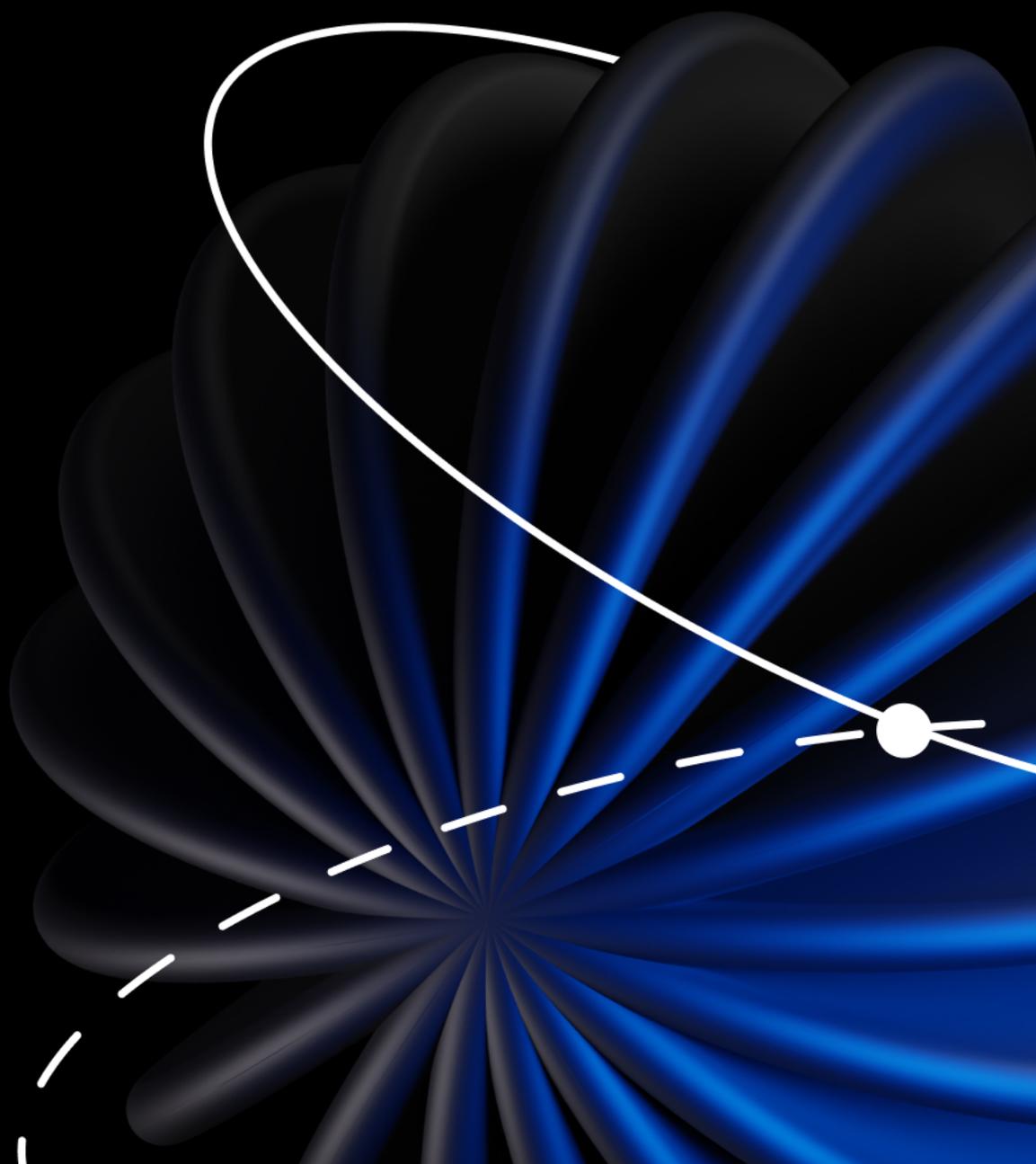




VK Budgeting

Руководство по установке

Версия 1.0.0



Оглавление

1	Состав релиза VK Budgeting	3
2	Требования к окружению	4
3	Версии системных компонент, необходимых для VK Budgeting	5
4	Доступ к образам VK Budgeting	6
5	Установка инфраструктурных сервисов и приложений	7
5.1	Установка PostgreSQL и других баз данных	7
5.2	Установка MonetDB	8
5.3	Подготовка кластера Kubernetes	8
6	Установка и запуск приложения VK Budgeting	10
6.1	Загрузка Helm-чартов	10
6.2	Настройка приложения	10

