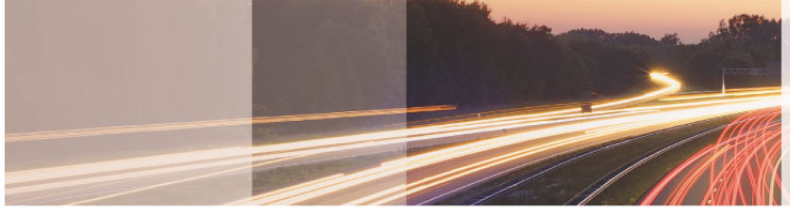


GARR

The Italian Academic & Research Network

www.garr.it

System Monitoring

Yahtfasa (...yet another hard task for a sys admin...)

Alfredo Pagano

WS9, Roma, 16.06.2009

Consortium GARR

Schedule

- **1° parte 09.30 - 11.00**
  - What, why, when, who, where
  - System Monitoring comparison
  - Zabbix shows 1
- **11.00 - 11.30**
  - Stretch your legs... or better.. Coffee Break!
- **2° parte 11.30 - 13.00**
  - Zabbix shows 2
  - Environment monitor (Probes, trigger...)

What (generally) we look for:

- **Real-time monitoring**
  - Performance monitoring
  - Availability monitoring
  - Integrity monitoring
  - Logging

- **Reporting and trending**
  - Easy integration of 3rd party tools
  - Analysis of yearly/monthly/daily statistics
  - SLA reports

- **Assuring SLA**
  - Hierarchial IT Services
  - Real-time SLA reporting

- **Escalations and notifications**
  - Repeated notifications
  - Unlimited escalations

  - Recovery messages
  - Be notified while problem is resolved

- **Dashboard**
  - Personalized dashboard
  - Favourite resources
  - High level view

What (generally) we look for:

- **Visualisation**
  - User-defined views and slide shows
  - Mapping
  - Graphing (pie charts, etc)
  - Zooming

- **Fast Problem Resolution**
  - Alerting users (Email, cell phone, SMS, IM, Browser Plugin)
  - Flexible notification conditions
  - Execute remote commands

- **WEB monitoring**
  - Response time
  - Download speed per second
  - Response code
  - Support of POST and GET methods

- **Flexibility**
  - Support of IPv4 and IPv6
  - Easily extendable native agents
  - Any notifications methods
  - Runs on any platform

- **Pro-active monitoring**
  - Automatic execution of remote commands
  - Automatic IPMI commands

- **Aggregate monitoring**
  - Monitoring of a group of hosts as a single host

- **Agent-based monitoring**
  - Native agent for any platform
  - Immune to connection problems

- **High performance agents**
  - All platforms supported (UNIX, Windows, Novell)
  - Memory utilisation
  - Network utilisation
  - Disk I/O
  - Disk space availability
  - File checksums
  - Monitoring of log files
  - and more

- **Agentless monitoring**
  - Monitoring of remote services (FTP, SSH, HTTP, other)
  - Support of SNMP v1,2,3
  - Support of IPMI
  - SNMP traps

- **Easy Administration**
  - Very fast learning curve (@#@%$!)
  - All data is stored in a database (Oracle, MySQL, PostgreSQL, SQLite)
  - Centralised configuration and storage of information

Scalability

- **Scalability**
  - Tested with 10,000 monitored devices and servers
  - Tested with 100,000 availability and perfomance checks
  - Processing of thousands of availability and performance checks per second

- **Auto discovery**
  - Discovery by IP range, services and SNMP
  - Automatic monitoring of discovered devices

- **Distributed monitoring**
  - Centralized configuration
  - Centralized access to all data
  - Up-to 1000 of nodes

- **XML data import / export**
  - Easy sharing of templates

- **Security**
  - Flexible user permissions
  - Authentication by IP address
  - Protection against brute force attacks

- **All Information is Available Online** (community….)
  - Manual
  - Forums
  - Wiki
- **Backed by a Company**
  - Annual support agreements
  - Turn-key solutions
  - Technical Account Manager
  - Professional Services

- **Open Source Solution**
  - No license driven limitations
  - Access to source code
  - Open to code audit

