

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ
ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

«ПЛАТФОРМА БОЦМАН КЛИК»

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

История изменений

Версия	Дата	Комментарий	Автор
1.0.0	13.03.2023	Создание документа, редаKTура	Кижменёв Артем
1.0.0	13.03.2023	Создание документа	Повалкин Дмитрий

Оглавление

ГЛОССАРИЙ	5
1. АННОТАЦИЯ.....	7
2. ОБЩАЯ ИНФОРМАЦИЯ.....	7
1.1 ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ ДЛЯ ЗАПУСКА УСТАНОВЩИКА.....	7
1.2 КОМПОНЕНТЫ УСТАНОВЩИКА.....	7
1.3 ТРЕБОВАНИЯ К СРЕДЕ УСТАНОВКИ И ФУНКЦИОНИРОВАНИЯ ПЛАТФОРМЫ.....	7
1.4 ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ К УЗЛАМ.....	8
1.5 ТРЕБОВАНИЯ К ТОПОЛОГИИ СЕТИ.....	9
1.6 ТРЕБОВАНИЯ ДОСТУПНОСТИ РЕСУРСОВ.....	9
1.7 УСТАНОВЛИВАЕМОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ НА УЗЛЫ КЛАСТЕРА.....	11
1.8 ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, УСТАНОВЛИВАЕМОЕ НА УЗЛЫ ПО РОЛЯМ.....	12
2 УСТАНОВКА ПЛАТФОРМЫ.....	13
2.1 ПОДГОТОВКА ПЛОЩАДОК ДЛЯ УСТАНОВКИ ПЛАТФОРМЫ*	13
2.2 КОНФИГУРАЦИОННЫЙ ФАЙЛ	13
2.3 ЗАПУСК И ПРОЦЕСС УСТАНОВКИ.....	15
3 НАЧАЛО РАБОТЫ И ОБСЛУЖИВАНИЕ КЛАСТЕРА.....	16
3.1 ДОБАВЛЕНИЕ УЗЛОВ В КЛАСТЕР	16
3.2 УДАЛЕНИЕ УЗЛОВ.....	17
3.3 УДАЛЕНИЕ КЛАСТЕРА	18
4 ИНТЕРФЕЙС УПРАВЛЕНИЯ.....	18
4.1 ПОДГОТОВКА К РАБОТЕ.....	18
4.2 ПЕРВЫЙ ВХОД И НАЧАЛО РАБОТЫ.....	19
4.3 КЛАСТЕР	21
4.4 РАБОЧИЕ ПРОЦЕССЫ.....	25
4.5 ПРИЛОЖЕНИЯ.....	28
4.6 ОБЗОР СЕРВИСА	30
4.7 ХРАНИЛИЩЕ	32
4.8 МОНИТОРИНГ	34
4.9 CILLIUM	34
4.10 LONGHORN.....	36
4.11 ДОПОЛНИТЕЛЬНЫЕ РЕСУРСЫ	37
4.12 BILLING.....	38
4.13 ИНСТРУМЕНТЫ КЛАСТЕРА.....	38
5 ИНТЕРФЕЙС УПРАВЛЕНИЯ. ГЛОБАЛЬНЫЕ ПРИЛОЖЕНИЯ	40
5.1 БЕСПРЕРЫВНАЯ ДОСТАВКА.....	40
5.2 УПРАВЛЕНИЕ КЛАСТЕРОМ.....	41
5.3 УПРАВЛЕНИЕ ВИРТУАЛИЗАЦИЕЙ	42
6 ИНТЕРФЕЙС УПРАВЛЕНИЯ. КОНФИГУРАЦИЯ	42

Информация, необходимая для эксплуатации программного обеспечения

6.1	ПОЛЬЗОВАТЕЛИ И АУТЕНТИФИКАЦИЯ	43
6.2	РАСШИРЕНИЯ	44
6.3	ГЛОБАЛЬНЫЕ НАСТРОЙКИ.....	45
6.4	НАСТРОЙКИ ПРОФИЛЯ ПОЛЬЗОВАТЕЛЯ.....	45
7	ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ	47
7.1	УСТАНОВКА ПРИЛОЖЕНИЯ ИЗ ЧАРТА	47
7.2	УСТАНОВКА ПРИЛОЖЕНИЯ С ИСПОЛЬЗОВАНИЕМ KUBECTL	52
7.3	КОНФИГУРАЦИЯ СЕТЕВЫХ ПОЛИТИК CILLIUM	54

ГЛОССАРИЙ

Термин/сокращение	Определение
AlertManager	Компонент Prometheus, служит для запуска оповещений через Email, Slack или другие клиентские уведомления
Ansible	Продукт с открытым кодом, который автоматизирует подготовку облачных решений, управление конфигурацией и развертывание приложений
Cilium	Программное обеспечение с открытым исходным кодом для обеспечения, защиты и наблюдения за сетевым подключением между рабочими нагрузками контейнеров, созданное в облаке и основанное на технологии ядра eBPF
CIS Benchmarks	Набор рекомендаций по настройке широкого спектра ПО (серверное, операционные системы, облачное ПО, десктопное ПО)
Docker	Проект с открытым исходным кодом для автоматизации развертывания приложений в виде переносимых автономных контейнеров, выполняемых в облаке или локальной среде
Hubble	Передовая доступная система управления безопасностью разработки, созданная для достижения наилучших результатов в области безопасности разработки
Hashicorp Terraform	Средство IaC с открытым кодом для подготовки и управления облачной инфраструктуры
Grafana	Аналитическая платформа с открытым исходным кодом, которая позволяет опрашивать и визуализировать данные, отправлять предупреждения и разбираться в метриках независимо от того, где они хранятся
Loki	Набор компонентов для полноценной системы работы с логами
Longhorn	Распределенное блочное хранилище для K8s
Prometheus	Бесплатное программное приложение, используемое для мониторинга событий и оповещения
Yandex CSI	Технология позволяет динамически резервировать бакеты S3-совместимых хранилищ и монтировать их к подам кластера в виде постоянных томов Kubernetes (PersistentVolume)
CD	Непрерывное развертывание
CI	Непрерывная интеграция
CIS	Center for Internet Security – Центр Интернет Безопасности, некоммерческая организация, продвигающая передовые решения в области информационной безопасности
Configuration as Code (CaC), Infrastructure as Code (IaC)	Конфигурация в виде кода/ Инфраструктура в виде кода Определяет параметры конфигурации и инфраструктуры в удобочитаемом YAML-файле, который может храниться в виде исходного кода
cypher_key	Ключ шифрования
DNS	Система доменных имен
Docker	Программное обеспечение с открытым исходным кодом для автоматизации развертывания и управления приложениями в средах с поддержкой контейнеризации
Helm	Менеджер пакетов для Kubernetes
Hubble UI	Расширение Cilium, позволяющее схематично визуализировать потоки данных сетевого уровня
Ingress	Модуль управления внешними подключениями к кластеру

