

Как мы доработали zabbix-agent для мониторинга баз данных Oracle/MySQL/PostgreSQL/MSSQL на разных ОС

Михаил Григорьев



Что нам нужно, какое решение?

- Кросс-платформенное ([Linux](#), [Windows](#), [AIX](#), [Solaris](#), а может будет нужно HP-UX или FreeBSD?);
- Поддержка разных СУБД ([Oracle](#), [MSSQL](#), [MySQL](#), [PostgreSQL](#));
- Широкий спектр поддерживаемых версий ОС, от Windows 2003 Server или Oracle Linux 5 и выше;
- Широкий спектр поддерживаемых версий СУБД, от PostgreSQL 9.4 или MariaDB 5.5 и выше;
- Поддержка пула соединений к СУБД;
- Работа за NAT, поддержка шифрования (PSK/SSL) при работе с zabbix-server (proxy);
- Быстрая работа агента-мониторинга для частого опроса состояния СУБД, малое потребление ресурсов;
- Готовые модульные шаблоны под разные СУБД ([Oracle](#), [MSSQL](#), [MySQL](#), [PostgreSQL](#));
- Максимально простая настройка агента и шаблонов;
- Компактное решение, минимум зависимостей;
- Возможность кастомизации собираемых данных;
- Открытый исходный код;

Инструменты мониторинга баз данных

- ODBC мониторинг (встроенный в Zabbix)
- Zabbix agent с помощью UserParameter (сторонние скрипты на bash/python/powershell/etc)
- Модули для Zabbix agent (libzbxpgsql, libzbxredis, zabbix-module-mysql, etc)
- Внешними агентами (mamonsu, DBforBIX, ZabbixDBA, zbxdb, zbxora, etc)
- Zabbix agent 2 (Redis, Memcached, PostgreSQL, MySQL, недавно появился плагин Oracle)

Достоинства и недостатки решений

- ODBC мониторинг

- + Встроен в Zabbix

- + Поддерживает большое количество СУБД (Oracle, MySQL, PostgreSQL, MSSQL, IBM DB2, MongoDB, Firebird, etc);

- Официальные шаблоны для популярных СУБД (MySQL, MSSQL, Oracle)(с февраля 2020);

- Работает только на zabbix-server и zabbix-proxy, необходимо прямое соединение с СУБД => Невозможно использовать на серверах СУБД находящихся за NAT;

- Нет поддержки пула соединений => Каждый запрос – это новое соединение с СУБД (в версии 4.4 появился db.odbc.get, таким образом можно минимизировать количество запросов, 1 запрос – много данных, далее препроцессинг);

- Не кросс-платформенное решение (нельзя использовать там где нет *nix вообще);

Достоинства и недостатки решений

- Zabbix Agent с помощью UserParameter (сторонние скрипты на bash/python/powershell/etc)
 - + Работают непосредственно на сервере с СУБД (активный или пассивный режим агента, при активном NAT не страшен);
 - + Быстро и относительно легко можно написать свое “коленочное” решение;
 - + Кросс-платформенность за счет скриптов на разных языках;
 - Нет поддержки пула соединений => Каждый запуск скрипта – это новое соединение с СУБД;
 - При большом количестве собираемых метрик и высокой частоте опроса возрастает нагрузка на сервер;
 - Нет единого решения под Windows/Linux/AIX и разные СУБД;
 - Невозможность использования решение из-за инфраструктурных ограничений или политики безопасности;
 - Необходимо ставить дополнительное ПО (например python);

Достоинства и недостатки решений

- Модули для Zabbix Agent (libzbxpgsql, libzbxredis, zabbix-module-mysql, etc)

+ Работают непосредственно на сервере с СУБД (активный или пассивный режим агента, при активном NAT не страшен);

- Работают только на Linux;

- Нет единого модуля под все популярные СУБД, каждый модуль – своя СУБД;

- Поддержка многих модулей заброшена, модули и шаблоны не развиваются;

- Нет поддержки пула соединений => Каждый запрос – это новое соединение с СУБД;

- Многие модули нужно самостоятельно собирать из исходников, нужен квалифицированный инженер;

