Ай-ТИ Фреш

ТЕХНИЧЕСКИЙ РАЗБОР

K3s в офисе на 40 человек: маленький Kubernetes вместо Compose

Наш типовой кластер: 3 ноды K3s v1.36 с embedded etcd, Traefik v3, Longhorn и GitOps на Argo CD

Июль 2026

itfresh.ru · ИТ-аутсорсинг для юридических лиц

О ЧЁМ ЭТОТ РАЗБОР

Суть проблемы

У клиента с собственной разработкой (3–5 человек) сервисы живут на трёх VPS с Docker Compose: релиз — 40 минут ручной работы админа, упавший контейнер тянет соседей по хосту, отката нет, пароли лежат в compose-файлах в git, о падении узнают от клиентов. Мы переводим такой стек на K3s: rolling-деплои, самовосстановление подов, откат за 30 секунд и GitOps — на тех же трёх VPS.

Почему это важно бизнесу

- Релиз без админа: от merge в main до прода — минуты вместо 40 минут ручной работы и очереди к инженеру
- Пятничные инциденты исчезают: сломанный релиз откатывается командой kubectl rollout undo за 30 секунд
- Падение сервиса чинится само: K3s пересоздаёт под и уводит нагрузку с отказавшей ноды без звонка инженеру
- Пароли уходят из git: утечка репозитория перестаёт означать утечку доступов к базе клиентов
- Экономия ~80 часов работы админа в год на релизах и инцидентах — проект окупается примерно за 3 месяца

ЧТО МЫ ЗАКЛАДЫВАЕМ

Ключевые параметры реализации

v1.36.2

канал stable K3s: Kubernetes v1.36, containerd v2.3, Traefik v3.7, CoreDNS v1.14 — один бинарн...

docs.k3s.io, release notes v1.36.X

3

server-ноды с embedded etcd (--cluster-init): кворум 2 из 3 переживает отказ любой одной ноды

docs.k3s.io, HA Embedded etcd

12 ч

интервал снапшотов etcd по умолчанию (cron 0 */12), ретенция 5; копию уводим в S3 ключами --et...

docs.k3s.io, Backup and Restore

4vCPU/8ГБ

наш стандарт ноды под прод (минимум по докам — 2 CPU / 2 ГБ на server); диски — только NVMe/SSD

наш стандарт, docs.k3s.io Requirements

×3

реплики каждого тома Longhorn на трёх нодах: при отказе ноды под переезжает к живой реплике

longhorn.io, defaults

3 мин

период опроса git в Argo CD (timeout.reconciliation, 180s): изменение манифеста само докатывае...

argo-cd docs, наш стандарт

РЕАЛИЗАЦИЯ 1 ИЗ 3

Ядро: три server-ноды K3s с embedded etcd

Что настраиваем

Три VPS 4 vCPU / 8 ГБ / NVMe у российского хостера; каждая нода — control-plane + worker

Как мы это делаем

1. Первая нода: curl -sfL https://get.k3s.io | INSTALL_K3S_CHANNEL=stable sh -s - server --cluster-init --secrets-encryption
2. Вторая и третья: sh -s - server --server https://node1:6443 с токеном из /var/lib/rancher/k3s/server/node-token — кворум etcd из трёх нод
3. Снапшоты состояния: --etcd-snapshot-schedule-cron "0 */12 * * *", --etcd-snapshot-retention 5 + выгрузка в S3 (--etcd-s3 --etcd-s3-endpoint/-bucket)
4. registries.yaml в /etc/rancher/k3s/: зеркала и креды приватного Harbor для containerd на всех нодах
5. Доступ: kubeconfig /etc/rancher/k3s/k3s.yaml, инженерам — именные сертификаты и RBAC; порт 6443 открыт только с наших админ-IP

РЕЗУЛЬТАТ

Control-plane и API переживают отказ любой из трёх нод; состояние кластера восстанавливается из S3-снапшота командой k3s server --cluster-reset --cluster-reset-restore-path. Развёртывание всех трёх нод с нуля занимает у нас меньше часа.

КЛЮЧЕВОЙ НЮАНС

Дефолтный datastore — SQLite через kine: быстрее на одном server, но без HA. Embedded etcd включается только флагом --cluster-init на первой ноде; server-нод должно быть нечётное число — две хуже одной.

