

Zabbix. Monitoring the CPU temperature of a Windows machine

We will use an active Zabbix agent to monitor the processor temperature. I am sure there is no need to explain the importance of monitoring the CPU temperature.

NOTE 1.

How to install Python 2.7 is described here: <https://www.mihanik.net/tihaja-ustanovka-python-2-7/>

NOTE 2.

Zabbix agent (client) installation method is described here: <https://www.mihanik.net/ustanovka-zabbix-klienta/>

1. Collecting temperature data using a script.

Everything related to measuring the temperature of the processor (s) will be located in the directory **C:\Zabbix\scripts\OHMR**

Скрипт написан на **Python 2.7**, он собирает данные по температуре процессора (процессоров) и формирует 2 файла:

- **C: \ zabbix \ scripts \ OHMR \ cpu0.txt - temperature of the first processor**
- **C: \ zabbix \ scripts \ OHMR \ cpu1.txt - temperature of the second processor.**

NOTE 3.

If there is no second processor in the computer, then the cpu1.txt file will always have zero temperature.

NOTE 4.

To obtain the processor temperature, OpenHardwareMonitor (its console version) will be used. The project page can be found here: <https://github.com/openhardwaremonitor/openhardwaremonitor>.

The fact is that you cannot find out the actual processor temperature using WMI. To find out the current temperature, you need to read data from the processor registers, to read data from the processor registers, you need to work from the zero protection ring, and to get into the zero ring, you need to write a driver ... Here OpenHardwareMonitor does all this rough work for us. We just have to analyze what he will give us.

The script itself is saved under the name: **C:\zabbix\scripts\OHMR\OHMR.py**.

Text of scripts:

```
# -*- coding: utf-8 -*-  
  
import os
```

```

import sys
import ctypes

#
# Функция проверки наличия прав администратора
# Вход: нет
# Выход: true - есть права администратора,
#        false - нет прав администратора
#
def is_admin():
    try:
        return ctypes.windll.shell32.IsUserAnAdmin()
    except:
        return False

#####
# Start of program
#####
def main(argv=None):

    # Проверяем наличие прав админа.
    if not is_admin():
        sys.exit ('Not enough permissions to run the script !!!')

    # Устанавливаем кодировку по умолчанию.
    reload(sys)
    sys.setdefaultencoding('utf8')

    # Устанавливаем имена файлов, где будет храниться температура процессоров
    cpu0 = 'C:/zabbix/scripts/OHMR/cpu0.txt'
    cpu1 = 'C:/zabbix/scripts/OHMR/cpu1.txt'

    # Получаем результат работы OHMR
    OHMR = []
    OHMR =
os.popen("C:\zabbix\scripts\OHMR\OpenHardwareMonitorReport.exe").read().splitlines()

    # Вычисляем температуру 0-го и, возможно, 1-го процессора

    n0 = 0
    sum0 = 0

    n1 = 0
    sum1 = 0

    for i in OHMR:
        if (':' in i) and ('(/intelcpu/0/temperature/' in i):
            t = i.split()
            n0 = n0 + 1
            sum0 = sum0 + int(t[8])

            if (':' in i) and ('(/intelcpu/1/temperature/' in i):
                t = i.split()
                n1 = n1 + 1
                sum1 = sum1 + int(t[8])

    if n0>0:
        sum0 = sum0 / n0

    if n1>0:
        sum1 = sum1 / n1

    # Сохраняем температуру 0-го процессора в файл
    f = open(cpu0,'w')
    try:
        f.write(str(sum0))
    except Exception:
        pass
    finally:

```

```

        f.close()

# Сохраняем температуру 1-го процессора в файл
f = open(cpu1, 'w')
try:
    f.write(str(sum1))
except Exception:
    pass
finally:
    f.close()

if __name__ == "__main__":
    sys.exit(main())

```

2. Install the script on the system.

It is better to run the script every 5-10 minutes. I usually do this every 10 minutes - not too often, but not too rarely.

You can schedule script execution using the Windows Scheduler. The scheduler task can be created manually, or using a bat file.

I usually do this with the help of a "batch file", I gave the name to my batch file **INSTALL_OHMR_py.bat** .

