

ТЕХНИЧЕСКИЙ РАЗБОР

Zabbix для 1С: сеансы, блокировки СУБД и кластер под контролем

Как мы ловим зависший сеанс и блокировку СУБД раньше,
чем бухгалтер



Ай-ТИ Фреш

Июль 2026

itfresh.ru · ИТ-аутсорсинг для юридических лиц

Суть проблемы

Заказчик спохватывается, когда бухгалтер не может провести накладную: рабочий процесс 1С завис, СУБД держит блокировку, лицензии кластера исчерпаны. Мы разворачиваем в Zabbix 7.0 LTS контроль трёх слоёв — RAS/rac кластера 1С, СУБД (MS SQL/PostgreSQL) и техжурнала, — чтобы увидеть признаки до жалобы пользователя и снять блокировку без потери рабочего времени бухгалтерии.

Почему это важно бизнесу

- Простой 1С в момент закрытия периода или зарплаты стоит бухгалтерии рабочего дня — блокировка проведения останавливает весь учётный контур.
- Зависший rghost с нулевым числом соединений держит гигабайты памяти и не освобождает лицензии — новые пользователи не могут войти в базу.
- Блокировка на уровне СУБД видна администратору БД за секунды, но остаётся незаметной для 1С-специалиста без прямого доступа к DMV.
- Разбор инцидента постфактум по логам занимает часы; готовый триггер в Zabbix с историей сеанса сокращает разбор до минут.
- Аварийный перезапуск rghost по превышению памяти без мониторинга рвёт все активные сеансы и лицензии пользователей одновременно.



Ключевые параметры реализации

20 сек

дефолтный таймаут ожидания управляемой блокировки данных информационной базы 1С
its.1c.ru, метод. поддержка

80% / 95%

пороги «временно допустимый» / «критический» объема памяти рабочих процессов сервера 1С (доля...
v8.1c.ru, платформа 1С

1.5 / 6 сек

границы APDEX satisfied/tolerating (T0 и 4×T0) для ключевых операций 1С — наш стандарт порогов
its.1c.ru, методика APDEX

30 сек

порог триггера по blocking_session_id и wait_time в sys.dm_exec_requests MS SQL
наш стандарт, MS Learn

1 сек

deadlock_timeout PostgreSQL по умолчанию — с него мы включаем log_lock_waits
postgrespro.ru, документация

60 сек

интервал опроса Zabbix-item сеансов и блокировок кластера 1С через ras
наш стандарт



Бухгалтерия ТСЖ: блокировка проведения документов при закрытии месяца

Что настраиваем

Клиент ITfresh — ТСЖ на 1С:Бухгалтерии НКО, MS SQL Server 2019, до 8 одновременных сеансов

Как мы это делаем

- 1 Разворачиваем RAS на сервере 1С и шаблон Zabbix agent 2 с UserParameter, вызывающим `rac session list --cluster=<uuid> --infobase=<uuid>`.
- 2 Заводим item'ы: число активных сеансов, сеансы дольше 60 секунд с текущим вызовом СУБД, память и число соединений rphost из `rac process list`.
- 3 На стороне MS SQL подключаем шаблон MSSQL by ODBC и триггер на `blocking_session_id<>0` при `wait_time` выше 30000 мс в `sys.dm_exec_requests`.
- 4 При срабатывании триггера действие Zabbix шлёт алерт дежурному с UUID сеанса — он снимает блокировку через `rac session terminate`.
- 5 Итог: закрытие месяца проходит без разрыва сеансов, время реакции на блокировку сократилось с 40 минут по звонку до 3 минут по алерту.

РЕЗУЛЬТАТ

Инцидент «не проводится накладная» перестал доходить до звонка директору — дежурный видит блокировку в Zabbix раньше пользователя и снимает её через рас до эскалации; простой учётного контура при закрытии периода сократился до единиц минут.

КЛЮЧЕВОЙ НЮАНС

Мониторить нужно оба слоя одновременно — управляемую блокировку 1С и блокировку СУБД, иначе видна только половина картины и решение принимается вслепую.