How to choice

- **Comparison of “network” monitoring systems**

From wikipedia:

http://en.wikipedia.org/wiki/Comparison_of_network_monitoring_systems

Matrix 47x22

47 different applications

22 Characteristics

How to choice. Characteristics

The main characteristics

- SLA Reports
- Logical Grouping
- Trending
- Trend Prediction
- Auto Discovery
- Agent - An agent is a program running on the host being monitored.
- SNMP (Version ?)
- Syslog
- External Scripts - The ability to execute action by running scripts written by the user
- Plugins - Official or user-written extensions that enables fetching new parameters from the monitored hosts
- Plugin Creation - Writing new plugins can be a common task if the user need to extend the product's capabilities

How to choice. Characteristics

- ...
- Triggers / Alerts -Triggers are rules to detect if the system status is compliant with users specifications
- WebApp - Web application that can be used for viewing graphs, systems status, and eventually editing parameters like monitored hosts, triggers, rules
- Distributed Monitoring
- Inventory
- Data Storage Method
- License
- Maps - Maps are a graphical representation of the components being monitored
- Access Control -Access Control is the ability to secure monitoring data via multiple levels of detail based on a password or other security device. Note that even if no access control is supported by the application, the Apache webserver can still block specific pages.
- Events - Events are the ability to acknowledge and record remedial actions

Good rate and GPL

1. Nagios/Cacti (~*)
2. ZenOSS (~*)
3. OpenNMS
4. Groundwork
5. Ganglia
6. Opsview
7. Pandora FMS (*)
8. Hyperic (*)
9. **Zabbix (*)**
10. Osmius
11. Collectd
12. Munin

- * all green! (~*) almost all green!

Zabbix

- http://www.zabbix.com/
- http://www.zabbix.com/documentation.php
  - ZABBIX Manual v1.6.pdf (320pp)
- http://www.zabbix.com/forum/
- http://www.zabbix.com/wiki/doku.php
  - Howtos, Cookbooks, Templates
- https://support.zabbix.com (Bug report)

Install & configure

- You can use this script: install_zabbix.sh
  - * Installs Zabbix 1.6.4 on CentOS / Red Hat 5
  - * Drops an existing database
  - * Does not install MySQL; to install type "yum install mysql-server"
  - * Does not install zabbix packages, it uses source from zabbix.com
  - Configure agent and server with the most useful options
    - http://www.garr.it/pagano/zabbix/install_zabbix.sh

Configure 1/3

- ./configure --enable-agent --enable-server --with-mysql --with-libcurl --with-net-snmp --with-jabber -with-ldap
  - Libcurl Version 7.13.1 or higher required for WEB monitoring module. Optional.
- Other options:

```
--with-oracle[=ARG]
                    use Sqlora8 library [default=no], default is
                    to search through a number of common places for the
                    Sqlora8 files.
 --with-sqlora8[=ARG]
                    use Sqlora8 library [default=no], same as
                    --with-oracle.
 --with-sqlite3[=ARG]
                    use SQLite 3 library [default=no], optionally
                    specify the prefix for sqlite3 library
--with-pgsql[=ARG]
                    use PostgreSQL library [default=no],
                    optionally specify path to pg_config
```

Configure 2/3

- What ODBC driver do you want to use (please select only one):

--with-iodbc[=ARG]

use odbc driver against iODBC package [default=no]

--with-unixodbc[=ARG]

use odbc driver against unixODBC package [default=no], optionally specify path to odbc_config.