РЕАЛИЗАЦИЯ 2 ИЗ 3

Ingress, TLS, хранилище и секреты

Что настраиваем

Платформенный слой поверх кластера: Traefik v3, cert-manager, Longhorn, Sealed Secrets

Как мы это делаем

1. Встроенный Traefik v3.7 настраиваем не правкой чарта, а через HelmChartConfig в /var/lib/rancher/k3s/server/manifests — переживает апгрейды K3s
2. cert-manager v1.20 + ClusterIssuer Let's Encrypt (ACME HTTP-01 через Traefik): сертификаты всех доменов клиента продлеваются сами
3. Longhorn на трёх нодах: numberOfReplicas=3, пакет open-iscsi на каждой ноде, StorageClass longhorn по умолчанию вместо local-path
4. Секреты: включён --secrets-encryption (шифрование Secrets на уровне datastore) + Sealed Secrets — в git лежит только зашифрованный SealedSecret
5. Прод-PostgreSQL выносим на отдельный VPS со своими бэкапами — тяжёлую stateful-СУБД в маленьком кластере не держим

РЕЗУЛЬТАТ

HTTPS всех доменов продлевается автоматически, том при отказе ноды переподключается к живой реплике Longhorn за десятки секунд, боевые пароли нельзя достать ни из git-репозитория, ни из дампа datastore.

КЛЮЧЕВОЙ НЮАНС

Traefik конфигурируем только через HelmChartConfig — прямые правки затираются обновлением K3s. Longhorn требует open-iscsi и SSD/NVMe: на HDD тройная репликация превращает IO подов в минуты ожидания.

РЕАЛИЗАЦИЯ 3 ИЗ 3

GitOps-конвейер и наблюдаемость

Что настраиваем

CI/CD и мониторинг: Gitea + Act Runner, Harbor, Argo CD, kube-prometheus-stack, Loki, Velero

Как мы это делаем

1. Сборка: Act Runner собирает образ, тег — только sha коммита (никаких latest), пуш в Harbor со сканированием уязвимостей Trivy
2. Argo CD: Application с automated sync + prune + selfHeal; git-репозиторий манифестов — единственный источник состояния прода
3. kube-prometheus-stack: алерты в Telegram на CrashLoopBackOff, заполнение PVC >80%, недоступность ноды; Loki собирает логи всех подов
4. Velero + node-agent: ежедневный бэкап манифестов и PV в S3 в дополнение к etcd-снапшотам
5. system-upgrade-controller: Plan на канал stable — обновление K3s по одной ноде с cordon/drain, без остановки сервисов

РЕЗУЛЬТАТ

Релиз от merge до прода — 3–5 минут без участия админа; любое ручное изменение в кластере Argo CD возвращает к состоянию git (selfHeal). Инцидент виден в Telegram раньше, чем о нём позвонит клиент.

КЛЮЧЕВОЙ НЮАНС

GitOps работает только с immutable-тегами: на image:latest Argo CD не увидит новый образ. Диск под Prometheus и Loki считаем заранее: retention 15 дней на трёх нодах — это ещё ~20–30 ГБ хранилища.

ЧТО МЫ ВЫУЧИЛИ НА ПРАКТИКЕ

Подводные камни

- ✕ **K3s без команды разработки**
  Меньше 8 сервисов, релизы реже раза в неделю, некому жить в kubectl — кластер станет чёрным ящиком одного админа. Остаёмся на Compose; 1С и СУБД не т…

- ✕ **SQLite-datastore на растущем кластере**
  Дефолтный kine+SQLite — только для одного server. Сразу ставим --cluster-init с embedded etcd: миграция позже дороже переустановки.

- ✕ **Тег latest в манифестах**
  Argo CD не видит новый образ под тем же тегом — деплой «не едет». Только immutable-теги по sha коммита, которые проставляет CI.

- ✕ **Чётное число server-нод**
  Два узла etcd хуже одного: потеря любого — потеря кворума и отказ API. Держим 1 или 3 server-ноды, воркеры — сколько угодно.

- ✕ **Longhorn на медленных дисках**
  Тройная репликация множит IO: на HDD поды ждут тома до минуты. Только NVMe/SSD и отдельный раздел под /var/lib/longhorn.

- ✕ **Secrets «как есть» — это base64**
  Base64 — не шифрование. Включаем --secrets-encryption в K3s, в git кладём только SealedSecret, боевые пароли — Sealed Secrets/Vault.