Производство: исчерпание лицензий из-за зависших rghost

Что настраиваем

Клиент ITfresh — производственная компания, 1С:ERP, 2 рабочих сервера кластера, PostgreSQL 15

Как мы это делаем

- 1 Настраиваем шаблон PostgreSQL by Zabbix agent 2 — снимаем connections и locks, а число ожидающих в pg_stat_activity — кастомным SQL-запросом.
- 2 Пишем UserParameter поверх рас process list, который парсит memory-current и connections каждого rghost и долю занятой памяти от лимита.
- 3 Заводим триггер: connections=0 и memory-current выше 70% лимита дольше 10 минут — верный признак зависшего процесса без пользователей.
- 4 Связываем триггер с эскалацией дежурному: при connections=0 сеансов терять нечего, поэтому зависший rghost снимается на уровне ОС (kill pid) — кластер сам поднимает...
- 5 Дополнительно ставим item на число бэкендов, ожидающих чужую блокировку дольше deadlock_timeout, через pg_blocking_pids().

РЕЗУЛЬТАТ

Зависшие rghost перестали занимать лицензии до утра — дежурный получает алерт ночью и снимает процесс за 5 минут, вместо разбора в 9:00, когда все места лицензий уже заняты пользователями.

КЛЮЧЕВОЙ НЮАНС

Единственный надёжный признак зависания rghost без остановки штатными средствами — connections=0 при высокой занятой памяти процесса.



Юрфирма: медленные отчёты и APDEX ниже норматива

Что настраиваем

Клиент ITfresh — юридическая фирма, 1С:Документооборот, тонкий клиент, MS SQL Server Express

Как мы это делаем

- 1 Включаем подсистему БСП «Оценка производительности» и настраиваем выгрузку замеров APDEX в таблицу СУБД для сбора Zabbix.
- 2 Задаём целевое время T0 по методике APDEX: 3 сек для проведения документов, 5 сек для формирования отчётов — как в типовых конфигурациях.
- 3 Триггер срабатывает, когда APDEX по группе «Отчёты» опускается ниже 0.85 (граница зоны «хорошо») на протяжении 15 минут подряд.
- 4 Параллельно включаем технологический журнал с фильтром по событию DBMSSQL и порогом длительности, чтобы видеть конкретный медленный запрос.
- 5 По алерту дежурный сверяет план запроса в SQL Server Management Studio и точно обновляет статистику или индекс до потока жалоб пользователей.

РЕЗУЛЬТАТ

APDEX по ключевым отчётам вернулся выше 0.94 (зона «отлично») за счёт того, что деградацию стали ловить по тренду метрики, а не по субъективным жалобам «всё тормозит».

КЛЮЧЕВОЙ НЮАНС

APDEX без привязки к конкретному событию техжурнала — просто цифра; ценность появляется, когда триггер сразу указывает на SQL-запрос, который её испортил.

Подводные камни

✗ Мониторинг только ОС, без RAS/rac

Zabbix видит, что rphost жив и CPU в норме, но не видит зависший сеанс внутри кластера — нужен отдельный слой через RAS.

✗ Открытый порт RAS без ограничений

RAS по умолчанию слушает порт 1545 без обязательного пароля администратора кластера — открытый наружу порт становится вектором атаки.

✗ Игнорирование управляемых блокировок 1C

Блокировка на уровне СУБД может отсутствовать, а транзакция стоит из-за управляемой блокировки 1C — её видно только в техжурнале.

✗ Слишком частый опрос rac

Опрос сеансов раз в 5-10 секунд создаёт лишнюю нагрузку на RAS и искажает статистику соединений на нагруженном кластере.

✗ Триггер без учёта фоновых заданий

BackgroundJob регулярно держит сеанс часами — без исключения из триггера дежурный получает лавину ложных алертов.

✗ Нет связки item СУБД с сеансом 1C

blocking_session_id в MS SQL не сопоставлен с UUID сеанса 1C — блокировка видна в базе, а виновника в кластере найти нельзя.

✗ Забыли про лицензии при мониторинге памяти

Автоперезапуск по превышению памяти обрывает все активные лицензии клиентов — нужен алерт раньше самого перезапуска.

✗ APDEX без разбивки по операциям

Общий APDEX по всей базе сглаживает деградацию одного тяжёлого отчёта — считать нужно отдельно по ключевым операциям.