Text of script:

```

@echo off

Rem Предполагаем, что на Windows XP скрипт запускается администратором.
Rem Для более старших систем это неверно.

rem Получаем версию ОС
ver | find "5.1."

rem Windows XP ?
If %errorlevel%==0 (
    rem Пропускаем проверку админских прав
    GOTO SKIPADMIN
)

SET HasAdminRights=0

FOR /F %i IN ('WHOAMI /PRIV /NH') DO (
    IF "%i"=="SeTakeOwnershipPrivilege" SET HasAdminRights=1
)

IF NOT %HasAdminRights%==1 (
    ECHO .
    ECHO Not enough permissions to run the script !!!
    ECHO .
    GOTO END
)

:SKIPADMIN

```

```

rem Получаем версию ОС
rem Windows XP ?

ver | find "5.1."

If %errorlevel%==0 (
rem Windows XP
    SHTASKS /Create /RU "NT AUTHORITY\SYSTEM" /SC HOURLY /ST 00:00:00 /TN
"OHMR0" /TR "python \"C:\zabbix\scripts\OHMR\OHMR.py\"
    SHTASKS /Create /RU "NT AUTHORITY\SYSTEM" /SC HOURLY /ST 00:10:00 /TN
"OHMR1" /TR "python \"C:\zabbix\scripts\OHMR\OHMR.py\"
    SHTASKS /Create /RU "NT AUTHORITY\SYSTEM" /SC HOURLY /ST 00:20:00 /TN
"OHMR2" /TR "python \"C:\zabbix\scripts\OHMR\OHMR.py\"
    SHTASKS /Create /RU "NT AUTHORITY\SYSTEM" /SC HOURLY /ST 00:30:00 /TN
"OHMR3" /TR "python \"C:\zabbix\scripts\OHMR\OHMR.py\"
    SHTASKS /Create /RU "NT AUTHORITY\SYSTEM" /SC HOURLY /ST 00:40:00 /TN
"OHMR4" /TR "python \"C:\zabbix\scripts\OHMR\OHMR.py\"
    SHTASKS /Create /RU "NT AUTHORITY\SYSTEM" /SC HOURLY /ST 00:50:00 /TN
"OHMR5" /TR "python \"C:\zabbix\scripts\OHMR\OHMR.py\"
) else (
rem HE Windows XP
    SHTASKS /Create /RU "NT AUTHORITY\SYSTEM" /SC DAILY /ST 00:00 /RI 10
/DU 24:00 /TN "OHMR" /TR "python \"C:\zabbix\scripts\OHMR\OHMR.py\" /RL
HIGHEST /F
)

del /f /q C:\zabbix\scripts\OHMR\cpu0.txt
del /f /q C:\zabbix\scripts\OHMR\cpu1.txt

python C:\zabbix\scripts\OHMR\OHMR.py

:END

EXIT /B

```

3. We form a template in Zabbix.

At the same time, we add a simple template to Zabbix.

First, I will give a description of the template in pictures.

Create a template named: **Active Computer – Python – CPU Temperature...**

Группы узлов сети | **Шаблоны** | Узлы сети | Обслуживание | Действия | Корреляция событий | Обнаружение | Услуги

Шаблоны

Все шаблоны / Active Computer - Python - CPU T... | Группы элементов данных 1 | Элементы данных 2 | Триггеры 8 | Графики 1 | Комплексные экраны | Правила обнаружения | Веб-сценарии

Шаблон | Присоединенные шаблоны | Теги | Макросы

* Имя шаблона:

Видимое имя:

* Группы:
начните печатать для поиска

Описание:

Item group: CPU

Группы узлов сети | **Узлы сети** | Шаблоны | Обслуживание | Действия | Корреляция событий | Обнаружение | Услуги

Группы элементов данных

Все шаблоны / Active Computer - Python - CPU T... | Группы элементов данных 1 | Элементы данных 2 | Триггеры 8 | Графики 1 | Комплексные экраны | Правила обнаружения | Веб-сценарии

* Имя:

Data items:

- **CPU0 Temperature** – `vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt]`
- **CPU1 Temperature** – `vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt]`

0 элементов | | | | | | | | | | |

Имя	Описание	Ссылка	Тип	Состояние	Оформление
Ссылка на элемент	Действие	Адрес координат	Ссылка на элемент	Ссылка на элемент	Ссылка на элемент
Ссылка на элемент	Действие	Адрес координат	Ссылка на элемент	Ссылка на элемент	Ссылка на элемент
Ссылка на элемент	Действие	Адрес координат	Ссылка на элемент	Ссылка на элемент	Ссылка на элемент

Имя:

Описание:

Ссылка на элемент:

Адрес координат:

Тип:

Состояние:

Оформление:

Элементы данных

Элемент данных Предобработка

* Имя

Тип

* Ключ

Тип информации

Единица измерения

* Интервал обновления

* Период хранения истории

* Период хранения динамики изменений

Отображение значения

Новая группа элементов данных

Группы элементов данных

Заполнение поля инвентаря узла сети

Описание

Активировано

Группы узлов сети | Шаблоны | **Узлы сети** | Обслуживание | Действия | Корреляция событий | Обнаружение | Услуги

