

11 Performance tuning

This is a work in progress.

Overview

It is very important to have Zabbix system properly tuned for maximum performance.

Hardware

General advice on hardware:

- Use fastest processor available
- SCSI or SAS is better than IDE (performance of IDE disks may be significantly improved by using utility hdparm) and SATA
- 15K RPM is better than 10K RPM which is better than 7200 RPM
- Use fast RAID storage
- Use fast Ethernet adapter
- Having more memory is always better

Operating system

- Use latest (stable!) version of OS
- Exclude unnecessary functionality from kernel
- Tune kernel parameters

Zabbix configuration parameters

Many parameters may be tuned to get optimal performance.

zabbix_server

StartPollers

General rule - keep value of this parameter as low as possible. Every additional instance of zabbix_server adds known overhead, in the same time, parallelism is increased. Optimal number of instances is achieved when queue, on average, contains minimum number of parameters (ideally, 0 at any given moment). This value can be monitored by using internal check zabbix[queue].

See the "[See also](#)" section at the bottom of this page to find out how to configure optimal count of zabbix processes.

DebugLevel

Optimal value is 3.

DBSocket

MySQL only. It is recommended to use DBSocket for connection to the database. That is the fastest and the most secure way.

Database engine

This is probably the most important part of Zabbix tuning. Zabbix heavily depends on the availability and performance of database engine.

- use fastest database engine, i.e. MySQL or PostgreSQL
- use stable release of a database engine
- rebuild MySQL or PostgreSQL from sources to get maximum performance
- follow performance tuning instructions taken from MySQL or PostgreSQL documentation
- for MySQL, use InnoDB table structure
- ZABBIX works at least 1.5 times faster (comparing to MyISAM) if InnoDB is used. This is because of increased parallelism. However, InnoDB requires more CPU power.
- tuning the database server for the best performance is highly recommended.
- keep database tables on different hard disks
- 'history', 'history_str', 'items', 'functions', 'triggers', and 'trends' are most heavily used tables.
- for large installations keeping MySQL temporary files in tmpfs is:
 - MySQL >= 5.5: not recommended ([MySQL bug #58421](#))
 - MySQL < 5.5: recommended

GUI debugging

Problems related to the frontend performance may be diagnosed using the frontend [debug mode](#).

General advice

- monitor required parameters only
- tune 'Update interval' for all items. Keeping a small update interval may be good for nice graphs, however, this may overload Zabbix
- tune parameters for default templates
- tune housekeeping parameters
- do not monitor parameters which return the same information.
- avoid the use of triggers with long period given as function argument. For example, max(3600) will be calculated significantly slower than max(60).

Viewing Zabbix process performance with "ps" and "top"