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Kubernetes, k8s	Платформа с открытым исходным кодом для управления кластером контейнерных приложений и сервисов
kubectl	Инструмент командной строки для управления кластером Kubernetes
Pod	Термин нотификации Kubernetes, объединяющий в себе перечень контейнеров приложения с общими ресурсами хранения и сетевыми ресурсами
RBAC	Метод регулирования доступа к компьютерным или сетевым ресурсам на основе ролей отдельных пользователей
S3	Simple Storage Service. Протокол передачи данных
S3-хранилище	Объектное хранилище. Позволяет хранить большие объемы данных в исходном формате без иерархии и разбивки на отдельные каталоги. Не имеет ограничений по масштабированию
ssh_key	Учетные данные для доступа по протоколу ssh
Stateless	Сервис без сохранения состояния
Terraform	Инструмент декларативного управления инфраструктурой
VMware	Технология виртуализации сервера
VShpere	Платформа виртуализации облачных вычислений от VMware
YAML	Формат представления данных, доступный для восприятия конечным пользователем
APM	Автоматизированное рабочее место - место работы оператора ПК
Кластер	Набор из нескольких серверов (узлов), на которых установлены компоненты среды контейнерной оркестрации
Контейнер	Экземпляр исполняемого программного обеспечения, объединяющий двоичный код приложения со связанными файлами конфигурации, библиотеками, зависимостями и средой выполнения
Контейнеризация	Технология изоляции процессов на основании пространства имен и групп пользователей операционной системы
Оркестрация контейнеров	Автоматизация и управление жизненным циклом контейнеров и услуг: автоматизация планирования, развертывания, масштабируемости, балансировки нагрузки, доступности и организации сетей контейнеров
ОС	Операционная система
Пространство имен	Множество, объединяющее модель, абстрактное хранилище или окружение, созданное для логической группировки уникальных идентификаторов (имен)
Платформа	«Платформа Боцман Клик»
ПО	Программное обеспечение
Репозиторий	Место хранения и поддержки структурированных данных
Секрет	Любая конфиденциальная информация. Например: логин, пароль, ключ и пр.
ЯндексОблако, YandexCloud, ЯО	Публичная облачная платформа

1. АННОТАЦИЯ

Настоящее руководство содержит информацию по установке и эксплуатации комплексной автоматизированной «Платформы Боцман Клик».

«Платформа Боцман Клик» — это совокупность программных средств, обеспечивающих комплексное управление кластеров kubernetes с набором готовых инструментов для развертывания, мониторинга, балансировки нагрузок, автомасштабирования, строгих политик безопасности и резервного копирования.

Платформа предназначена для создания и управления виртуальным частным облаком и может применяться во всех индустриях.

2. ОБЩАЯ ИНФОРМАЦИЯ

1.1 ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ ДЛЯ ЗАПУСКА УСТАНОВЩИКА

- На АРМ оператора должен быть установлен docker
- Сетевой доступ АРМ оператора до целевой площадки
- Подготовленное S3 хранилище для состояний кластера
- Подготовка Яндекс облака или vSphere
- Компоненты установщика

1.2 КОМПОНЕНТЫ УСТАНОВЩИКА

- Приложение Bootsman
С его помощью производятся все манипуляции с кластером
- Файл конфигурации config.yml
Хранение настроек для инсталлятора
- Ключ шифрования cipher_key
- Ключ шифрования файла-состояния в S3.
Должен быть сохранен, при утрате дальнейшее управление кластером с помощью приложения - невозможно.
Ключ должен быть 32х символьным
- Приватный ssh-ключ
Ключ для подключения к компонентам кластера
- Образ подготовленной виртуальной машины (Для VMware)
Используется как шаблон для создания всех узлов

1.3 ТРЕБОВАНИЯ К СРЕДЕ УСТАНОВКИ И ФУНКЦИОНИРОВАНИЯ ПЛАТФОРМЫ

Платформа работает под управлением следующих операционных систем:

Операционная система	Версия системы	Версия ядра
RedOS	MUROM (7.3.2)	5.15
Ubuntu	Focal/Jellyfish	5.4/5.15

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

В качестве аппаратного обеспечения Платформы используются:

- VMware vSphere версии гипервизора не ниже 7.0.3
- Yandex Cloud

Каждая инсталляция включает в себя создание следующих типов узлов:

- Bastion
Точка входа в кластер. Обеспечивает защиту и используется как прокси для ssh подключений. Обладает инструментами для управления кластера
- Master
Узлы обеспечивающие работоспособность кластера
- Worker
Узел обслуживающие полезную нагрузку

Для обеспечения отказоустойчивости минимальная конфигурация состоит из: Bastion, трех Master- и двух Worker-узлов

Количество узлов в кластере оркестрации контейнеров для приложений определяется требованиями к вычислительным ресурсам прикладного ПО с учетом факторов резервирования и механизмов обновления.

1.4 ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ К УЗЛАМ

Число узлов, а также их конфигурация определяется требованиями прикладного ПО с учетом фактором резервирования и обновления. Ниже представлены минимальные требования к узлам для их функционирования.

Master

Параметр	Минимальная конфигурация
CPU (vCPU)	4
RAM (GB)	8
Storage (GB)	32

Worker

Параметр	Минимальная конфигурация
CPU (vCPU)	4
RAM (GB)	8
Storage (GB)	60

Bastion

Параметр	Минимальная конфигурация
CPU (vCPU)	2
RAM (GB)	4
Storage (GB)	30

1.5 ТРЕБОВАНИЯ К ТОПОЛОГИИ СЕТИ

1. Узлы должны принадлежать одной сети
2. Узлы должны иметь доступ к ресурсам в сети интернет, указанным далее по тексту

1.6 ТРЕБОВАНИЯ ДОСТУПНОСТИ РЕСУРСОВ

В процессе установки потребуются дополнительные компоненты, доступность которых определяется в зависимости от выбора Операционной системы:

Ubuntu 20.04

Ресурс	Описание
http://archive.ubuntu.com/ubuntu	Основной репозиторий для ubuntu 20.04
http://security.ubuntu.com/ubuntu	Репозиторий с обновлениями безопасности
https://download.docker.com	Репозиторий с docker
https://get.helm.sh	Инструмент Helm
https://storage.googleapis.com	Инструмент kubectl
https://docker.io	Репозиторий Docker контейнеров
https://github.com/rancher/rke/releases/download	Инструмент rke
https://git.stsoft.team	git&helm репозиторий stsoft
https://charts.rancher.io/	Helm репозиторий rancher
registry.terraform.io	Репозиторий операторов terraform