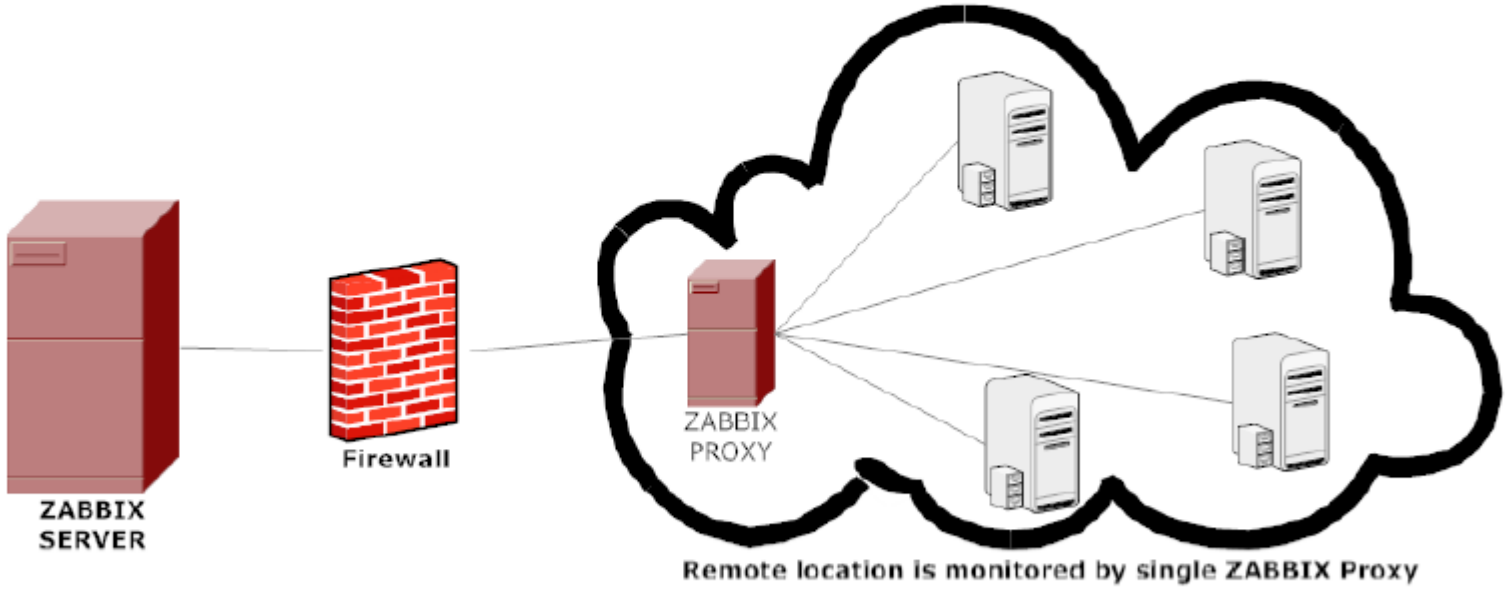
Configure 3/3

- --enable-static Build statically linked binaries
- --enable-ipv6 Turn on support of IPv6
- --with-openipmi
- **--enable-proxy** Turn on proxy server

16.1.Why use Proxy

ZABBIX Proxy can be used for many purposes:

- Offload ZABBIX Server when monitoring thousands of devices
- Monitor remote locations



- Monitor locations having unreliable communications
- Simplify maintenance of distributed monitoring