Элементы данных

Все шаблоны / Active Computer - Python - CPU T... | Группы элементов данных 1 | **Элементы данных 2** | Триггеры 8 | Графики 1 | Комплексные экраны | Правила обнаружения | Веб-сценарии

Элемент данных | Предобработка

Имя: CPU1 Temperature

Тип: Zabbix агент (активный)

Ключ: vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt]

Тип информации: Числовой (целое положительное)

Единица измерения:

Интервал обновления: 300s

Период хранения истории: 90d

Период хранения динамики изменений: 365d

Отображение значения: Как есть

Новая группа элементов данных:

Группы элементов данных: -Нет-, CPU

Заполнение поля инвентаря узла сети: -Нет-

Описание:

Активировано

Triggers:

- Information{HOST.NAME}. CPU temperature is over 50 degrees { Active Computer – Python – CPU
Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last()}>=50 and { Active Computer – Python – CPU
Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last()}<60
- Information{HOST.NAME}. CPU temperature is over 50 degrees { Active Computer – Python – CPU
Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last()}>=50 and { Active Computer – Python – CPU
Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last()}<60
- Warning {HOST.NAME}. CPU temperature is over 60 degrees 60{ Active Computer – Python – CPU
Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last()}>=60 and { Active Computer – Python – CPU
Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last()}<70
- Warning {HOST.NAME}. CPU temperature is over 60 degrees { Active Computer – Python – CPU
Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last()}>=60 and { Active Computer – Python – CPU
Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last()}<70
- Average {HOST.NAME}. CPU temperature is over 70 degrees { Active Computer – Python – CPU
Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last()}>=70 and

- { Active Computer – Python – CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last()}<80
- Average {HOST.NAME}. CPU temperature is over 70 degrees { Active Computer – Python – CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last()}>=70 and { Active Computer – Python – CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last()}<80
- High {HOST.NAME}. CPU temperature is over 80 degrees { Active Computer – Python – CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last()}>=80
- High {HOST.NAME}. CPU temperature is over 80 degrees { Active Computer – Python – CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last()}>=80

The screenshot shows the Zabbix Triggers configuration interface. At the top, there are navigation tabs: Группы узлов сети, Шаблоны, Узлы сети, Обслуживание, Действия, Корреляция событий, Обнаружение, Услуги. The main title is "Триггеры". Below it, there are several tabs: Все шаблоны / Active Computer - Python - CPU T..., Группы элементов данных 1, Элементы данных 2, Триггеры 3, Графики 1, Комплексные экраны, Правила обнаружения, Веб-сценарии. A "Фильтр" button is on the right.

The configuration area includes fields for "Группы узлов сети", "Узлы сети", "Имя", "Важность", "Статус", "Состояние", and "Значение". There are also buttons for "Выбрать", "Добавить", "Создать", "Равно", "Изменить", "Удалить", "Inherited", "Обнаружен", and "With dependencies".

Below the configuration area is a table of triggers:

Важность	Имя	Выражение	Состояние	Тип
Информация	{HOST.NAME}. Температура процессора выше 50 градусов	(Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last())>=50 and (Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last())<60	Деактивировано	
Информация	{HOST.NAME}. Температура процессора выше 50 градусов	(Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last())>=50 and (Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last())<60	Деактивировано	
Предупреждение	{HOST.NAME}. Температура процессора выше 60 градусов	(Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last())>=60 and (Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last())<70	Активировано	
Предупреждение	{HOST.NAME}. Температура процессора выше 60 градусов	(Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last())>=60 and (Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last())<70	Активировано	
Средняя	{HOST.NAME}. Температура процессора выше 70 градусов	(Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last())>=70 and (Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last())<80	Активировано	
Средняя	{HOST.NAME}. Температура процессора выше 70 градусов	(Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last())>=70 and (Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last())<80	Активировано	
Высокая	{HOST.NAME}. Температура процессора выше 80 градусов	(Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last())>=80	Активировано	
Высокая	{HOST.NAME}. Температура процессора выше 80 градусов	(Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu1.txt].last())>=80	Активировано	