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Ubuntu 22.04

Ресурс	Описание
http://mirror.yandex.ru/ubuntu	Основной репозиторий для ubuntu 22.04
http://security.ubuntu.com/ubuntu	Репозиторий с обновлениями безопасности
https://download.docker.com	Репозиторий с docker
https://get.helm.sh	Инструмент Helm
https://storage.googleapis.com	Инструмент kubectl
https://docker.io	Репозиторий Docker контейнеров
https://github.com/rancher/rke/releases/download	Инструмент rke
https://git.stsoft.team	git&helm репозиторий stsoft
https://charts.rancher.io/	Helm репозиторий rancher
registry.terraform.io	Репозиторий операторов terraform

RedOS

Ресурс	Описание
https://repo1.red-soft.ru	Репозиторий RedOS
https://get.helm.sh	Инструмент Helm
https://storage.googleapis.com	Инструмент kubectl
https://docker.io	Репозиторий Docker контейнеров
https://github.com/rancher/rke/releases/download	Инструмент rke
https://git.stsoft.team	git&helm репозиторий stsoft
https://charts.rancher.io/	Helm репозиторий rancher
registry.terraform.io	Репозиторий операторов terraform

1.7 УСТАНОВЛИВАЕМОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ НА УЗЛЫ КЛАСТЕРА

В процессе установки на узлы кластера устанавливается следующее базовое ПО:

Ubuntu	Redos
cloud-init	cloud-init
openssh-server	openssh-server
nfs-common	nfs-utils
python3-pip	python3-pip
lvm2	lvm2
open-iscsi	iscsi-initiator-utils
net-tools	net-tools
htop	htop
vim	vim
tmux	tmux
ncdu	ncdu
tcpdump	tcpdump
strace	strace
tree	tree
iftop	iftop
nano	nano
traceroute	traceroute
nmon	nmon
mtr	mtr
iostat	iostat
mc	mc
snappd	snappd
wget	wget
curl	curl
lsof	lsof

Ubuntu	Redos
rsync	rsync
python3-setuptools	python3-setuptools
zsh	zsh
apt-transport-https	
ca-certificate	ca-certificates
software-properties-common	
virtualenv	python3-virtualenv

1.8 ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УСТАНОВЛИВАЕМОЕ НА УЗЛЫ ПО РОЛЯМ

Master- и Worker-узлы

Ubuntu	RedOS
docker-ce 5:20.10.23~3-0~ubuntu-bionic	docker-ce 3:20.10.10-1.el7
docker-ce-cli 5:20.10.23~3-0~ubuntu-bionic	docker-ce-cli 1:20.10.10-1.el7
containerd.io 1.6.16-1	containerd.io 1.5.8-2.el7

Bastion

Ubuntu/RedOS
Helm
kubectl

2 УСТАНОВКА ПЛАТФОРМЫ

2.1 ПОДГОТОВКА ПЛОЩАДОК ДЛЯ УСТАНОВКИ ПЛАТФОРМЫ*

vSphere	Yandex Cloud
Загрузить базовый образ (.iso) ubuntu или redos в кластер	Подготовить Cloud и Fodler для инсталляции
Создать ResourcePool для инсталляции	Создать сервис-аккаунт с правами editor, и создать авторизационный ключ
	Получить свой OAuth токен

*В контексте настоящего руководства предварительная настройка vSphere и Yandex Cloud не рассматривается.

2.2 КОНФИГУРАЦИОННЫЙ ФАЙЛ

```
# Тип провайдера, yc - ЯО, vsp - vSphere
cloudId: yc
# Абсолютный путь до приватного ssh-ключа
privateKeyPath: /id/rsa/path
# Настройки для подключения к s3 хранилищу
s3:
  # Адрес к хранилищу
  endpoint: localhost.localdomain
  # Включить SSL
  secure: true
  # Имя ключа для подключения
  accessKeyId: 2345defgh
  # Секрет для подключения
  secretAccessKey: 12345678
  # Имя бакета
  bucketName: "bucket-name"
  # Директория для хранения state-файла внутри s3-бакета. Должен быть
  # уникальным для каждого кластера.
  prefixFolder: "bootsman-1"
cloudConfig:
  # Путь до kubecofig
  k8sKubeConfig: "~/kube/config"
  # Базовое доменное имя
  domainSuffix: soft-s.tech
  # Доменное имя для Rancher. Без указания протокола (http/https)
  rancherWebHostname: bootsman-test.soft-s.tech
  # Системный пользователь
  rkeUser: bootsman
  # Registry proxy
  registryProxy: "https://registry.bootsman.local"
infrastructure:
  # Используемая операционная система. "redos" или "ubuntu"
  os: ubuntu
  # Название сети, в которой будет работать кластер. В ЯО - сеть будет
  # создана, в vSphere сеть должна существовать
  resVpcNetworkName:
```

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

```
# Настройка Узлов
instances:
  # Тип хранилища. Только для ЯО
  diskType: network-ssd
  bastion:
    # Число vCPU
    cores: 2
    # Объем оперативной памяти, GB
    memory: 4
    # Объем дискового пространства, GB
    diskSize: 30
  master:
    # Число Master-узлов. Должно быть 3 и более, причем должно быть
    нечетным
    count: 3
    # Число vCPU
    cores: 4
    # Объем оперативной памяти, GB
    memory: 8
    # Объем дискового пространства, GB
    diskSize: 32
  worker:
    # Число Worker-узлов. Возможный минимум - 1. Рекомендуемый минимум - 2,
    для обеспечения отказоустойчивости.
    count: 2
    # Число vCPU
    cores: 4
    # Объем оперативной памяти, GB
    memory: 8
    # Объем дискового пространства, GB
    diskSize: 60
  # Дополнительный узел для ЯО
  frontend:
    # Число vCPU
    cores: 2
    # Объем оперативной памяти, GB
    memory: 6
    # Объем дискового пространства, GB
    diskSize: 40

# Настройки для подключения к ЯО
yandex:
  # Авторизационный ключ ЯО (Создать в web-интерфейсе яндекс облака и скачать
  в виде json)
  keysData: '{}'
  # id folder в ЯО
  folderId:
  # Зона доступности ЯО. Варианты: ru-central1-a, ru-central1-b, ru-central1-
  c
  providerZone:
  # Токен для авторизации в ЯО
  oAuthToken:
  # id целевого клауда в ЯО
  cloudId:
  # Идентификатор образа
  img:

# Настройки для подключения в vSphere
vsphere:
  # Имя пользователя vSphere
  login:
```

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

```
# Пароль пользователя
password:
# Web-адрес vSphere
baseUrl: http://10.0.1.10
# Имя датацентра
datacenter: "vSAN Datacenter"
# Используемый Resource Pool. Он должен быть преднастроен
resourcePool: "bootsman_ts"
```

2.3 ЗАПУСК И ПРОЦЕСС УСТАНОВКИ

Запуск процесса установки осуществляется посредством интерфейса командной строки.

Все операции, связанные с установкой и базовым обслуживанием, выполняются на APM Оператора, в директории расположения файла bootsman.