Достоинства и недостатки решений

- Внешними агентами (mamonsu, DBforBIX, ZabbixDBA, zbxdb, zbxora, etc)
 - + Работают непосредственно на сервере с СУБД (активный или пассивный режим агента, при активном NAT не страшен);
 - + Многие агенты кросс-платформенны, но не все проверены на Windows/AIX;
 - + Многие агенты “поддерживают” пул соединений;
 - Некоторые агенты давно заброшены и не развиваются, нет готовых шаблонов под разные СУБД (DBforBIX, ZabbixDBA);
 - Некоторые агенты сложны в настройке (DBforBIX, zbxdb);
 - Невозможность использования решение из-за инфраструктурных ограничений или политики безопасности (например, требуется PSK/SSL шифрование при отправке данных мониторинга);
 - Необходимо ставить дополнительное ПО (например Python, Java или Perl);

Достоинства и недостатки решений

- Zabbix Agent 2 – за ним будущее

- + Работает непосредственно на сервере с СУБД (активный или пассивный режим агента, при активном NAT не страшен, поддержка шифрования PSK/SSL);

- + Написан на Golang => кросс-платформенный, но только под Linux и Windows;

- + Есть поддержка мониторинга Redis, Memcached, PostgreSQL, MySQL, недавно появился плагин Oracle;

- Написан на Golang => Не настолько кросс-платформенный как хотелось бы, AIX версии < 7.2 не поддерживается компилятором Golang, о Solaris или FreeBSD мечтать не приходится;

- Пока реализована поддержка далеко не всех популярных СУБД (нет MSSQL);

- Не поддерживает работу в виде службы в Windows и в виде демона в Linux;

Что мы имеем в итоге, какое решение выбрать?

Увы, но никакое решение не удовлетворяет нашим потребностям.

2 года назад мы начали внедрение мониторинга СУБД с помощью Zabbix на основе UserParameter и написали свои скрипты и шаблоны с нуля.

Для Linux/AIX на bash (1 скрипт, порядка 7000 строк кода).

Для Windows – powershell/c# (несколько скриптов, 4000 строк кода).

Но через 2 года пройдя по всем проблемам, мы осознали, что нужно новое решение, которое бы отвечало нашим требованиям (см. слайд 2).

Примерно за 1 месяца (декабрь 2019) я доработал Zabbix agent (на Си) чтобы он мог осуществлять базовый мониторинг 3 популярных баз (Oracle, MySQL, PostgreSQL) под Windows/Linux/AIX.

За основу была взята версия Zabbix v4.4, на которую мы перешли с своей версии Zabbix v3.4.17.

Что может доработанный Zabbix-agent ?

- Он остался кросс-платформенным (поддерживается сборка под [Linux](#), [Windows](#), [AIX](#), [Solaris](#)), можно собрать под другие ОС, но нам пока не нужно;
- Он поддерживает мониторинг [Oracle](#), [MySQL](#), [PostgreSQL](#), [MSSQL](#) на всех ОС, не нужны никакие скрипты*, костыли и прочее (что именно может мониторить для каждой СУБД ниже);
- Пока (на 14.09.2020) не реализован пул соединений, но эта задача в ближайшем приоритете;
- Zabbix-agent как и раньше поддерживает мониторинг параметров ОС, поддерживает шифрования (PSK/SSL), это никуда не исчезло из базовых возможностей агента;
- Быстрая работа мониторинга, малое потребление ресурсов CPU и RAM (да здравствует Си!);

Zabbix agent (DBMON) (7 элементов данных)

[DBMON] Agent memory usage (VmHWM) ?	14.09.2020 10:02:07	5.21 MB
[DBMON] Agent memory usage (VmPeak) ?	14.09.2020 10:02:07	134.13 MB
[DBMON] Agent memory usage (VmRSS) ?	14.09.2020 10:02:07	5.21 MB
[DBMON] Agent memory usage (VmSize) ?	14.09.2020 10:02:07	134.13 MB
[DBMON] Agent ping	14.09.2020 10:02:42	Up (1)
[DBMON] Host name of zabbix_agentd running	14.09.2020 09:26:52	crmcluster.sc
[DBMON] Version of zabbix_agent(d) running	14.09.2020 09:26:52	5.0.2

Что может доработанный Zabbix-agent ?