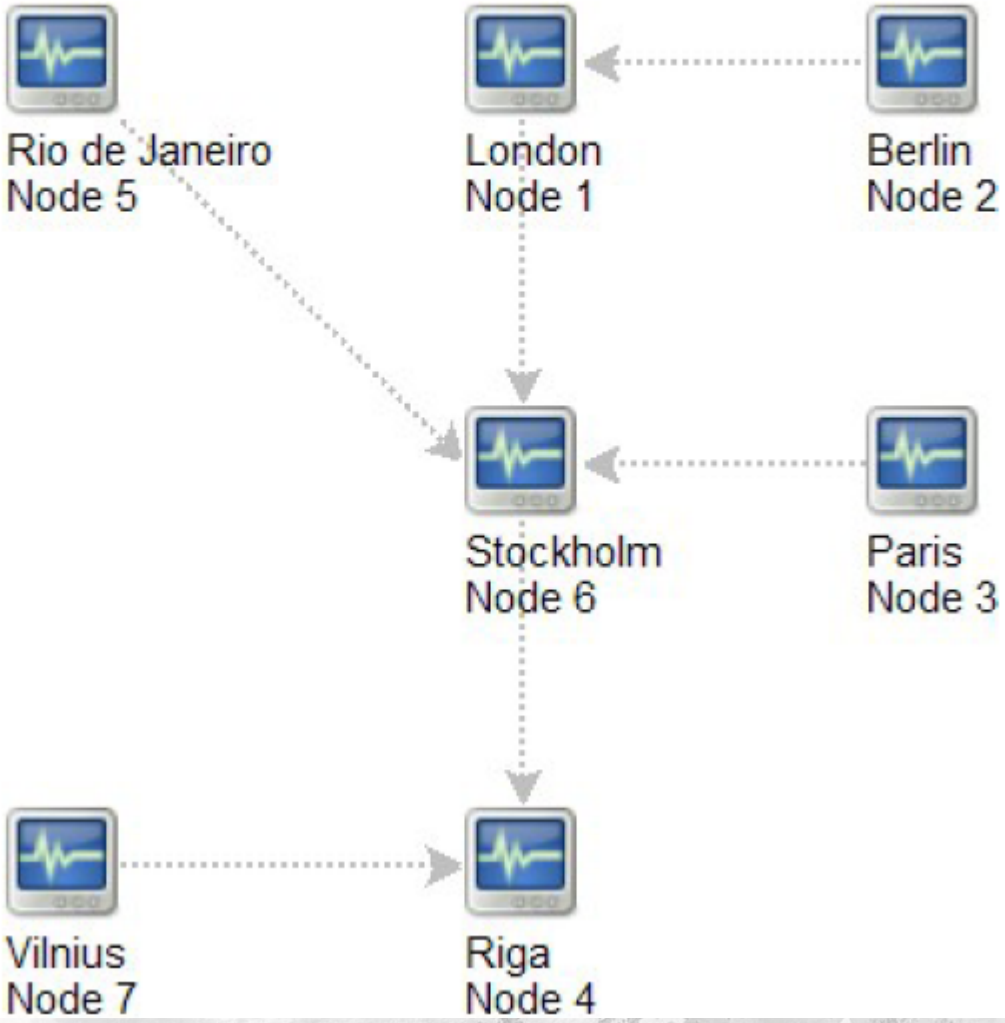
Zabbix zabbix_server.conf

[root@mon zabbix]# more zabbix_server.conf |wc -l
152

- # This defines unique NodeID in **distributed setup (see Distributed Monitoring on** ZABBIX Manual **)**,
- # Default value 0 (standalone server)
- # This parameter must be between 0 and 999
  - **#NodeID=0**
  - ZABBIX supports up-to **1000** (one thousand) Nodes in a distributed setup.

17.3.3.More complex setup

The setup consists of seven Nodes. Each Node may be configured either locally (using local WEB interface) or from one of its Master Nodes.



Zabbix_server.conf

- # Number of pre-forked instances of pollers
- # Default value is 5
- # This parameter must be between 0 and 255
  - **#StartPollers=5**

- # Number of pre-forked instances of IPMI pollers
- # Default value is 0
- # This parameter must be between 0 and 255
  - **#StartIPMIPollers=0**

- # Number of pre-forked instances of pollers for unreachable hosts
- # Default value is 1
- # This parameter must be between 0 and 255
  - **#StartPollersUnreachable=1**

Zabbix zabbix_server.conf

- # Number of pre-forked instances of trappers
- # Default value is 5
- # This parameter must be between 0 and 255
  - **#StartTrappers=5**

- # Number of pre-forked instances of ICMP pingers
- # Default value is 1
- # This parameter must be between 0 and 255
  - **#StartPingers=1**

- # Number of pre-forked instances of discoverers
- # Default value is 1
- # This parameter must be between 0 and 255
  - **#StartDiscoverers=1**

- # Number of pre-forked instances of HTTP pollers
- # Default value is 1
- # This parameter must be between 0 and 255
  - **#StartHTTPPollers=1**

Agent + server processes

- **[root@mon zabbix]# ps uax|grep zabbix | wc –l**
  - **29**

- [root@mon zabbix]# ps uax|grep zabbix
  - root 545 0.0 0.0 61144 724 pts/1 S+ 11:14 0:00 grep zabbix
  - zabbix 13962 0.0 0.0 48908 704 ? SN May14 0:00 /usr/local/sbin/zabbix_agentd
  - zabbix 13964 0.3 0.0 48908 1500 ? SN May14 168:30 /usr/local/sbin/zabbix_agentd
  - zabbix 13965 0.0 0.0 48924 948 ? SN May14 20:21 /usr/local/sbin/zabbix_agentd
  - zabbix 13966 0.0 0.0 48924 944 ? SN May14 20:19 /usr/local/sbin/zabbix_agentd
  - zabbix 13967 0.0 0.0 48924 948 ? SN May14 20:24 /usr/local/sbin/zabbix_agentd
  - zabbix 13968 0.0 0.0 48944 824 ? SN May14 2:49 /usr/local/sbin/zabbix_agentd
  - zabbix 15921 0.0 0.0 71988 1708 ? SN May26 0:00 /usr/local/sbin/zabbix_server
  - zabbix 15926 0.6 0.0 130128 3948 ? SN May26 164:24 /usr/local/sbin/zabbix_server
  - zabbix 15927 0.6 0.0 130096 3912 ? SN May26 167:23 /usr/local/sbin/zabbix_server
  - zabbix 15928 0.6 0.0 130044 3872 ? SN May26 166:31 /usr/local/sbin/zabbix_server

.....