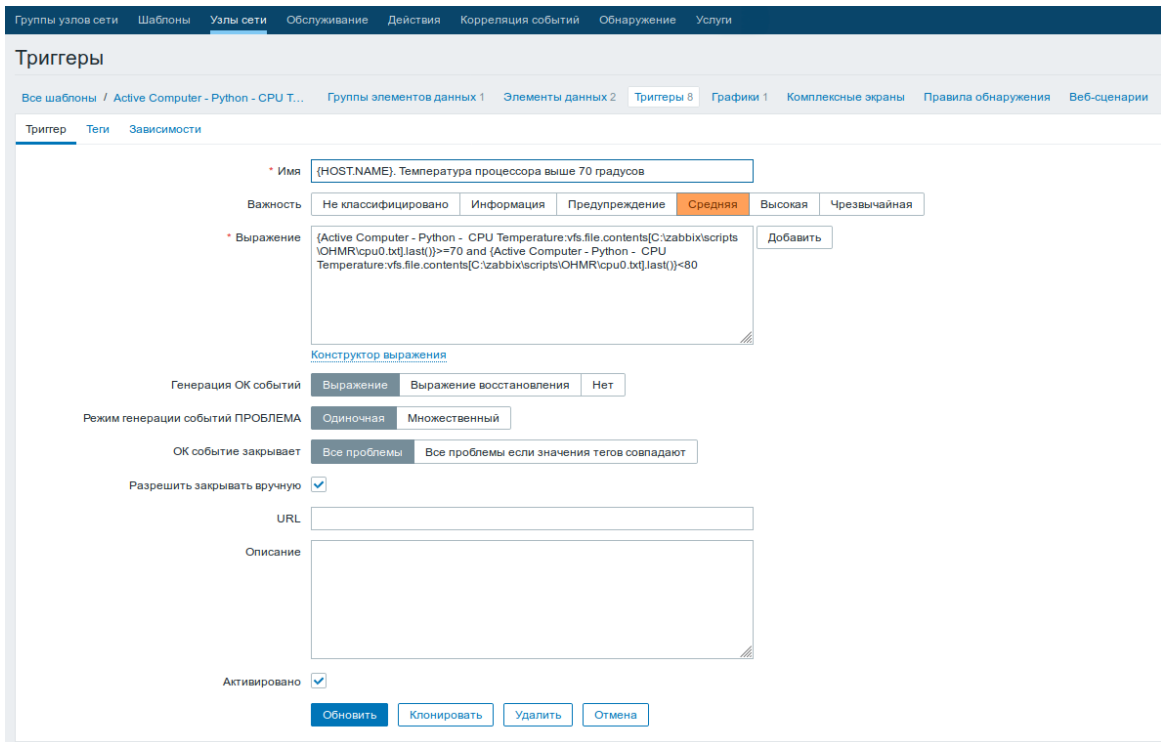
Собрано 8 из 8 элементов

The screenshot shows the Zabbix Triggers configuration interface for a specific trigger. At the top, there are navigation tabs: Группы узлов сети, Шаблоны, Узлы сети, Обслуживание, Действия, Корреляция событий, Обнаружение, Услуги. The main title is "Триггеры". Below it, there are several tabs: Все шаблоны / Active Computer - Python - CPU T..., Группы элементов данных 1, Элементы данных 2, Триггеры 8, Графики 1, Комплексные экраны, Правила обнаружения, Веб-сценарии.

