



Ай-ТИ Фреш

ТЕХНИЧЕСКИЙ РАЗБОР

1С на Linux с PostgreSQL: экономия на лицензиях и скорость

Кластер 1С:Предприятие 8.3 на Debian 12 с Postgres Pro 1С
— наша методология внедрения

Июль 2026

itfresh.ru · ИТ-аутсорсинг для юридических лиц

Суть проблемы

Клиент платит за Windows Server, MS SQL и CAL сотни тысяч рублей при 20-50 пользователях 1С, а база при этом тормозит: отчёты идут минутами, проведение документов блокирует коллег. Мы переносим серверную 1С на Debian 12 + Postgres Pro 1С: лицензии ОС и СУБД — 0 ₽, а тюнинг СУБД даёт запас скорости. Самодельная миграция грозит потерей данных и днями простоя учёта.

Почему это важно бизнесу

- Экономия: Windows Server + MS SQL Standard + CAL на 30-50 мест — сотни тысяч рублей; Linux и PostgreSQL — 0 ₽ за лицензии
- Простой бухгалтерии в закрытие периода — срыв отчётности в ФНС и штрафы; мы мигрируем с проверенным планом отката
- Тормозящая 1С крадёт время каждого сотрудника: минута на проведение документа × 35 человек — часы фонда оплаты труда в день
- Продление лицензий Microsoft официально недоступно; Linux-стек легален, бессрочен и не зависит от вендора
- Повреждённый бэкап обнаруживается в аварию: без проверки восстановимости потеря учёта за недели — реальный сценарий



Ключевые параметры реализации

8.3.27

Актуальная платформа: единый
инсталлятор setup-full,
systemd-юнит
srv1cv8-<версия>@default
по докам 1C:Предприятие

1C-17

Сборка Postgres Pro для 1C с
патчами online_analyze и plantuner
— вместо ванильного PostgreSQL
Postgres Pro 1C 17

25% RAM

shared_buffers от памяти узла СУБД
(доки: не более 40%);
effective_cache_size закладываем
75%
доки Postgres Pro + наш стандарт

1.1

random_page_cost для SSD/NVMe
вместо дефолтных 4.0 —
планировщик выбирает индексы
наш стандарт

1545/tcp

Порт RAS: управление и
мониторинг кластера 1C утилитой
rac по SSH, без консоли Windows
по докам 1C:Предприятие

30 дней

Ретенция ежедневных pg_dump -Fc;
каждый дамп автоматически
проверяем pg_restore -l
наш стандарт



Узел СУБД: Postgres Pro 1C на Debian 12

Что настраиваем

Выделенный сервер/VM под СУБД: инициализация кластера, тюнинг postgresql.conf, сетевой доступ

Как мы это делаем

- 1 Подключаем репозиторий 1c-17 Postgres Pro (pgpro-repo-add.sh), ставим postgrespro-1c-17 — сборку с патчами под 1C; ванильный PostgreSQL из apt не используем
- 2 Инициализируем кластер: /opt/pgpro/1c-17/bin/pg-setup initdb с локалью ru_RU.UTF-8 — строго до создания баз, иначе ошибки сортировки в отчётах
- 3 postgresql.conf: shared_buffers=25% RAM, effective_cache_size=75%, work_mem 64-256MB, maintenance_work_mem=2GB, checkpoint_completion_target=0.9
- 4 Под NVMe: random_page_cost=1.1, effective_io_concurrency=200, max_wal_size=8GB; synchronous_commit=on — учётными данными не рискуем
- 5 pg_hba.conf: доступ только с IP сервера 1C по scram-sha-256, listen_addresses — внутренний интерфейс, порт 5432 снаружи закрыт

РЕЗУЛЬТАТ

СУБД сразу работает с русской сортировкой и адекватными планами под тяжёлые запросы 1C; типовые отчёты и проведение ускоряются относительно дефолтного конфига в разы — без лицензий MS SQL и без доводки после инцидентов.

КЛЮЧЕВОЙ НЮАНС

Опираемся на приложение «Настройка Postgres Pro для решений 1C»: сборка часть параметров выставляет сама, но shared_buffers, work_mem и autovacuum всё равно считаем под профиль конкретной базы.



