (TRIGGER_VALUE_UNKNOWN), 表示触发器状态不确定, 仅仅出现在events表中。该字段代表事件的是否有问题的状态

(2)source字段的取值为: 0 (EVENT_SOURCE_TRIGGERS), 表示触发器产生的事件;

1 (EVENT_SOURCE_DISCOVERY), 表示发现产生的事件;

2 (EVENT_SOURCE_AUTO_REGISTRATION), 表示自动注册产生的事件

(3)object字段的取值为: 0 (EVENT_OBJECT_TRIGGER), 表示trigger; 1 (EVENT_OBJECT_DHOST), 表示发现的主机;

2 (EVENT_OBJECT_DSERVICE), 表示发现的服务; 3, ~~~

(4) acknowledged字段的取值为:

0 (EVENT_NOT_ACKNOWLEDGED), 表示事件未确认;

1 (EVENT_ACKNOWLEDGED) 表示事件已经确认

(5) value_changed字段的取值为：

0 (TRIGGER_VALUE_CHANGED_NO) ，标志值没有发生改变；

1 (TRIGGER_VALUE_CHANGED_YES) ，表示值发生改变

(6) clock：表示事件产生的时间戳（毫秒）

(7) objectid:表示对应不同object的id值

triggers表：

status:取值为0 (TRIGGER_STATUS_ENABLED) ，表示启用触发器；1 (TRIGGER_STATUS_DISABLED) ，表示禁用触发器

value:0(TRIGGER_VALUE_FALSE),表示未触发；

1 (TRIGGER_VALUE_TRUE) ，表示触发；

2 (TRIGGER_VALUE_UNKNOWN) ，表示未知，仅仅在events表中

value_flags:0(TRIGGER_VALUE_FLAG_NORMAL),表示正常；

1 (TRIGGER_VALUE_FLAG_UNKNOWN) ，表示未知

priority:0(TRIGGER_SEVERITY_NOT_CLASSIFIED),表示未分类；

1 (TRIGGER_SEVERITY_INFORMATION) ，表示消息；

2 (TRIGGER_SEVERITY_WARNING) ，表示警告；

3 (TRIGGER_SEVERITY_AVERAGE) ，表示一般严重；

4 (TRIGGER_SEVERITY_HIGH) ，表示严重；

5 (TRIGGER_SEVERITY_DISASTER) ，表示灾难

Data cache statistics. Cache - one of values (modes: all, float, uint, str, log, text), history (modes: pfree, total, used, free), trend (modes: pfree, total, used, free), text (modes: pfree, total, used, free)

Configuration cache statistics. Cache - buffer (modes: pfree, total, used, free).'