Перед запуском следует отредактировать config.yml

Перед запуском следует отредактировать config.yml
Подготовить ключ шифрования вручную или командой:

```
echo $RANDOM | md5sum | head -c 32 > cypher_key
```

Запустить инсталляцию

```
./bootsman up
```

Вывод успешного выполнения инсталляции

```
# Инициализация провайдеров terraform
2023/02/16 14:06:50 Initialization infrastructure
# Генерация промежуточного файла конфигурации hosts.cfg
2023/02/16 14:06:59 Generate hosts
# Проверка terraform-инструкций
2023/02/16 14:06:59 Initialization infrastructure.
# Подготовка конфигурации
2023/02/16 14:07:00 Initialization infrastructure..
# Применение terraform-инструкций
2023/02/16 14:07:00 Building infrastructure
# Запуск проверки состояния созданных виртуальных машин
2023/02/16 14:07:44 Building infrastructure.
# Успешное создание виртуальных машин
2023/02/16 14:08:26 Successfully created bastion at 158.160.8.2
2023/02/16 14:08:28 Successfully created master-0 at 192.168.142.20
2023/02/16 14:08:31 Successfully created master-1 at 192.168.142.14
2023/02/16 14:08:33 Successfully created master-2 at 192.168.142.31
2023/02/16 14:08:36 Successfully created worker-0 at 192.168.142.5
2023/02/16 14:08:42 Successfully created worker-1 at 192.168.142.38
2023/02/16 14:08:43 Successfully created worker-2 at 192.168.142.30
# Установка требуемых пакетов на Master и Worker узлы
2023/02/16 14:08:43 Configuring servers
2023/02/16 14:11:21 Configuring servers.
# Установка docker на Master и Worker узлы
2023/02/16 14:11:21 Configuring docker
# Установка пакетов и инструментов на bastion
2023/02/16 14:13:55 Configuring bastion
# Загрузка контейнеров для кластера
2023/02/16 14:16:24 Preloading components
```

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

```
# Генерация cluster.yml для приложения rke
2023/02/16 14:25:36 Generating k8s cluster config
# Создание rke кластера
2023/02/16 14:25:57 Building k8s cluster
# Генерация kubeconfig и перенос на bastion
2023/02/16 14:36:16 Generating kubeconfig
# Установка CNI плагин cilium
2023/02/16 14:36:33 Install k8s network
# Установка драйвера хранилища
2023/02/16 14:39:31 Install storage driver
# Установка Rancher
2023/02/16 14:41:13 Install k8s components
# Установка мониторинга
2023/02/16 14:43:51 Install monitoring system
# Установка системы логирования
2023/02/16 14:47:15 Install logging system
# Установка и активация плагинов
2023/02/16 14:49:05 Install UI extention
# Сообщение об окончании установки
2023/02/16 14:52:18 Bootsman finished your infrastructure.
# Сообщение с сгенерированным паролем для входа
2023/02/16 14:52:18 Your bootstrap password is
MYss8YxsxgXZNIItjRZ661VayzAFDvvgb
```

Платформа готова к работе.

Расширенный журнал установки "bootsman.log" создается и дополняется на ПК Оператора, в той же директории, где расположен установочный файл.

3 НАЧАЛО РАБОТЫ И ОБСЛУЖИВАНИЕ КЛАСТЕРА

Стандартная модель работы платформы предусматривает взаимодействие посредством интерфейса командной строки и веб-интерфейса. В текущем разделе рассмотрен пример взаимодействия через командную строку:

- Добавление узлов в кластер
- Удаление узлов из кластера
- Удаление кластера

3.1 ДОБАВЛЕНИЕ УЗЛОВ В КЛАСТЕР

Перед запуском в той же директории должен быть файлы для целевого кластера:

1. config.yml
2. key
3. cypher_key

Запустить приложение с флагом add

Добавить два мастера (добавляемое число мастеров должно быть четным):

```
./bootsman add --master 2
```

Добавить два воркера (добавляется любое количество)

```
./bootsman add --worker 2
```

Вывод успешного выполнения добавления узла

```
# Инициализация провайдеров terraform:
```

```
2023/02/28 21:44:42 Initialization infrastructure
```


ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

```
# Генерация промежуточного файла конфигурации hosts.cfg
2023/02/28 21:44:44 Generate hosts
# Получение списка узлов
2023/02/28 21:44:44 Get list
# Добавление информации о новых нодах
2023/02/28 21:44:44 Add node part
2023/02/28 21:45:47 Add node part
# Применение terraform-инструкций:
2023/02/28 21:46:57 Initialization infrastructure
# Генерация промежуточного файла конфигурации hosts.cfg
2023/02/28 21:47:00 Generate hosts
# Запуск проверки состояния созданных виртуальных машин
2023/02/28 21:47:01 Building new node(s) infrastructure.
# Сообщении об успешном запуске новых узлов
2023/02/28 21:47:01 Successfully created bastion-0 at 10.0.104.234
2023/02/28 21:47:02 Successfully created master-0 at 10.0.104.237
2023/02/28 21:47:03 Successfully created master-1 at 10.0.104.232
2023/02/28 21:47:03 Successfully created master-2 at 10.0.104.236
2023/02/28 21:47:04 Successfully created master-3 at 10.0.104.220
2023/02/28 21:47:04 Successfully created master-4 at 10.0.104.219
2023/02/28 21:47:05 Successfully created worker-1 at 10.0.104.231
2023/02/28 21:47:06 Successfully created worker-2 at 10.0.104.233
2023/02/28 21:47:06 Successfully created worker-4 at 10.0.104.221
# Установка требуемых пакетов на новые узлы
2023/02/28 21:47:06 Base init node
# Установка требуемых пакетов
2023/02/28 21:49:05 Init node
# Загрузка контейнеров для новый узлов
2023/02/28 21:50:35 Preloading components
# Инсталляция rke в исполняемую среду
2023/02/28 21:51:49 Generating cluster binary
# Обновление кластера, обновление kubeconfig
2023/02/28 21:51:57 Update node
# Успешное добавление узлов
2023/02/28 21:56:58 Added 2 master(s) successfull
```

3.2 УДАЛЕНИЕ УЗЛОВ

Перед запуском в той же директории должны быть в наличии файлы для целевого кластера:

1. config.yml
2. key
3. cypher_key

Есть несколько правил удаления нод, нарушение предусмотрено приложением и сообщит об этом перед внесением изменений:

1. Мастер-ноды удаляются только четным числом
2. Рабочих мастер-нод(Статус - Ready) не должно остаться меньше 3х после удаления

Удаление нод из кластера

```
./bootsman remove --ip <ip1>,<ip2>
```

```
# Инициализация провайдеров terraform
2023/02/28 21:52:22 Initialization infrastructure
# Генерация промежуточного файла конфигурации hosts.cfg
2023/02/28 21:52:24 Generate hosts
# Проверка статуса Мастер-нод
2023/02/28 21:52:25 Check masters ready
# Инсталляция rke в исполняемую среду
```