Кластер 1С:Предприятие 8.3.27 и лицензирование

Что настраиваем

Узел приложений: ragent/rmngr/rphost, служба RAS, сетевой ключ HASP или программные лицензии

Как мы это делаем

- 1 Ставим платформу единым инсталлятором `setup-full-8.3.27.<сборка>-x86_64.run` (компоненты server, ws); недостающие библиотеки Debian 12 доставляем заранее
- 2 Регистрируем systemd-юнит `srv1cv8-8.3.27.<сборка>@default` (enable + start); в nftables открываем 1540 (ragent), 1541 (rmngr), 1560–1591 (rphost)
- 3 Поднимаем RAS на 1545 и работаем через ras: список сеансов, принудительное завершение, свойства кластера — всё по SSH
- 4 Сетевой HASP: пакет haspd на узле с ключом + 475/tcp и 475/udp в фаерволе; программные лицензии привязываем к стабильным параметрам сервера
- 5 В консоли кластера задаём интервал перезапуска rphost и предел памяти рабочих процессов — страховка от деградации долгих сеансов

РЕЗУЛЬТАТ

Сервер приложений стартует автоматически, порты предсказуемы для фаервола, лицензии видны всем рабочим станциям; администрирование кластера полностью удалённое — консоль Windows не нужна.

КЛЮЧЕВОЙ НЮАНС

Единый дистрибутив .run — с 8.3.20, с 8.3.21 сервер живёт на systemd: юнит версионирован (`srv1cv8-<версия>@default`), обновление платформы = установка новой версии рядом и переключение юнита, откат занимает мин...



Бэкапы, регламент СУБД и мониторинг

Что настраиваем

Оба узла: дампы PostgreSQL, autovacuum, сеансы кластера через RAS, техжурнал платформы

Как мы это делаем

- 1 Ночной `pg_dump -Fc` по каждой базе + еженедельный `pg_dumpall --globals-only`; ретенция 30 дней, вторая копия уходит на внешний узел
- 2 Каждый дамп сразу проверяем `pg_restore -l`; раз в месяц — полное контрольное восстановление на тестовый инстанс с открытием базы в 1C
- 3 `autovacuum` не отключаем: `autovacuum_max_workers=4`, для крупных таблиц регистров снижаем `autovacuum_vacuum_scale_factor` до 0.01
- 4 Мониторим `pg_stat_activity` и `pg_stat_statements` + `rac session list`; алерт в Telegram при транзакции длиннее 10 минут
- 5 Техжурнал 1C (`logcfg.xml`) включаем точно — события EXCP и таймауты блокировок, с ротацией, чтобы не заполнить диск

РЕЗУЛЬТАТ

RPO — сутки по дампам (с WAL-архивом — минуты), восстановимость подтверждается регулярно, распухание таблиц и зависшие сеансы видим до жалоб пользователей.

КЛЮЧЕВОЙ НЮАНС

Бэкап без контрольного восстановления — лотерея: повреждённый дамп находят в момент аварии. Проверку `pg_restore -l` встраиваем в тот же cron-скрипт, который делает сам дамп.



Подводные камни

✗ Ванильный PostgreSQL из apt

Без патчей под 1C — ошибки сортировки и медленные планы. Ставим только сборку Postgres Pro 1C или сборку с сайта 1C.

✗ Локаль после initdb

Кластер, инициализированный в C-локали, не переделать на лету. Переинициализируем initdb --locale=ru_RU.UTF-8 до загрузки баз.

✗ work_mem «на глазок» вверх

work_mem выделяется на операцию в каждой сессии: 1GB × десятки соединений = OOM-killer. Считаем от max_connections, отчётам — точно.

✗ Отключённый autovacuum

Таблицы регистров распухают, планы деградируют за недели. Не выключаем, а настраиваем пороги scale_factor и число воркеров.

✗ Бэкап без проверки

Повреждённый дамп обнаруживают в аварию. Автоматический pg_restore -l после каждого дампа + ежемесячное тестовое восстановление.

✗ HASP не виден по сети

Забыв haspd или закрыт порт 475. Ставим haspd на узел с ключом и открываем 475/tcp+udp в фаерволе для подсети клиентов.