Как правильно

МИНИМУМ

- Шаблон Zabbix agent 2 и UserParameter поверх rsc session list и rsc process list
- Триггер на connections=0 при памяти rghost выше 70% лимита дольше 10 минут
- Алерт дежурному при появлении blocking_session_id в MS SQL или PostgreSQL

НОРМАЛЬНО

- Item и триггер на управляемые блокировки 1С через техжурнал TLOCK/TDEADLOCK
- MSSQL by ODBC или PostgreSQL by Zabbix agent 2 с опросом DMV и pg_stat_activity
- Расчёт APDEX по ключевым операциям через подсистему БСП с выгрузкой в СУБД
- Дашборд кластера 1С: сеансы, лицензии, память rghost, число рабочих процессов

ХОРОШО

- Автосопоставление сеанса 1С с blocking_session_id СУБД по SPID/UUID на дашборде
- Действия Zabbix по эскалации: шаблонная команда rsc session terminate и снятие завис...
- Долгосрочный тренд APDEX и памяти rghost для планирования апгрейда кластера
- Регламентная проверка таймаутов блокировки под нагрузочный профиль клиента

Чек-лист самопроверки

- RAS запущен на боевом сервере кластера 1С, порт администрирования закрыт от внешнего доступа
- Шаблон Zabbix agent 2 с UserParameter на рас session/process list опрашивается не чаще 60 секунд
- Триггер на сеансы с connections=0 при памяти rphost выше 70% лимита настроен и проверен
- MSSQL by ODBC или PostgreSQL by Zabbix agent 2 подключены, DMV и pg_stat_activity опрашиваются
- Триггер на blocking_session_id и wait_time (MS SQL), pg_blocking_pids() (PostgreSQL) выше 30 сек
- Технологический журнал 1С включён с фильтром TLOCK/TDEADLOCK и порогом длительности
- APDEX по ключевым операциям считается отдельно, целевые T0 согласованы с заказчиком
- Действия Zabbix эскалируют алерт дежурному с UUID сеанса или SPID для быстрого разбора
- Регламентные задания BackgroundJob исключены из триггеров по долгим сеансам
- Дашборд кластера 1С с сеансами, лицензиями, памятью и блокировками доступен дежурной смене

Если хотя бы на два вопроса ответ «нет» или «не знаю» — тема требует внимания.



Как поможет ITFresh

ITFresh — ИТ-аутсорсинг для юридических лиц до 50 рабочих мест в Москве и области. 15+ лет практики, собственная инфраструктура в дата-центре МТС (8 серверов Dell Xeon Platinum).

- Разворачиваем и настраиваем RAS/rac, шаблоны Zabbix agent 2 и UserParameter под кластер и СУБД клиента
- Пишем триггеры и дашборды, связывающие сеансы 1С с блокировками MS SQL/PostgreSQL в единую картину
- Настраиваем технологический журнал и подсистему оценки производительности APDEX под операции клиента
- Берём дежурство по алертам Zabbix и снимаем блокировки или зависшие процессы до жалобы пользователей
- Проводим аудит существующего мониторинга 1С и закрываем слепые зоны между кластером, СУБД и ОС

15+

лет в ИТ-поддержке

50

рабочих мест — наш профиль

МТС

дата-центр, Москва



КОНТАКТЫ

Обсудить вашу задачу

Сайт **itfresh.ru**

Телефон **+7 903 729-62-41**

Telegram **@ITfresh_Boss**

Бесплатно посмотрим вашу инфраструктуру по этому чек-листу и скажем, где тонко — без обязательств.



itfresh.ru

Техническая база

- 01** Zabbix agent 2 — конфигурация и UserParameter (zabbix.com — 2026)
- 02** 1C:Предприятие 8.3 — Глава 5. Администрирование (rac/ras) (its.1c.ru — 2026)
- 03** Оценка интегральной производительности по методике APDEX (its.1c.ru — 2026)
- 04** sys.dm_exec_requests / sys.dm_tran_locks (Transact-SQL) (learn.microsoft.com — 2026)
- 05** Просмотр информации о блокировках: pg_locks, pg_stat_activity (postgrespro.ru — 2026)
- 06** Ограничение объёма памяти рабочих процессов кластера 1C (v8.1c.ru — 2026)
- 07** MSSQL by ODBC / PostgreSQL by Zabbix agent 2 — шаблоны (git.zabbix.com — 2026)
- 08** Технологический журнал: события TLOCK, TDEADLOCK, TTIMEOUT (its.1c.ru — 2026)

Основано на официальной документации продуктов и нашей практике внедрения.