- ✕ **Открытые порты кластера наружу**
  6443, 2379-2380, 10250 из интернета недопустимы: фаервол, API только с админ-IP/VPN, VXLAN 8472/udp — только между нодами.

- ✕ **Поды без requests/limits**
  Один прожорливый под съедает ноду, kubelet начинает evict соседей. Ставим requests/limits каждому деплойменту + PodDisruptionBudget.

УРОВНИ РЕАЛИЗАЦИИ

Как правильно

МИНИМУМ

- Один server K3s канала stable + worker; бэкап /var/lib/rancher/k3s по расписанию
- Встроенный Traefik + cert-manager с Let's Encrypt на все внешние сервисы
- Манифесты в git, деплой из CI, теги образов только по sha коммита
- API-порт 6443 закрыт фаерволом, доступ только с админ-IP

НОРМАЛЬНО

- 3 server-ноды с embedded etcd (--cluster-init), снапшоты etcd в S3 каждые 12 ч
- Longhorn с тремя репликами на NVMe; --secrets-encryption + Sealed Secrets
- Argo CD с automated sync + selfHeal, приватный registry Harbor
- kube-prometheus-stack + Loki, алерты в Telegram дежурному

ХОРОШО

- system-upgrade-controller: авто-обновление K3s по каналу stable с drain нод
- Velero + node-agent: ежедневный бэкап PV и манифестов в S3 вне площадки
- NetworkPolicy между namespace, именные сертификаты и RBAC для инженеров
- Квартальный тест восстановления кластера из снапшота на чистых VPS

ПРОВЕРЬТЕ У СЕБЯ

Чек-лист самопроверки

- [ ] Переживёт ли кластер отказ одной ноды без остановки сервисов (3 server, кворум etcd)?
- [ ] Снимаются ли снапшоты etcd каждые 12 часов и уходят ли копии в S3 за пределы кластера?
- [ ] Все ли образы деплоятся с immutable-тегом по коммиту — в манифестах нет image:latest?
- [ ] Закрыт ли порт API 6443 фаерволом и доступен ли он только с админ-IP или через VPN?
- [ ] Можно ли откатить неудачный релиз одной командой (kubectl rollout undo / Argo CD rollback)?
- [ ] Лежат ли боевые пароли вне git (Sealed Secrets/Vault) и включён ли --secrets-encryption?
- [ ] Стоят ли requests/limits у каждого пода и настроен ли алерт на CrashLoopBackOff?
- [ ] Проверялось ли восстановление из бэкапа Velero/etcd-снапшота за последние 3 месяца?
- [ ] Вынесена ли прод-СУБД из кластера на выделенный сервер со своими бэкапами?
- [ ] Обновляется ли K3s по каналу stable, а не заморожен на версии годовой давности?

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

ЧТО МЫ ДЕЛАЕМ

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).

- Аудит текущего Docker Compose-стека и честный вердикт: нужен ли вам K3s вообще
- Кластер под ключ за 7 рабочих дней: 3 ноды, embedded etcd, Longhorn, мониторинг, GitOps
- Миграция приложений из compose-файлов в манифесты + CI/CD на Gitea, Harbor и Argo CD
- Обучение команды разработки: kubectl, манифесты, GitOps-процесс — 4 часа, входит в проект
- Сопровождение на абонентке: обновления K3s, реакция на алерты, тесты восстановления

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

ДОКУМЕНТАЦИЯ И НАШ СТЕК

Техническая база

01 K3s Docs — Installation Requirements / Architecture (docs.k3s.io — v1.36)

02 K3s Docs — High Availability Embedded etcd (--cluster-init) (docs.k3s.io — v1.36)

03 K3s Docs — Backup and Restore, CLI etcd-snapshot (S3, cluster-reset) (docs.k3s.io — v1.36)

04 K3s Docs — Networking Services, Traefik via HelmChartConfig (docs.k3s.io — v1.36)

05 K3s Release Notes v1.36.X — версии containerd, Traefik, CoreDNS (docs.k3s.io — 2026)

06 Longhorn Docs — Node Requirements / Best Practices (longhorn.io — 1.11)

07 Argo CD Docs — Automated Sync Policy, selfHeal/prune (argo-cd.readthedocs.io — 3.x)

08 Наш шаблон кластера ITfresh: k3s-3node + GitOps baseline (itfresh.ru — 2026)

Основано на официальной документации продуктов и нашей практике внедрения.