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

```
2023/02/28 21:52:33 Generating cluster binary
# Обновление кластера
2023/02/28 21:52:41 Update node
# Получение списка узлов
2023/02/28 21:55:56 Get list
# Удаление нод
2023/02/28 21:55:56 Remove node part
2023/02/28 21:56:17 Remove node part
# Получение списка узлов
2023/02/28 21:56:36 Get list
#
2023/02/28 21:56:36 Initialization infrastructure
# Генерация файла конфигурации hosts.cfg
2023/02/28 21:56:38 Generate hosts
# Успешное удаление
2023/02/28 21:56:39 remove done
```

3.3 УДАЛЕНИЕ КЛАСТЕРА

```
#Перед запуском в той же директории должен быть файлы для целевого кластера:
#1. config.yml
#2. key
#3. cypher_key
#Команда для удаления кластера:
./bootsman destroy
```

Результатом выполнения команды является полное удаление всех компонентов платформы.

Дальнейшая работа с платформой осуществляется через вэб-интерфейс.

4 ИНТЕРФЕЙС УПРАВЛЕНИЯ

4.1 ПОДГОТОВКА К РАБОТЕ

Как было указано в п.2.3, платформа готова к работе после появления строк:

```
2023/02/16 14:52:18 Your bootstrap password is
MYss8YxsxgXZNIItjRZ661VayzAFDvgb
```

Доменное имя, указанное в конфигурационном файле, также является адресом вэб-интерфейса панели управления.

```
# Базовое доменное имя
domainSuffix: soft-s.tech
# Доменное имя для Rancher. Без указание протокола (http/https)
rancherWebHostname: bootsman-test.soft-s.tech
```

Для работы внутри изолированной сети необходимо в файлах hosts на APM операторов добавить запись вида*:

```
10.0.104.230 bootsman-test.soft-s.tech cilium-editor.soft-s.tech
```

Где [10.0.104.230] - ip адрес любой worker-ноды

[bootsman-test.soft-s.tech cilium-editor.soft-s.tech] поддомены основного веб-интерфейса и редактора сетевых политик.

*Дополнительный вариант использования ДНС в контексте настоящей инструкции не рассматривается.

4.2 ПЕРВЫЙ ВХОД И НАЧАЛО РАБОТЫ

При первом переходе по адресу в браузере система предложит ввести только bootstrap пароль (см.п.2.3).

Второй и последующие входы организованы либо посредством локальных пользователей, либо посредством осуществления интеграции с существующим каталогом пользователей и выдачей необходимых прав.

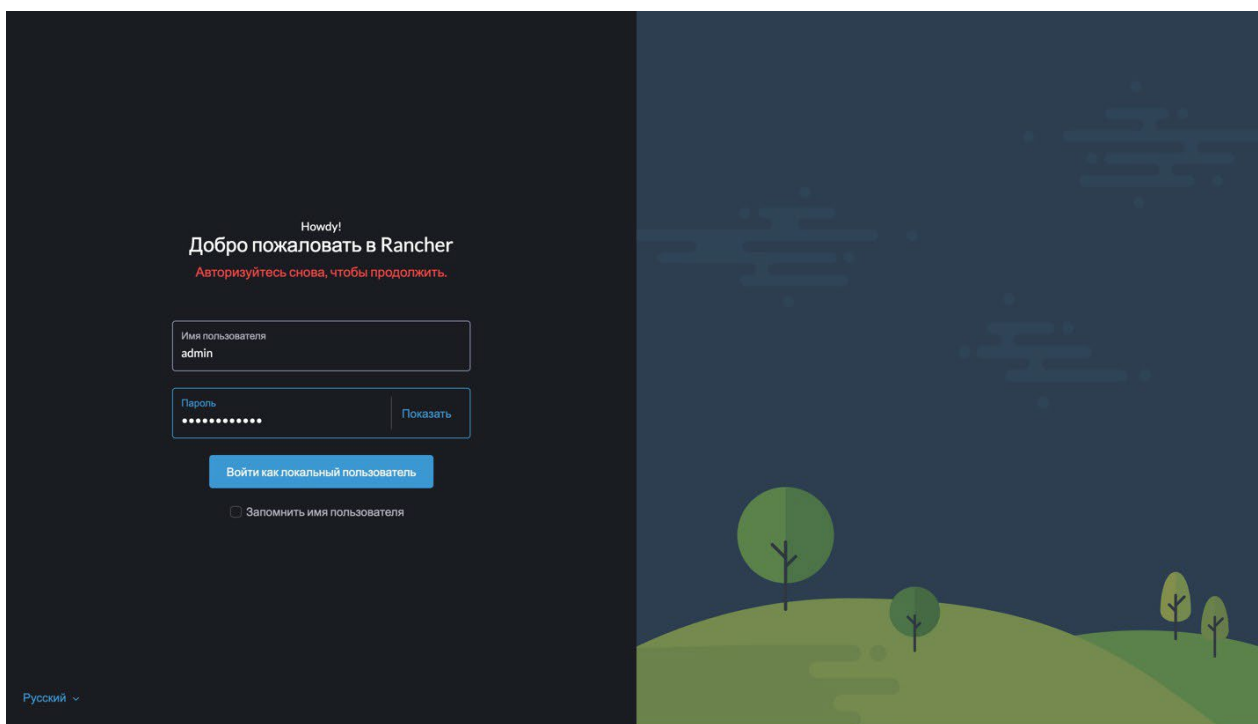


Рис. 1. Окно авторизации.

На странице, следующей после страницы авторизации, отображается основная информация о кластерах.