1 Состав релиза VK Budgeting

- Helm-чарты для сервисов.
- Образы для UI и backend.

2 Требования к окружению

Портал данных бизнеса поддерживает работу на следующих операционных системах: OS Astra Linux, РЕД ОС или Debian.

Для приложения рекомендуется использование Kubernetes-кластера, состоящего из воркер- и мастер-нод с данными минимальными характеристиками. Допускается расположение воркер и мастер нод на одном виртуальном сервере. Распределение приложение по серверам отражено в документе План технической инфраструктуры.

Воркер-ноды

Для стабильной работы приложения рекомендуется, чтобы суммарные ресурсы воркер-нод соответствовали следующим минимальным требованиям:

- **ЦПУ:** от 12 ядер CPU,
- **ОЗУ:** от 24 GB RAM.

Каждая воркер-нода должна иметь не менее 6 GB RAM, так как бекенд приложения может использовать около 4 GB памяти.

Мастер-ноды

Рекомендуется использовать минимум три мастер-ноды с конфигурацией не менее:

- **ЦПУ:** от 4 ядра CPU,
- **ОЗУ:** от 8 GB RAM.

3 Версии системных компонент, необходимых для VK Budgeting

- Kubernetes \geq v1.31
- Helm \geq v3.5
- Postgres = v15.8
- Kafka = v3.7.2
- MonetDB = v11.53.3
- Minio = minio:2024.10.13
- Keycloak = v26.2.0

4 Доступ к образам VK Budgeting

Helm-чарты, образы и бинарные файлы хранятся в репозитории Вендора Nexus <https://repo.taxmon.ru>. Для доступа необходимо запросить логин и пароль

5 Установка инфраструктурных сервисов и приложений

5.1 Установка PostgreSQL и других баз данных

Для хранения данных VK Budgeting используется база данных PostgreSQL v15.8. Установку рекомендуется проводить на отдельном виртуальном сервере вне кластера Kubernetes. После установки сервера БД необходимо создать базы данных и пользователей.

```
create user applier with password '{% applier_password %}';
create user audit with password '{% audit_password %}';
create user backend with password '{% backend_password %}';
create user checker with password '{% checker_password %}';
create user core_dashboard with password '{% core_dashboard_password %}';
create user core_macros with password '{% core_macros_password %}';
create user core_model with password '{% core_model_password %}';
create user data_editor with password '{% data_editor_password %}';
create user data_viewer with password '{% data_viewer_password %}';
create user db_manager with password '{% db_manager_password %}';
create user gvs with password '{% gvs_password %}';
create user keycloak with password '{% keycloak_password %}';
create user model_applier with password '{% model_applier_password %}';

create database applier with owner applier;
create database audit with owner audit;
create database backend with owner backend;
create database core_model with owner core_model;
create database db_manager with owner db_manager;
create database gvs with owner gvs;
create database keycloak with owner keycloak;
```

```
create database oltp with owner applier;
```

Где:

user — имя пользователя.

password — пароль пользователя.

database — наименование базы данных.

owner — владелец БД.

Во время настройки приложения укажите имя пользователя, пароль и название БД в файле values.yaml.

5.2 Установка MonetDB

Для хранения данных VK Budgeting используется база данных MonetDB v11.53.3. Установку рекомендуется проводить на отдельном виртуальном сервере вне кластера Kubernetes. После установки сервера БД необходимо создать базу данных.

```
monetdb create {{ stage }}db  
monetdb release {{ stage }}db
```

Где:

stage — название БД, оно же отображено в названии стенда.