- Есть полностью готовые модульные шаблоны для мониторинга Oracle, MySQL, PostgreSQL для разных ОС

<input type="checkbox"/> Имя ▲	Узлы сети	Группы элементов данных	Элементы данных	Триггеры	Графики	Комплексные экраны	Обнаружение	Веб	Присоединенные шаблоны	Linked to templates	Теги
<input type="checkbox"/> App Zabbix Agent for AIX (Active, DBMON, Memory)	Узлы сети	Группы элементов данных 1	Элементы данных 4	Триггеры 2	Графики	Комплексные экраны	Обнаружение	Веб			
<input type="checkbox"/> App Zabbix Agent for Linux (Active, DBMON, Memory)	Узлы сети	Группы элементов данных 1	Элементы данных 4	Триггеры 2	Графики	Комплексные экраны	Обнаружение	Веб			
<input type="checkbox"/> App Zabbix Agent for Windows (Active, DBMON, Memory)	Узлы сети	Группы элементов данных 1	Элементы данных 2	Триггеры 2	Графики	Комплексные экраны	Обнаружение	Веб			
<input type="checkbox"/> DB MySQL (Active, DBMON, for all OS)	Узлы сети	Группы элементов данных 10	Элементы данных 159	Триггеры 23	Графики 7	Комплексные экраны 1	Обнаружение 5	Веб		MySQL for Linux (Active, DBMON), MySQL for Windows (Active, DBMON)	DBType: MySQL
<input type="checkbox"/> DB Oracle (Active, DBMON, for all OS)	Узлы сети	Группы элементов данных 4	Элементы данных 48	Триггеры 34	Графики 6	Комплексные экраны 1	Обнаружение 10	Веб		Oracle for AIX (Active, DBMON), Oracle for Linux (Active, DBMON), Oracle for Windows (Active, DBMON)	DBType: Oracle
<input type="checkbox"/> DB PostgreSQL (Active, DBMON, for all OS)	Узлы сети	Группы элементов данных 11	Элементы данных 65	Триггеры 11	Графики 2	Комплексные экраны	Обнаружение 1	Веб		PostgreSQL for Linux (Active, DBMON), PostgreSQL for Windows (Active, DBMON)	DBType: PostgreSQL
<input type="checkbox"/> MySQL for Linux (Active, DBMON)	Узлы сети 2	Группы элементов данных 10	Элементы данных 160	Триггеры 24	Графики 7	Комплексные экраны	Обнаружение 5	Веб	DB MySQL (Active, DBMON, for all OS)		OS: Linux
<input type="checkbox"/> MySQL for Windows (Active, DBMON)	Узлы сети	Группы элементов данных 10	Элементы данных 164	Триггеры 25	Графики 7	Комплексные экраны	Обнаружение 5	Веб	DB MySQL (Active, DBMON, for all OS)		OS: Windows
<input type="checkbox"/> Oracle for AIX (Active, DBMON)	Узлы сети	Группы элементов данных 4	Элементы данных 49	Триггеры 35	Графики 6	Комплексные экраны	Обнаружение 12	Веб	DB Oracle (Active, DBMON, for all OS)		OS: AIX
<input type="checkbox"/> Oracle for Linux (Active, DBMON)	Узлы сети	Группы элементов данных 4	Элементы данных 49	Триггеры 35	Графики 6	Комплексные экраны	Обнаружение 12	Веб	DB Oracle (Active, DBMON, for all OS)		OS: Linux
<input type="checkbox"/> Oracle for Windows (Active, DBMON)	Узлы сети	Группы элементов данных 4	Элементы данных 54	Триггеры 38	Графики 6	Комплексные экраны	Обнаружение 10	Веб	DB Oracle (Active, DBMON, for all OS)		OS: Windows
<input type="checkbox"/> PostgreSQL for Linux (Active, DBMON)	Узлы сети	Группы элементов данных 11	Элементы данных 66	Триггеры 12	Графики 2	Комплексные экраны	Обнаружение 1	Веб	DB PostgreSQL (Active, DBMON, for all OS)		OS: Linux
<input type="checkbox"/> PostgreSQL for Windows (Active, DBMON)	Узлы сети	Группы элементов данных 11	Элементы данных 68	Триггеры 13	Графики 2	Комплексные экраны	Обнаружение 1	Веб	DB PostgreSQL (Active, DBMON, for all OS)		OS: Windows

Отображено 13 из 13 найденных

- **Дополнительный шаблон мониторинга агента**
- **Базовый шаблон под любую ОС**
- **Шаблон для мониторинга MySQL для каждой ОС**
- **Шаблон для мониторинга Oracle для каждой ОС**
- **Шаблон для мониторинга PostgreSQL для каждой ОС**

Что может доработанный Zabbix-agent ?

- Минимум настроек у агента

Для Oracle/MySQL нужно указать логин и пароль для подключения к СУБД в `zabbix_agentd_dbmon.conf`, все остальное настраивается через макросы в web-frontend.