✗ Выгрузка .dt как бэкап

.dt требует монопольного режима и не гарантирует загрузку обратно. Это инструмент переноса; резервная копия — только pg_dump.

✗ 1C и СУБД на одном узле без лимитов

rphost при тяжёлом отчёте отъедает память у shared_buffers. Разносим по узлам или жёстко лимитируем память рабочих процессов.

Как правильно

МИНИМУМ

- Сборка Postgres Pro 1C (postgrespro-1c-17), initdb с локалью ru_RU.UTF-8
- Тюнинг: shared_buffers 25% RAM, effective_cache_size 75%, random_page_cost 1.1
- Ежедневный pg_dump -Fc с ретенцией 30 дней и автопроверкой pg_restore -l
- systemd-юнит srv1cv8 в автозапуске, порты 1540/1541/1560-1591 под контролем фаервола

НОРМАЛЬНО

- Сервер 1C и СУБД на разных узлах; pg_hba.conf — scram-sha-256 с IP сервера 1C
- RAS + rac: перезапуск rphost по расписанию, лимиты памяти рабочих процессов
- Вторая копия бэкапов на внешнем узле + ежемесячное контрольное восстановление
- pg_stat_statements и алерты на транзакции >10 мин и распухание таблиц

ХОРОШО

- WAL-архив (pg_probackup/wal-g): восстановление на момент времени, RPO — минуты
- Тестовый контур: обновления платформы и конфигураций сначала на копии базы
- Техжурнал 1C (logcfg.xml) на EXCP и блокировки + контроль APDEX ключевых операций
- Поточная реплика PostgreSQL на второй узел для быстрого переключения при аварии



Чек-лист самопроверки

- | | |
|--|---|
| ☐ Стоит ли сборка PostgreSQL именно для 1C (Postgres Pro 1C), а не ванильная из репозитория дистрибутива? | ☐ Поднят ли RAS (1545) и может ли дежурный инженер снять список сеансов рас без графической консоли? |
| ☐ Инициализирован ли кластер СУБД с локалью ru_RU.UTF-8 до загрузки первой базы? | ☐ Ограничен ли доступ к PostgreSQL в pg_hba.conf только адресом сервера 1C по scram-sha-256? |
| ☐ Рассчитаны ли shared_buffers, effective_cache_size и work_mem под фактический RAM, а не оставлены по умолчанию? | ☐ Заданы ли перезапуск rphost по расписанию и предел памяти рабочих процессов в консоли кластера? |
| ☐ Делается ли ежедневный pg_dump -Fc и проверяется ли каждый дамп на восстановимость автоматически? | ☐ Проведён ли аудит нетиповых доработок на СОМ-объекты перед миграцией с Windows? |
| ☐ Хранится ли копия бэкапов вне сервера 1C — на втором узле или внешнем хранилище? | ☐ Есть ли план отката: сколько часов займёт возврат на старый сервер, если что-то пойдёт не так? |

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).

- Миграция 1С с Windows/MS SQL на Debian + Postgres Pro 1С под ключ: аудит, перенос, контрольная сверка данных
- Тюнинг postgresql.conf под ваш профиль нагрузки и RAM, профилирование и ускорение медленных запросов
- Трёхуровневое резервное копирование с автопроверкой восстановимости и хранением вне площадки
- Мониторинг кластера 1С через RAS и СУБД через pg_stat_statements с алертами в Telegram
- Поддержка по SLA: обновления платформы 8.3, регламент СУБД, реакция на инцидент до 1 часа

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** Клиент-серверный вариант. Руководство администратора (its.1c.ru — 8.3.27)
- 02** Системные требования «1С:Предприятия 8»: СУБД PostgreSQL (v8.1c.ru — 2026)
- 03** Приложение I. Настройка Postgres Pro для решений 1С (postgrespro.ru — 17)
- 04** PostgreSQL: Server Configuration — Resource Consumption, WAL (postgresql.org — 17)
- 05** Сборки PostgreSQL для 1С (репозиторий 1с-17) (repo.postgrespro.ru — 2026)
- 06** Наш шаблон postgresql.conf и регламент бэкапа под 1С (itfresh.ru — 2026)

Основано на официальной документации продуктов и нашей практике внедрения.

