

Проекты VK WorkSpace

Версия 4.0

Оглавление

Обновление в docker compose	3
Обновление в Kubernetes	3

ВНИМАНИЕ

Настоятельно рекомендуется соблюдать последовательный порядок обновления версий.

Правильный подход: Обновляться строго с предыдущей версии на следующую (Версия N → Версия N+1).

Не рекомендуется: Пропускать версии при обновлении (например, Версия N → Версия N+2).

Добавлены новые сервисы:

- gotenberg;
- tenant_service_api;
- init_database_tool;
- toolbox.

ВНИМАНИЕ

Обязательно выполните резервное копирование БД и необходимых томов (minio, ssl volumes, etc)

Пожалуйста убедитесь в наличии достаточного количества свободного пространства - в результате данного обновления размер БД будет временно увеличен в два раза, а minio 30% от общего размера дискового пространства на время миграции.

Узнать размер docker тома с БД `{{ shell_product_name }}_database-service-volume-v17` можно командой `du -sh` - адрес размещения по умолчанию `/var/lib/docker/volumes/`

Пожалуйста выделите дополнительное время на процесс миграции и обновления. Процесс миграции для БД размером 3 Gb занимает от 30 минут на рекомендованной конфигурации 4/8.

Обновление в docker compose

Перед обновлением рекомендуется сделать [резервную копию](#).

Для обновления схемы БД и minio выполнить миграцию:

1. Распаковать релиз и перенести свои значения в .env;
2. Выключить Проекты VK WorkSpace: `docker compose -p {{ shell_product_name }} down;`
3. Загрузить docker образы: `docker load -i images.tar;`
4. Запустить миграцию: `docker compose -p {{ shell_product_name }} --env-file .env -f scripts/migrate-tenant-schema-db.yml up -d;`
5. Дождаться завершения миграции, статус контейнеров `{{ shell_product_name }}-adjust_migration_tool-1` и `{{ shell_product_name }}-init_database_tool-1` должен быть Exited (0). Проверить логи контейнеров: `docker logs {{ shell_product_name }}-adjust_migration_tool-1` и `docker logs {{ shell_product_name }}-init_database_tool-1`. В результате должно быть сообщение Migrations finished/Миграции выполнены.

В случае наличия сообщений об ошибках в логах контейнеров - свяжитесь со службой технической поддержки и предоставьте логи контейнеров на анализ.
6. Удалить контейнеры с миграцией: `docker compose -p {{ shell_product_name }} --env-file .env -f scripts/migrate-tenant-schema-db.yml down`

Если миграция успешно выполнена, то можно переходить к установке новой версии релиза.

Обновление в Kubernetes

Версия helm-чарта: 4.0.

helm чарт был разделен на:

- `{{ shell_product_name }}-infra` - инфраструктурные сервисы;
- `{{ shell_product_name }}` - приложение;

Перед обновлением рекомендуется сделать [резервную копию](#).

Для обновления схемы БД и minio выполнить миграцию:

1. Перед установкой необходимо подключиться к репозиториям docker и helm или скачать комплект поставки с docker образами и helm чартами локально.
2. Создайте свой файл с параметрами для `{{ shell_product_name }}-infra` `values_{{ shell_product_name }}_infra.yaml` и добавьте минимальный набор параметров для установки.

3. Создайте свой файл с параметрами для `{{ shell_product_name }}`
`values_{{ shell_product_name }}.yaml` и добавьте минимальный набор параметров для установки.
4. Удалите существующий релиз: `helm -n NAMESPACE uninstall RELEASE` (**сохранив Persistent Volumes!**).
5. Установите `{{ shell_product_name }}-infra` указав свой `values_{{ shell_product_name }}-infra.yaml`.
По умолчанию `timeout = 60` минут. Если `helm` упал на установке по таймауту, то сама миграция продолжит работать. Проверить ее состояние `kubectl -n {{ shell_product_name }} get jobs`, она пропадет из списка как только будет завершена успешно. **Не перезапускайте установку, если миграция еще идет.**

```
# Для онлайн установки
helm upgrade --install -n {{ shell_product_name }} -f values_{{ shell_product_name }}_infra.yaml --version VERSION --wait --create-namespace {{ shell_product_name }}-infra {{ shell_product_name }}/{{ shell_product_name }}-infra --timeout 60m

# Для оффлайн установки
helm upgrade --install -n {{ shell_product_name }} -f values_{{ shell_product_name }}_infra.yaml --wait --create-namespace {{ shell_product_name }}-infra {{ shell_product_name }}-infra/ --timeout 60m
```

6. Включите миграцию для `{{ shell_product_name }}` в `values_{{ shell_product_name }}.yaml`:

```
adjustMigration:
  enabled: true
initMigration:
  legacyMigrate: "true"
```


7. Установите `{{ shell_product_name }}` указав свой `values_{{ shell_product_name }}.yaml`.

```
# Для онлайн установки
helm upgrade --install -n {{ shell_product_name }} -f values_{{ shell_product_name }}.yaml --version VERSION --wait --create-namespace {{ shell_product_name }} {{ shell_product_name }}/{{ shell_product_name }}

# Для оффлайн установки
helm upgrade --install -n {{ shell_product_name }} -f values_{{ shell_product_name }}.yaml --wait --create-namespace {{ shell_product_name }} {{ shell_product_name }}/
```

8. **Выключить миграцию в `values_{{ shell_product_name }}.yaml`:**

```
adjustMigration:
  enabled: false
initMigration:
  legacyMigrate: "false"
```

 6 мая 2026 г.