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

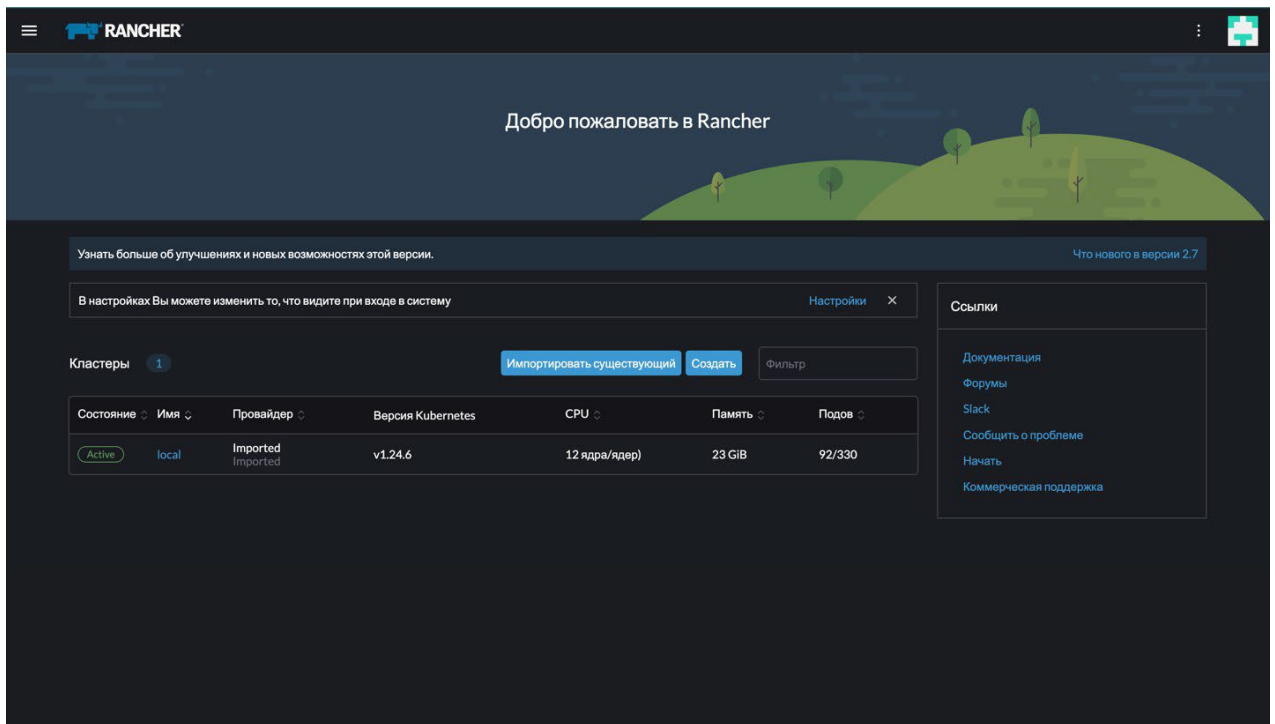


Рис. 2 Обзорная страница панели управления.

Элементы управления главной страницы размещены в левом и правом верхнем углах.

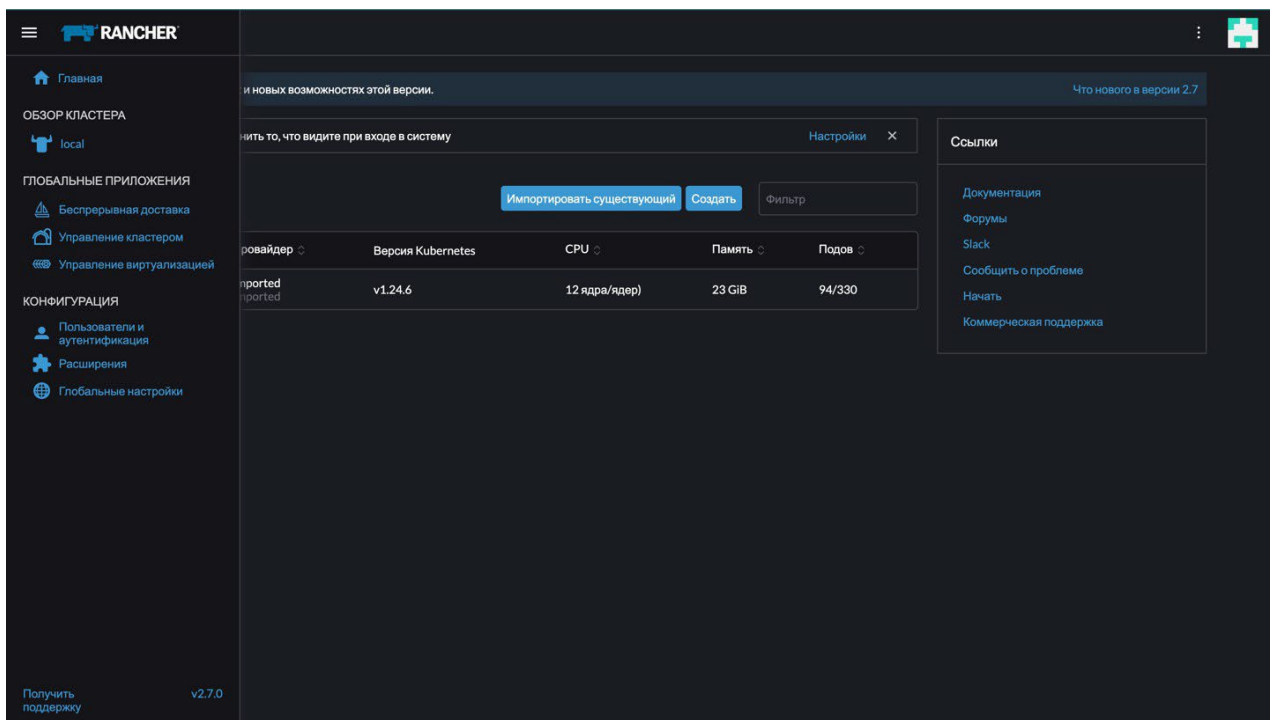


Рис. 3 Слева сверху меню управления подконтрольной инфраструктурой.

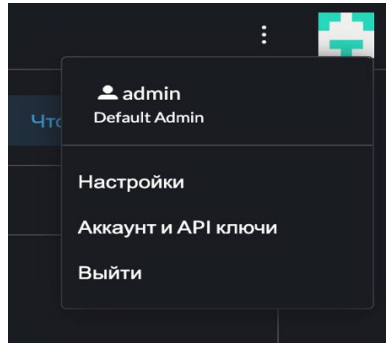


Рис. 4 Меню управления системой.

4.3 КЛАСТЕР

Основные элементы управления кластером отражены в меню.