5.3 Подготовка кластера Kubernetes

1. Подготовьте кластер Kubernetes версии 1.31 или выше.
2. Установите Helm версии 3.5 или выше.

5.3.1 Предварительная настройка кластера k8s

Для удобства заведите переменные окружения. DOCKERUSER и DOCKERPASSWORD указаны выше в документе:

```
export KUBECONFIG={% path/to/kubeconfig %}  
export NAMESPACE=budget  
export DOCKERUSER=myuser  
export DOCKERPASSWORD=mypassword
```

Пример создания namespace и secret для доступа к приватному репозиторию:

```
kubectl create namespace $NAMESPACE;
kubectl create secret docker-registry registrysecret --docker-server=registry-budgeting.taxmon.ru
--docker-username=$DOCKERUSER --docker-password=$DOCKERPASSWORD -n $NAMESPACE --
dry-run=client -o yaml | kubectl apply -f -;
kubectl patch serviceaccount default -n $NAMESPACE -p '{"imagePullSecrets": [{"name":
"registrysecret"}]}'
```

5.3.2 Установка ingress-nginx-controller

Выбор ingress controller зависит от специфики окружения.

Пример установки ingress-nginx-controller:

```
helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx;
helm repo update;
kubectl create namespace ingress-nginx
helm install ingress-nginx ingress-nginx/ingress-nginx --namespace ingress-nginx -f nginx-values.yml
--version 4.0.17;
```

5.3.3 Создание секретов для tls

Пример создания секрета для tls:

```
kubectl create secret tls {% tls-secret-name %} --namespace $NAMESPACE --cert=path/to/tls.crt --
key=path/to/tls.key
```

Бекенд приложения и фронтенд используют один и тот же домен. Для них будет один и тот же секрет. Названия секретов необходимо указать в values.yaml для сервисов.

6 Установка и запуск приложения VK Budgeting

6.1 Загрузка Helm-чартов

Для загрузки Helm-чартов для установки компонентов VK Budgeting воспользуйтесь одним из следующих способов:

1. Через веб-интерфейс Nexus Вендора:
 - a. Авторизуйтесь в Nexus с помощью ваших учетных данных.
 - b. Перейдите по ссылке на нужный чарт и скачайте его.
2. Через командную строку с помощью Helm CLI:
 - a. Добавьте репозиторий Helm, используя команду:

```
helm repo add nm https://repo.taxmon.ru/repository/nm-helm-registry/ --username $DOCKERUSER --password $DOCKERPASSWORD
```

- b. Обновите индекс репозитория:

```
helm repo update nm
```

- c. Скачайте необходимый чарт:

```
helm fetch nm/VKBGT --untar
```

6.2 Настройка приложения

Для настройки параметров приложения перейдите в директорию полученного чарта. Например:

```
cd VKBGT
```

Конфигурация приложения осуществляется в четырех файлах:

1. `values.yaml` — содержит общие настройки для приложения. Данный файл заполняется по результату предыдущих действий и является основным файлом конфигурации приложения. Настройки `keycloak client_id`, `secret`, `realm` остаются без изменений и могут быть заменены после завершения процесса настройки `keycloak`.
2. `values-budget-images.yaml` — содержит ссылки на нужные образы приложения и их версии. Укажите версии образов, которые будут использоваться в ключе `tag`,

3. `values-budget-resources.yaml` — содержит настройки ресурсов для приложения, заполнен уже по умолчанию, при необходимости можно изменить,
4. `values-secret-budget.yaml` — содержит чувствительные переменные конфигурации приложений.

Пример команды:

```
helm upgrade --install budget . -f values.yaml -f values-budget-images.yaml -f values-budget-resources.yaml -f values-secret-budget.yaml
```

После развертывания всех компонентов:

1. Откройте в браузере адрес `https://<YOURDOMAIN>`
2. Логин и пароль по умолчанию: `admin/123456`.