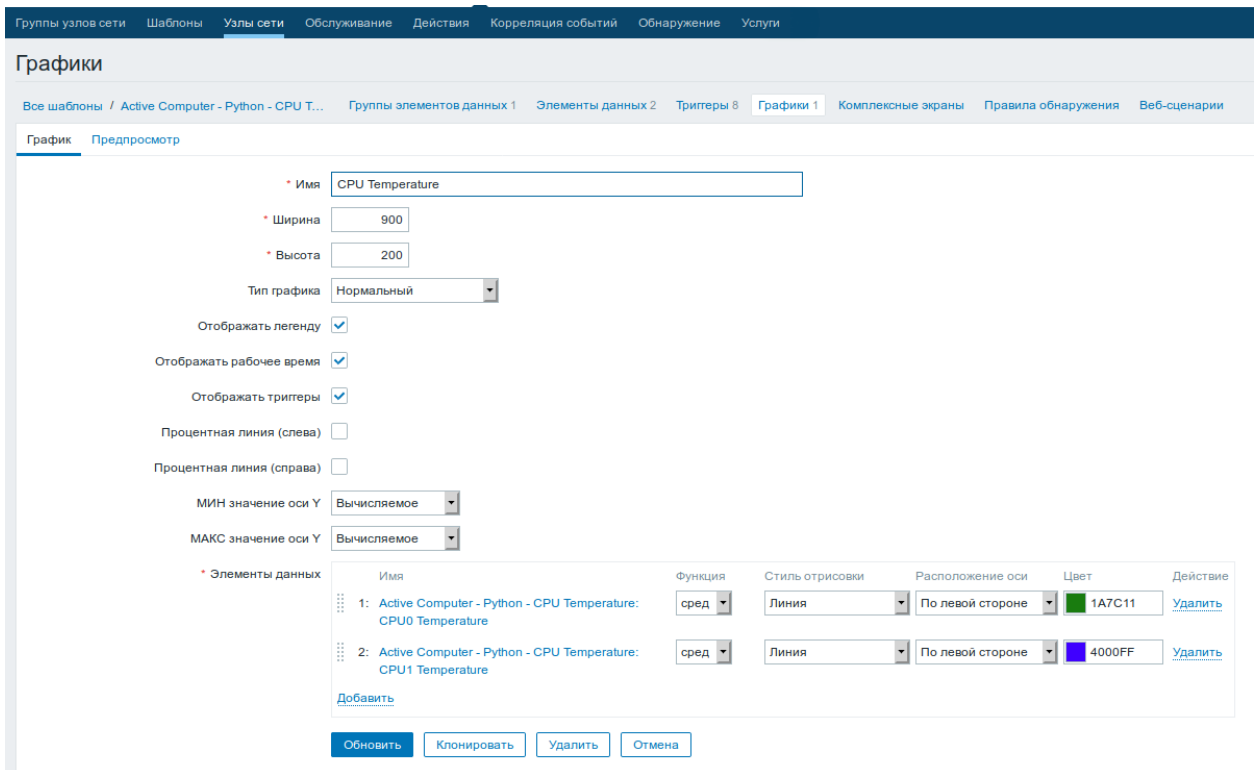
The configuration area includes fields for "Имя", "Важность", "Выражение", "Генерация ОК событий", "Режим генерации событий ПРОБЛЕМА", "ОК событие закрывает", "Разрешить закрывать вручную", "URL", "Описание", and "Активировано". There are also buttons for "Добавить", "Конструктор выражения", "Выражение", "Выражение восстановления", "Нет", "Одиночная", "Множественный", "Все проблемы", "Все проблемы если значения тегов совпадают", "Обновить", "Клонировать", "Удалить", and "Отмена".

The configuration details are as follows:

- Имя: {HOST.NAME}. Температура процессора выше 80 градусов
- Важность: Высокая
- Выражение: {Active Computer - Python - CPU Temperature:vfs.file.contents[C:\zabbix\scripts\OHMR\cpu0.txt].last()}>=80
- Генерация ОК событий: Выражение
- Режим генерации событий ПРОБЛЕМА: Одиночная
- ОК событие закрывает: Все проблемы
- Разрешить закрывать вручную:
- URL: (empty)
- Описание: (empty)
- Активировано:



And for greater clarity, let's add graphics ...

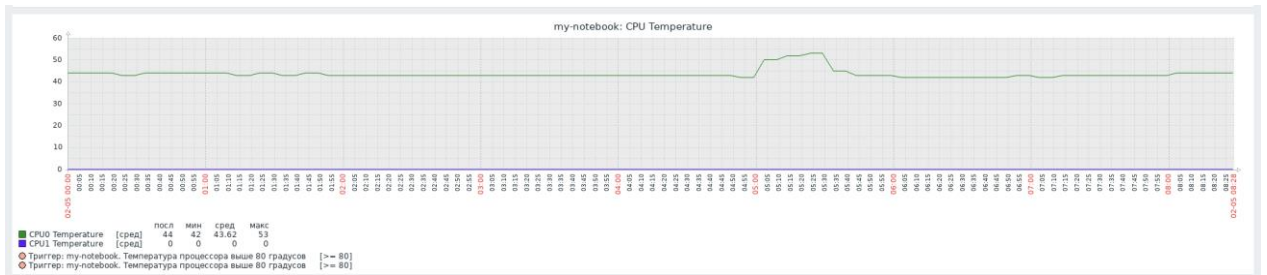


And here is the zipped file ready for import with the template described above: [Active Computer – Python – CPU Temperature.xml](#)

The scripts described earlier can be found in my repository: <https://github.com/mihanic2000/ZabbixActiveAgent/tree/master/zabbix/scripts/OHM>

4. Data collection.

It remains to assign the created template to the corresponding host and wait for the start of data arrival.



Имя	Последняя проверка	Последнее значение	Изменение
my-notebook			
CPU0 Temperature	05.02.2020 08:24:12	44	График
CPU1 Temperature	05.02.2020 08:24:12	0	График

END!