Zabbix zabbix_server.conf

- # Listen port for trapper. Default port number is 10051. This parameter
- # must be between 1024 and 32767
  - **#ListenPort=10051**

- # Source IP address for outgoing connections
  - **#SourceIP=**

- # Listen interface for trapper. Trapper will listen on all network interfaces
- # if this parameter is missing.
  - **#ListenIP=127.0.0.1**

- # How often ZABBIX will perform housekeeping procedure
- # (in hours)
- # Default value is 1 hour
- # Housekeeping is removing unnecessary information from
- # tables history, alert, and alarms
- # This parameter must be between 1 and 24
  - **#HousekeepingFrequency=1**

Zabbix zabbix_server.conf

- # How often ZABBIX will try to send unsent alerts
- # (in seconds)
- # Default value is 30 seconds
  - **SenderFrequency=30**

- # Uncomment this line to disable housekeeping procedure
  - **#DisableHousekeeping=1**

- # Specifies debug level
- # 0 - debug is not created
- # 1 - critical information
- # 2 - error information
- # 3 - warnings (default)
- # 4 - for debugging (produces lots of information)
  - **DebugLevel=3**

Zabbix zabbix_server.conf

Specifies how long we wait for agent response (in sec)
Must be between 1 and 30
Timeout=5

Specifies how many seconds trapper may spend processing new data Must be between 1 and 30.
Trapper ZABBIX Server process responsible for processing of ZABBIX Agent (active) checks, log files and data sent by sender.
#TrapperTimeout=5

After how many seconds of unreachability treat a host as unavailable
#UnreachablePeriod=45

How often check host for availability during the unavailability period
#UnavailableDelay=60

Name of PID file
PidFile=/var/run/zabbix-server/zabbix_server.pid

Zabbix zabbix_server.conf

Name of log file
If not set, syslog is used
LogFile=/var/log/zabbix-server/zabbix_server.log

Maximum size of log file in MB. Set to 0 to disable automatic log rotation.
#LogFileSize=1

Location for custom alert scripts
AlertScriptsPath=/etc/zabbix/alert.d/

Location of external scripts
ExternalScripts=/etc/zabbix/externalscripts

Zabbix zabbix_server.conf

- # Location of fping. Default is /usr/sbin/fping
- # Make sure that fping binary has root permissions and SUID flag set
  - #**FpingLocation=/usr/sbin/fping**

- # Location of fping6. Default is /usr/sbin/fping6
- # Make sure that fping binary has root permissions and SUID flag set
  - #**Fping6Location=/usr/sbin/fping6**

- # Temporary directory. Default is /tmp
  - **#TmpDir=/tmp**

Zabbix zabbix_server.conf

- # Frequency of ICMP pings (item keys 'icmpping' and 'icmppingsec'). Default is 60 seconds.
  - #**PingerFrequency=60**
- # Database host name
- # Default is localhost
  - **#DBHost=localhost**
- # Database name
- # SQLite3 note: path to database file must be provided. DBUser and DBPassword are ignored.
  - **DBName=zabbix**
- # Database user
  - **DBUser=root**
- # Database password
- # Comment this line if no password used
  - **DBPassword=DBPASSWD**
- # Connect to MySQL using Unix socket?
  - #**DBSocket=/tmp/mysql.sock**

Zabbix frontend: zabbix.conf.php


```
<?php

    global $DB;

    $DB["TYPE"]             = "MYSQL";
    $DB["SERVER"]           = "localhost";
    $DB["PORT"]             = "0";
    $DB["DATABASE"]         = "zabbix";
    $DB["USER"]             = "root";
    $DB["PASSWORD"]         = “PASSWD";
    $ZBX_SERVER             = "localhost";
    $ZBX_SERVER_PORT        = "10051";

    $IMAGE_FORMAT_DEFAULT   = IMAGE_FORMAT_PNG;
?>
```

Zabbix zabbix_agent.conf

- [root@lx1 root]# more /etc/zabbix/zabbix_agentd.conf |wc –l
  86
  - DEFAULT: the agent keeps track of what items to send to the server and at what intervals. The agent can poll the server at set intervals in order to keep track of what items it should be sending.
  - **Server=mon.dir.garr.it**
  - # Server port for sending active checks
    - **#ServerPort=10051**
  - # Unique hostname. Required for active checks.
    - **Hostname=(Same name setted on the dashboard, usually hostname –s)**
  - # Listen port. Default is 10050
    - **#ListenPort=10050**
  - # IP address to bind agent
  - # If missing, bind to all available IPs
    - **#ListenIP=127.0.0.1**

Zabbix zabbix_agent.conf

- # Number of pre-forked instances of zabbix_agentd.
- # Default value is 5
- # This parameter must be between 1 and 16
  - **StartAgents=5**

- # How often refresh list of active checks. 2 minutes by default.
  - #**RefreshActiveChecks=120**

- # Disable active checks. The agent will work in passive mode listening server.
  - #**DisableActive=1**

- # Enable remote commands for ZABBIX agent. By default remote commands disabled.