Для Oracle возможна работа как с подключением в формате easy connect так и через tnsname (опция `OracleUseLocalEnv=1` + переменные среды `ORACLE_SID/ORACLE_HOME` или `TWO_TASK/LOCAL`)

Для PostgreSQL/MySQL строка подключения полностью указывается в web-frontend через макрос.

```
Alias=dbmon.agent.ping[*]:agent.ping
Alias=dbmon.agent.version[*]:agent.version
Alias=dbmon.agent.hostname[*]:agent.hostname
DBTimeout=300
MySQLUser=zabbixmon
MySQLPassword=ieYee7exae7m
OracleUser=sys
OraclePassword=ieYee7exae7m
OraclePrimaryUser=sys
OraclePrimaryPassword=ieYee7exae7m
OracleUseLocalEnv=0
```

— Доп. алиасы для мониторинга агента

— Таймаут работы тредов с СУБД

— Логин и пароль для MySQL

— Логин и пароль для подключения к Oracle + доп. настройки

Что может доработанный Zabbix-agent ?

Пример макросов для Oracle (Красные – обязательные)

Макросы Инвентаризация Шифрование

Макросы узла сети Макросы узла сети и унаследованные

Макрос	Значение	Описание	
{DBS_ORACLE_CONN_STRING}	⇒ 127.0.0.1:1533/SC	Format connection string: username/userpassword@host[:port][[/service name]][:server][/instance_name]	Удалить Строка в формате easy connect
{DBS_ORACLE_INSTANCE}	⇒ SC	Oracle SID	Удалить Имя экземпляра
{DBS_ORACLE_LLD_INCLUDE_LISTENER}	⇒ ^(sc SC)\$	Only sc listener	Удалить Фильтр для LLD листенера
{DBS_ORACLE_LSNR_SERVICE_CMD_REGEXP}	⇒ ^.*sc.*\$	Only sc listener	Удалить
{DBS_ORACLE_MODE}	⇒ 1	0 - OCI_DEFAULT, 1 - OCI_SYSDBA, 2 - OCI_SYSPER, 3 - OCI_SYSASM, 4 - OCI_SYSDGD	Удалить Роль подключения к экземпляру
{DBS_ZABBIX_DBMON_PROCESS_CMD_REGEXP}	⇒ ^.*zabbix_agentd_dbmon_sc\..conf\$	Zabbix-agent config file	Удалить Фильтр конфига агента















[Добавить](#)

[Обновить](#) [Клонировать](#) [Полное клонирование](#) [Удалить](#) [Отмена](#)

Что может доработанный Zabbix-agent ?

- Компактное решение, минимум зависимостей

Для Windows всего 12 бинарных файлов + конфиг `zabbix_agentd_dbmon.conf`, есть инсталлятор в котором можно настроить базовые параметры.

Name	Date modified	Type	Size
 zabbix_agentd_dbmon.exe	01.09.2020 9:52	Application	1 025 KB
 libconfig.dll	30.06.2020 13:01	Application extension	82 KB
 libeay32.dll	30.06.2020 13:01	Application extension	2 058 KB
 libiconv.dll	30.06.2020 13:01	Application extension	925 KB
 libintl.dll	30.06.2020 13:01	Application extension	98 KB
 libmariadb.dll	30.06.2020 13:01	Application extension	1 050 KB
 libpq.dll	30.06.2020 13:01	Application extension	293 KB
 libssl32.dll	30.06.2020 13:01	Application extension	352 KB
 msvcp120.dll	30.06.2020 13:01	Application extension	645 KB
 msvcp140.dll	30.06.2020 13:01	Application extension	604 KB
 msucr120.dll	30.06.2020 13:01	Application extension	942 KB
 oci.dll	30.06.2020 13:01	Application extension	805 KB
 ssleay32.dll	30.06.2020 13:01	Application extension	352 KB
 zabbix_agentd_dbmon.conf	04.08.2020 8:40	CONF File	2 KB

Для RedHat-подобных дистрибутивов есть собственный репозиторий агента, все ставится из него – минимум телодвижений.

Есть отдельные сборки агента под разные Oracle RDBMS (11g/12c/18c/19c) + скрипт на bash для автоматизации развертывания.

Что может доработанный Zabbix-agent ?

