

## 11 Оптимизация производительности

Данный раздел в разработке.

### Обзор

Очень важно, чтобы система Zabbix была оптимизирована для получения максимальной производительности.

### Аппаратное обеспечение

Типовые советы по выбору аппаратной конфигурации:

- Используйте самый быстрый процессор из доступных
- SCSI или SAS лучше чем IDE (производительность IDE дисков может быть значительно улучшена с помощью утилиты hdparm) и SATA
- 15K RPM лучше чем 10K, которые в свою очередь лучше чем 7200 RPM
- используйте быстрое RAID хранилище
- используйте быстрый Ethernet адаптер
- большее количество ОЗУ всегда лучше

### Операционная система

- Используйте последнюю (стабильную!) версию ОС
- Отключите ненужный функционал в ядре
- Оптимизируйте настройки ядра

### Параметры конфигурации Zabbix

Можно оптимизировать многие параметры для получения оптимальной производительности.

#### `zabbix_server`

#### StartPollers

Основное правило - использовать как можно меньшее значение. Каждый лишний процесс zabbix добавляет определенную нагрузку, в то же время, распараллеливание увеличивается. Оптимальное количество процессов достигается, когда очередь, в среднем, содержит минимальное количество параметров (идеально, 0 в любой момент времени). За этим значением можно следить, используя внутренний элемент данных `zabbix[queue]`.

Смотрите раздел "[смотрите также](#)" внизу этой страницы, чтобы понять как настроить оптимальное количество zabbix процессов.

#### DebugLevel (Уровень журналирования)

Оптимальное значение 3.

## DBSocket

Только для MySQL. Для соединения с базой данных рекомендуется использовать DBSocket. Это более быстрый и более защищенный путь.

## Движок базы данных

Это возможно наиболее важная часть оптимизации Zabbix. Zabbix сильно зависит от доступности и производительности базы данных.

- используйте наиболее быстрый движок, т.е. MySQL или PostgreSQL
- используйте стабильный выпуск базы данных
- скомпилируйте MySQL или PostgreSQL из исходных кодов для получения максимальной производительности
- следуйте рекомендациям инструкций по оптимизации настроек из документации MySQL или PostgreSQL
- для MySQL, используйте InnoDB структуру таблиц
- Zabbix работает как минимум в 1,5 раза быстрее (по сравнению с MyISAM), если используется InnoDB. Такое поведение достигается из-за увеличенного распараллеливания запросов. Тем не менее, InnoDB требует большую производительность процессора.
- настоятельно рекомендуется оптимизировать настройки сервера базы данных для получения лучшей производительности.
- держите таблицы базы данных на разных жестких дисках.
- наиболее тяжелыми таблицами являются 'history', 'history\_str', 'items', 'functions', 'triggers', и 'trends'
- для больших инсталляций хранение временных файлов MySQL в tmpfs:
  - MySQL >= 5.5: не рекомендуется ([MySQL bug #58421 \[en\]](#))
  - MySQL < 5.5: рекомендуется

## Отладка GUI

Проблемы связанные с производительностью веб-интерфейса можно диагностировать при помощи [режима отладки](#) веб-интерфейса.

## Общая рекомендация

- наблюдайте только необходимые параметры
- оптимизируйте “Интервал обновления” по всем элементам данных. Использование небольшого интервала обновления может быть удобно для приятных графиков, но небольшой интервал обновления может перегрузить Zabbix
- оптимизируйте параметры стандартных шаблонов
- оптимизируйте параметры очистки истории
- не наблюдайте параметры возвращающие одни и те же значения
- избегайте использования триггеров с большими периодами в функциях. Например,

max(3600) будет вычисляться намного медленнее, чем max(60).

## Наблюдайте за производительностью Zabbix процессов с помощью утилит "ps" и "top"

Начиная с Zabbix 2.2, процессы отображают в командной строке текущую активность и значимую статистику, примерно так:

```

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]

```

Основной процесс является исключением. Вместо текущей активности отображается изначальная командная строка. Такое поведение помогает разделять в системе процессы с несколькими экземплярами Zabbix.

Такая возможность не реализована на платформах Microsoft Windows.

Если уровень журналирования указан равным **DebugLevel=4**, подобные сообщения о активности и статистике также записываются в файл журнала.

## Linux

В системах Linux команда `ps` может использоваться с командой `watch` для наблюдения, что делает Zabbix. Например, запуск команды `ps` 5 раз в секунду для наблюдения за активностью процессов:

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

Наблюдение только за процессами Zabbix прокси и агента:

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

Наблюдение только за процессами history syncer:

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

Команда `ps` предоставляет широкий вывод (примерно 190 символов), который требуется для некоторых сообщений активности. Если ваш терминал имеет менее чем 190 символов для текста, вы можете попробовать следующий способ:

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

для отображения только полезной информации без UID, PID, времени старта и прочего.

Можно также использовать команду `top` для наблюдения за производительностью Zabbix. Нажатие клавиши 'c' в команде `top` отобразит процессы с их командами строками. В наших тестах в Linux `top` и `atop` отображают корректно изменения активности процессов Zabbix, но в `htop` смена активности не отображалась.

## BSD systems

Если команда `watch` не установлена, подобный вывод можно получить следующим способом:

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

## AIX, HP-UX

Если команда `watch` не доступна, можно использовать следующий метод:

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

## Solaris

По умолчанию команда `ps` не отображает изменения активности. Вместо этой команды можно использовать `/usr/ucb/ps`. Если команда `watch` не установлена, периодическое обновление списка процессов может выполнить следующим способом:

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

В Solaris 11:

- `/usr/ucb/ps` по умолчанию не установлен. Вам возможно потребуется установить `ucb` пакет, т.е. `pkg install compatibility/ucb`,
- если демон Zabbix запускается от привилегированного пользователя, его активность не отображается непривилегированному пользователю.
- команда `sleep` принимает не только секунды, но также и дроби секунд (т.е. `sleep 0.2`).

## Смотри также

1. [Как настроить оптимальное количество zabbix процессов](#)

From:

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

Permanent link:

[https://www.zabbix.com/documentation/3.2/ru/manual/appendix/performance\\_tuning](https://www.zabbix.com/documentation/3.2/ru/manual/appendix/performance_tuning)

Last update: **2018/06/13 11:48**