Zabbix agent & Windows

- **A (pretty) Complete Windows Monitoring Solution**
  - http://www.zabbix.com/wiki/doku.php?id=howto:a_pretty_complete_windows_monitoring_solution

    - Windows 2000
    - 2003 Servers
    - Windows XP workstations

    - The typeperf.exe command will show you a list of registered system objects, typically used in perfmon, the windows performance monitoring application.
    - Use the UserParameter section in zabbix_agentd.conf to query any of these values.

Scalability - Hardware Requirements

- Actual configuration depends on number of active items and refresh rates very much.
- It is highly recommended to run the database on a separate box for large installations.

.....

Scalability - db size (History, trends, events)

- Some considerations:
  - **Number of processed values per second**
    - This is average number of new values ZABBIX server receives every second.
    - For example, if we have 3000 items for monitoring with refresh rate of 60
    - Seconds, number of values per seconds is calculated as 3000/60 = **50**.
      - It means that 50 new values are added to ZABBIX database every second.

Scalability - db size

- Housekeeper settings for **history**

- So, if we would like to keep 30 days of **history** and we receive 50 values per second, total number of values will be around (**30***24*3600)***50** = 129.600.000, or about 130 Million of values.

Scalability - db size

- Depending on used database engine, type of received values (floats, integers, strings, log files, etc), disk space for keeping a single value may vary from 40 bytes to hundreds of bytes. Normally it is around 50 bytes per value.

- In our case, it means that 130M of values will require 130M * 50 bytes = **6.5G** of disk space.

Scalability - db size

- Housekeeper setting for **trends**
  - ZABBIX keeps 1 hour max/min/avg/count statistics for each item in table **trends**. The data is used for trending and long period graphs.
  - ZABBIX database, depending on database type, requires about 128 bytes per each total.
  - Suppose we would like to keep trend data for 5 years. 3000 values will require

    (3000/1800)*(24*3600***365**)***128** = **6.3GB** per year, or **31.5GB** for 5 years.

Scalability - db size

- Housekeeper settings for **events**
  - Each ZABBIX event requires approximately 130 bytes of disk space.
  - It is hard number of events generated by ZABBIX daily. In worst case scenario, we may assume that ZABBIX generates one event per second.

    It means that if we want to keep 3 years of events, this would require

    3*365*24*3600***130** = **11GB**

DB size...Finally

- So, the total required disk space can be calculated as:
  - **Configuration (few MB) + History + Trends + Events**
    - **For 3000 items, 1 Y = 16.5 GB**
    - **@GARR**
      - **Number of hosts 79**
      - **Number of items 4673**
        - **[root@mon lib]# du -hs mysql/**
        - **14G mysql/**

DEMO

TIPS. External check

- N.B. Gli script vengono eseguiti esclusivamente dalla macchina SERVER!!

- 1. create an /etc/zabbix/externalscripts directory with correct rights
- 2. copy your script to this directory, suppose your script name is testhttpd.sh
- 3. on zabbix web interface, create an item with type "External Check" and key is testhttpd.sh[]
- 4. REMEMBER to set the square bracket!!
  - http://www.zabbix.com/forum/showthread.php?t=6984&page=3

- esempio: UPS che fa query snmp

TIPS. USER PARAMETER

- Per eseguire dei dalla *macchina stessa (agent)* sono utili gli "user parameter"

Format: UserParameter=<key>,<shell command>
UserParameter=system.test,who|wc -l

- 1. ricordarsi di impostare come key nel frontend lo stesso nome definito prima della virgola in UserParameter

- 2. ricordarsi di cambiare in agent.conf:
  EnableRemoteCommands=1
  DisableActive=0

- La key si puo' mettere senza parentesi quadri (se non ci sono parametri) o con le parentesi quadre
  - es. mon le statistiche di apache e mysql in MON

- UserParameter=proc.pcpu[*],ps -C $1 -o pcpu= | awk '{cpu+=$$1}; END {print cpu}'
- UserParameter=proc.rssmem[*],ps -C $1 -o rss= | awk '{mem+=$$1}; END {print mem}'

- la key nel frontend e' proc.pcpu[httpd] e proc.pcpu[mysq ld]
  - ricordarsi che si possono eseguire le query in locale