- Возможность кастомизации собираемых данных

Это возможность создавать собственные item's и писать SQL запросы в файле конфигурации, без перекомпиляции агента. Пока это реализовано для MySQL и PostgreSQL.

```
zabbix_agentd_dbmon_sql.conf x
1 queries = {
2
3 # MySQL version
4 # zabbix_get.exe -s 127.0.0.1 -k mysql.query.nojson[127.0.0.1,3306,mysqlver]
5 # Result:
6 # 5.7.29-log
7 mysqlver = "SELECT /*DBS_001*/ VERSION() AS VERSION;";
8
9 # MySQL info:
10 # zabbix_get.exe -s 127.0.0.1 -k mysql.query.onerow[127.0.0.1,3306,mysqlinfo]
11 # Result:
12 # {"SERVER_UUID":"07e11217-5fbl-1lea-818e-00155d4ed51b","VERSION":"5.7.29-log"}
13 mysqlinfo = "SELECT /*DBS_002*/ UUID() AS SERVER_UUID, VERSION() AS VERSION;";
14
15 # MySQL database discovery:
16 # zabbix_get.exe -s 127.0.0.1 -k mysql.query.discovery[127.0.0.1,3306,mysqldbdiscovery]
17 # Result:
18 # [
19 # {
20 #   "{#DBNAME}": "blog"
21 # },
22 # {
23 #   "{#DBNAME}": "mysql"
24 # },
25 # {
26 #   "{#DBNAME}": "sys"
27 # }
28 # ]
29 mysqldbdiscovery = "SELECT /*DBS_003*/ s.SCHEMA_NAME AS DBNAME "
30 "FROM information_schema.schemata s WHERE s.SCHEMA_NAME NOT REGEXP '(information_schema|performance_schema)' "
31 "GROUP BY s.SCHEMA_NAME;";
32
33 };
```

Доработанный Zabbix-agent в цифрах

> 90 метрик Oracle (версии 10i/11g/12c/18c/19c)

> 70 триггеров

- Мониторинг основных показателей экземпляра (uptime, доступность, статус, сессии, процессы, лимиты по дата-файлам, блокировки, статус archiver и прочее);
- Мониторинг Single database/Pluggable database (размер, role, open_mode, flashback и прочее);
- Мониторинг Fast Recovery Area (FRA);
- Мониторинг бэкапов;
- Мониторинг permanent, undo, temporary tablespace;
- Мониторинг Dataguard (MRP status, apply and transport lag);
- Мониторинг ошибок в alert.log (ORA-00600 и другие);
- Мониторинг Archive Log Destination;
- Мониторинг параметров экземпляра (изменение дефолтных параметров);
- Мониторинг Automatic Storage Management (Oracle ASM);
- Мониторинг работы листенера и сервисов;

Доработанный Zabbix-agent в цифрах

> 120 метрик MySQL (MySQL 5.5-7/8.0, MariaDB 5.5/10.1-4, Percona 5.6-7/8.0)

> 25 триггеров

- Мониторинг основных показателей экземпляра (uptime, доступность, версия, threads, query cache, slow query, innodb buffer pool и прочее) , возможность легко добавлять новые наблюдаемые показатели через web-frontend;
- Мониторинг основных параметров (query cache, репликации), возможность легко добавлять новые наблюдаемые параметры через web-frontend;
- Мониторинг и расчет основных показателей производительности (Buffer pool dirty, Buffer pool efficiency, Buffer pool utilization, Query cache efficiency и прочее);
- Мониторинг каждой базы (размер, кодировки);
- Мониторинг репликации (статус, лаг);
- Мониторинг размеров больших таблиц (TOP10 по размеру, по количеству записей);
- Мониторинг блокировок в InnoDB;
- Мониторинг error.log на предмет ошибок, блокировок и прочего;

Доработанный Zabbix-agent в цифрах

> 100 метрик PostgreSQL (9.4 – 12.2)

> 15 триггеров

- Мониторинг основных показателей экземпляра (uptime, доступность, версия);
- Количество соединений;
- Чекпойнты;
- Блокировки;
- Подготовленные транзакции;
- Процессы autovacuum;
- Скорость генерации WAL;
- Скорость чтения и записи;
- Размеры БД, deadlock's, и прочее;

Планы на будущее

- Разработка мониторинга MSSQL (уже идет);
- Реализация пула соединений к СУБД (ноябрь 2020);
- Расширить возможность легкой кастомизации собираемых данных без перекомпиляции агента (декабрь 2020);

Спасибо за внимание!



Сайт компании: <https://db-service.ru>

Телефон: +7 (495) 374 84 96

Email: support@db-service.ru