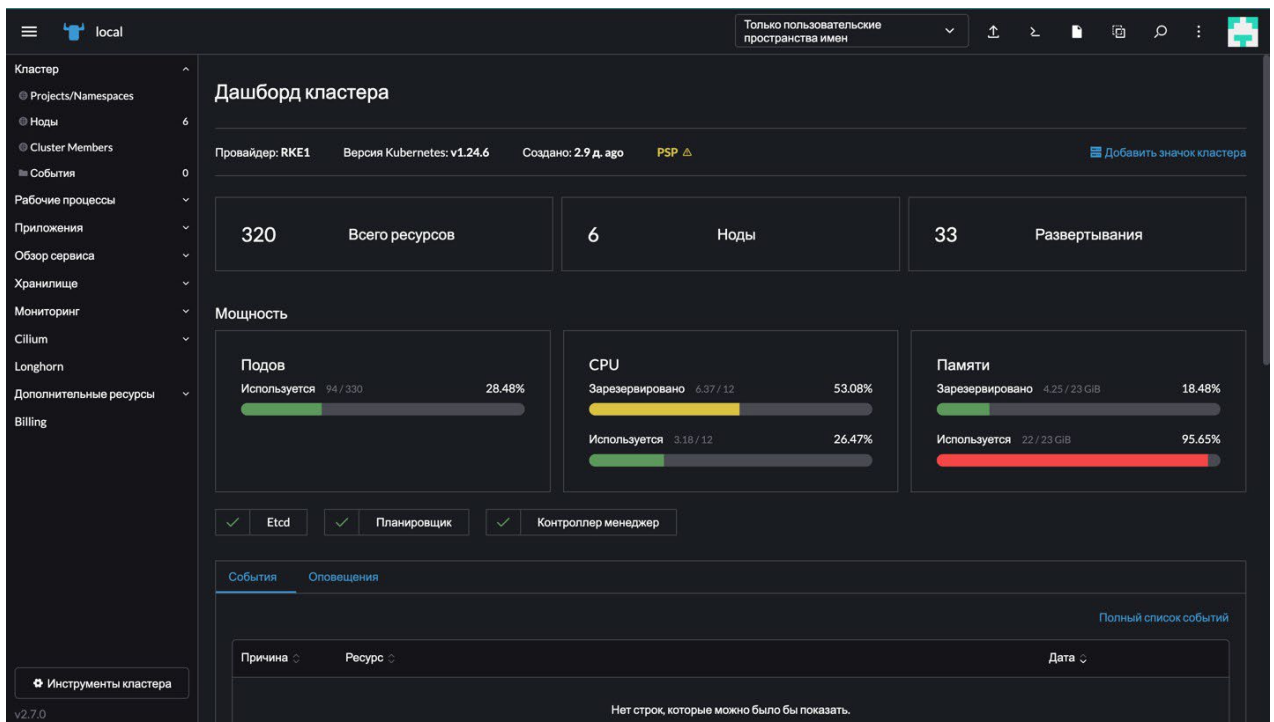


Рис. 5 Обзор пункта меню «Кластер»

Раздел «Project/Namespaces» позволяет управлять существующими проектами и пространствами имен и создавать новые по взаимодействию с одноименными кнопками.

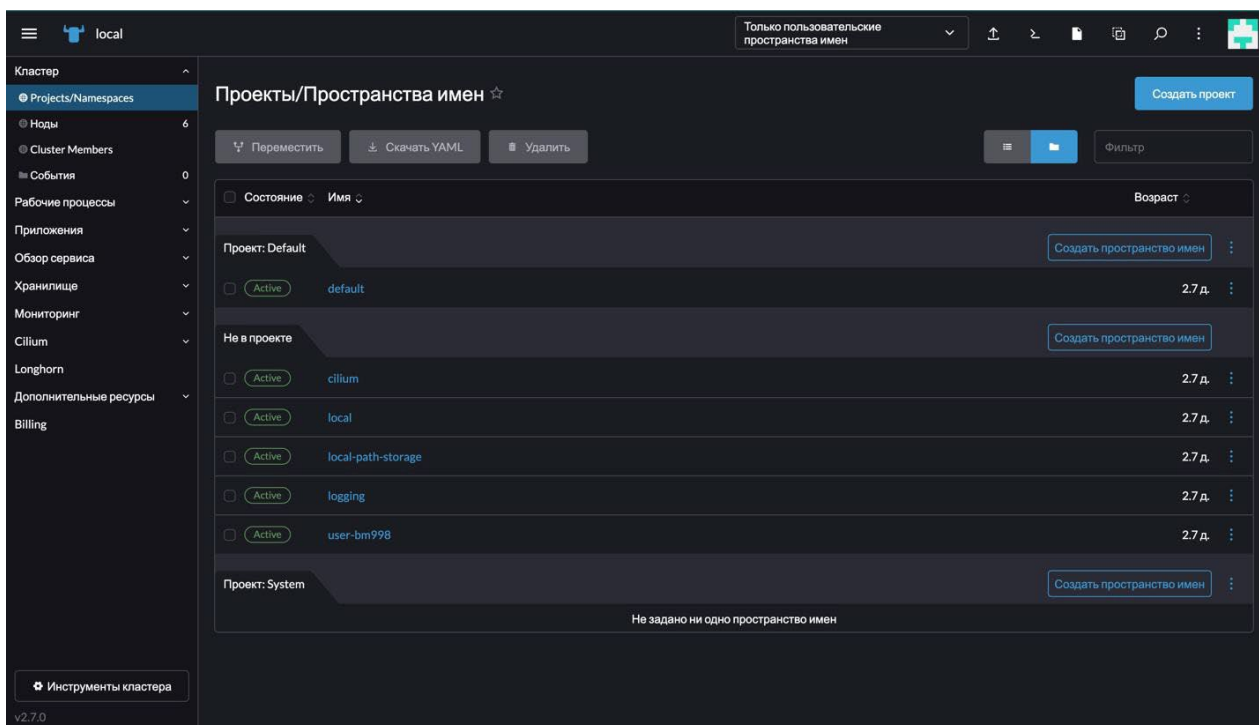


Рис. 6 Обзор пункта меню «Project/Namespaces»

Элементы управления под заголовками.

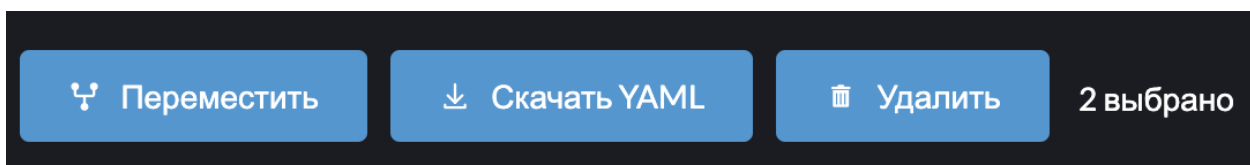


Рис. 7 Основные элементы взаимодействия

Позволяют управлять выбранными элементами через чекбокс, скачать IaC-манифесты элементов.

Элементы управления указанные выше унифицированы и механика их работы одинакова во всех случаях их использования, вне зависимости от страницы, на которой они используются.

Создание проекта/пространства имен предусматривает определение политики доступов для участников, выделение квот ресурсов в рамках проекта, лимитов по умолчанию для контейнера, лейблов и примечаний для удобства мониторинга и управления.