Since Zabbix 2.2 processes change their commandlines to display current activity and meaningful statistics, like:

```
UID      PID  PPID  C  STIME TTY      TIME CMD
zabbix22 4584    1   0 14:55 ?        00:00:00 zabbix_server -c
/home/zabbix22/zabbix_server.conf
zabbix22 4587  4584   0 14:55 ?        00:00:00 zabbix_server: configuration
```

```

syncer [synced configuration in 0.041169 sec, idle 60 sec]
zabbix22 4588 4584 0 14:55 ? 00:00:00 zabbix_server: db watchdog
[synced alerts config in 0.018748 sec, idle 60 sec]
zabbix22 4608 4584 0 14:55 ? 00:00:00 zabbix_server: timer #1
[processed 3 triggers, 0 events in 0.007867 sec, 0 maint.periods in 0.005677
sec, idle 30 sec]
zabbix22 4609 4584 0 14:55 ? 00:00:00 zabbix_server: timer #2
[processed 2 triggers, 0 events in 0.004209 sec, idle 30 sec]
zabbix22 4637 4584 0 14:55 ? 00:00:01 zabbix_server: history syncer #4
[synced 35 items in 0.166198 sec, idle 5 sec]
zabbix22 4657 4584 0 14:55 ? 00:00:00 zabbix_server: vmware collector
#1 [updated 0, removed 0 VMware services in 0.000004 sec, idle 5 sec]
zabbix22 4670 1 0 14:55 ? 00:00:00 zabbix_proxy -c
/home/zabbix22/zabbix_proxy.conf
zabbix22 4673 4670 0 14:55 ? 00:00:00 zabbix_proxy: configuration
syncer [synced config 15251 bytes in 0.111861 sec, idle 60 sec]
zabbix22 4674 4670 0 14:55 ? 00:00:00 zabbix_proxy: heartbeat sender
[sending heartbeat message success in 0.013643 sec, idle 30 sec]
zabbix22 4688 4670 0 14:55 ? 00:00:00 zabbix_proxy: icmp pinger #1
[got 1 values in 1.811128 sec, idle 5 sec]
zabbix22 4690 4670 0 14:55 ? 00:00:00 zabbix_proxy: housekeeper
[deleted 9870 records in 0.233491 sec, idle 3599 sec]
zabbix22 4701 4670 0 14:55 ? 00:00:08 zabbix_proxy: http poller #2
[got 1 values in 0.024105 sec, idle 1 sec]
zabbix22 4707 4670 0 14:55 ? 00:00:00 zabbix_proxy: history syncer #4
[synced 22 items in 0.008565 sec, idle 5 sec]
zabbix22 4738 1 0 14:55 ? 00:00:00 zabbix_agentd -c
/home/zabbix22/zabbix_agentd.conf
zabbix22 4739 4738 0 14:55 ? 00:00:00 zabbix_agentd: collector [idle 1
sec]
zabbix22 4740 4738 0 14:55 ? 00:00:00 zabbix_agentd: listener #1
[waiting for connection]
zabbix22 4741 4738 0 14:55 ? 00:00:00 zabbix_agentd: listener #2
[processing request]

```

The main process is an exception. Instead of current activity the original commandline is shown. This helps to distinguish processes on systems with multiple Zabbix instances.

This feature is not implemented for Microsoft Windows.

If logging level is set to **DebugLevel=4** these activity and statistics messages are also written into log file.

Linux

On Linux systems `ps` command can be used together with `watch` command for observing how Zabbix is doing. For example, to run `ps` command 5 times per second to see process activities:

```
watch -n 0.2 ps -fu zabbix
```

To show only Zabbix proxy and agent processes:

```
watch -tn 0.2 'ps -f -C zabbix_proxy -C zabbix_agentd'
```

To show only history syncer processes:

```
watch -tn 0.2 'ps -fC zabbix_server | grep history'
```

The `ps` command produces a wide output (approximately 190 columns) as some activity messages are long. If your terminal has less than 190 columns of text you can try

```
watch -tn 0.2 'ps -o cmd -C zabbix_server -C zabbix_proxy -C zabbix_agentd'
```

to display only commandlines without UID, PID, start time etc.

`top` command also can be used for observing Zabbix performance. Pressing 'c' key in `top` shows processes with their commandlines. In our tests on Linux `top` and `atop` correctly displayed changing activities of Zabbix processes, but `htop` was not displaying changing activities.

BSD systems

If `watch` command is not installed, a similar effect can be achieved with

```
while [ 1 ]; do ps x; sleep 0.2; clear; done
```

AIX, HP-UX

If `watch` command is not available, one can try

```
while [ 1 ]; do ps -fu zabbix; sleep 1; clear; done
```

Solaris

By default the `ps` command does not show changing activities. One option is to use `/usr/ucb/ps` instead. If `watch` command is not installed, a periodically updated list of processes can be shown with

```
while [ 1 ]; do /usr/ucb/ps gxww; sleep 1; clear; done
```

On Solaris 11:

- `/usr/ucb/ps` is not installed by default. You may need to install `ucb` package, e.g. `pkg install compatibility/ucb`,
- if Zabbix daemon has been started by privileged user its activities are not shown to non-privileged user.
- the `sleep` command accepts not only whole seconds but also fractions of second (e.g. `sleep 0.2`).

See also

1. [How to configure optimal count of zabbix processes](#)

From:

<https://www.zabbix.com/documentation/3.2/> - **Zabbix Documentation 3.2**

Permanent link:

https://www.zabbix.com/documentation/3.2/manual/appendix/performance_tuning

Last update: **2017/10/11 13:11**