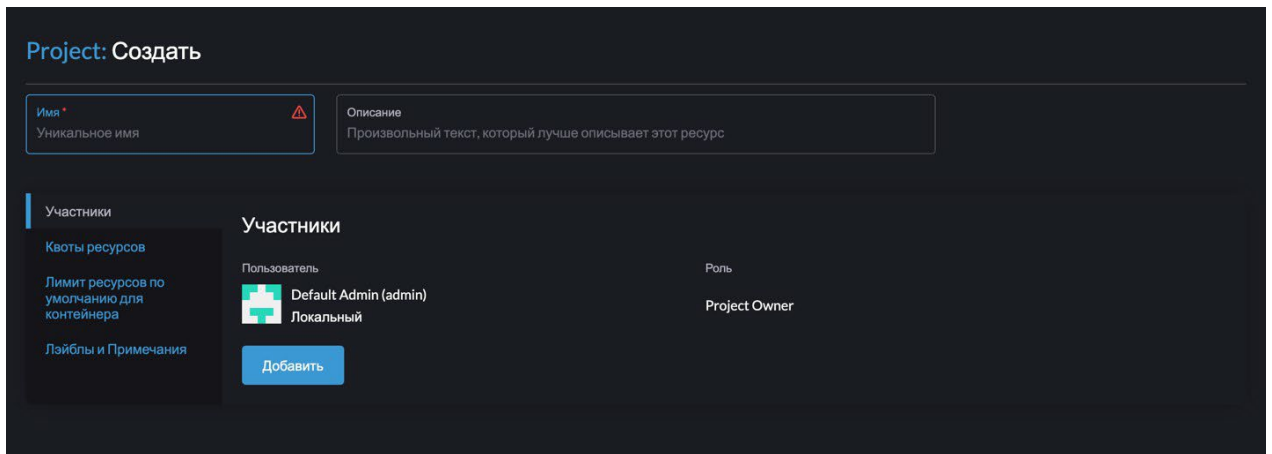


Рис. 8 Создание проекта.

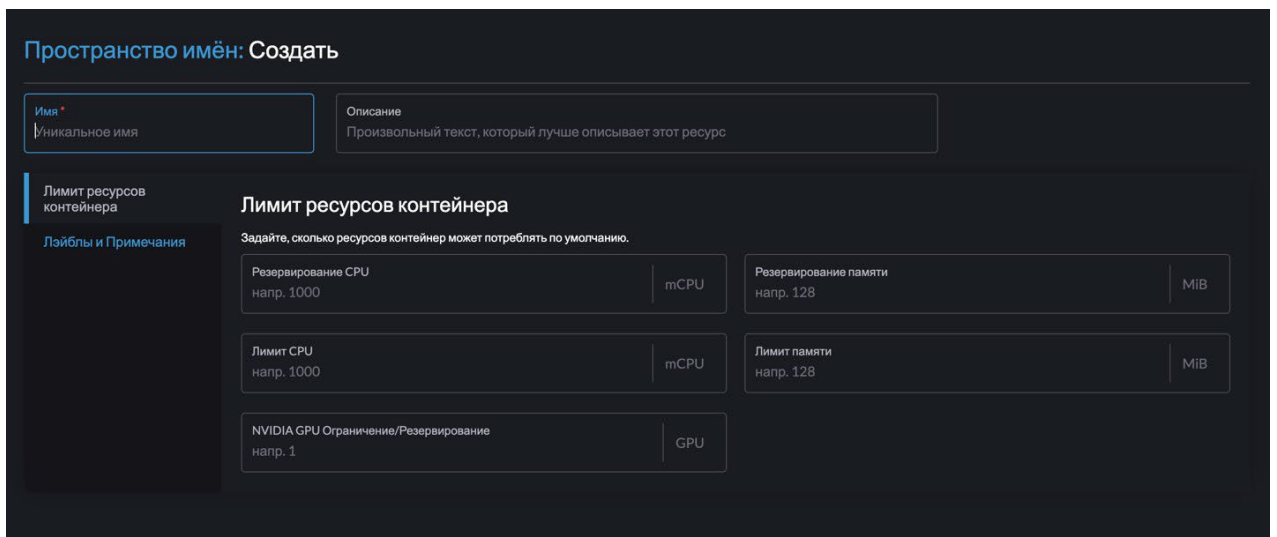


Рис 9. Создание пространства имен.

Подпункт «Ноды» позволяет управлять базовыми ресурсами Kubernetes и отслеживать их текущее состояние.

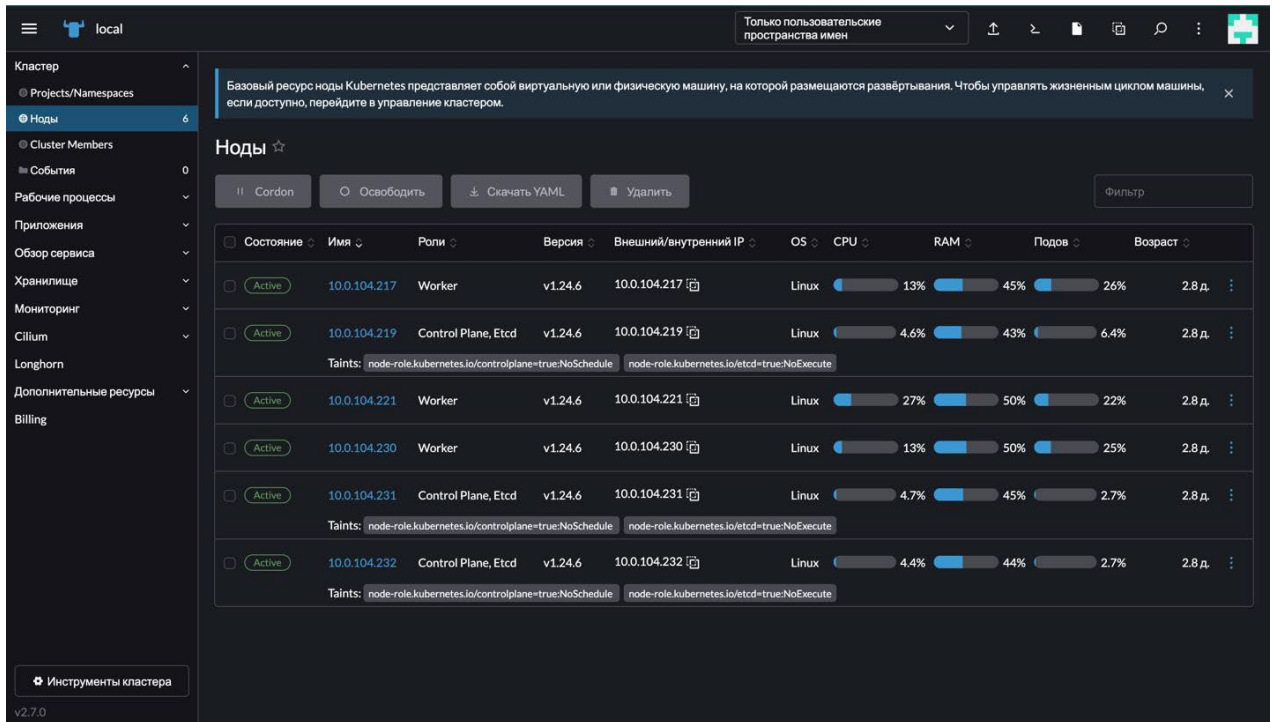


Рис. 10 Окно управления «нодами».

Подпункт «Cluster members» позволяет просматривать участников, допущенных к управлению кластером и делегировать существующим пользователям полномочия управления текущим кластером.

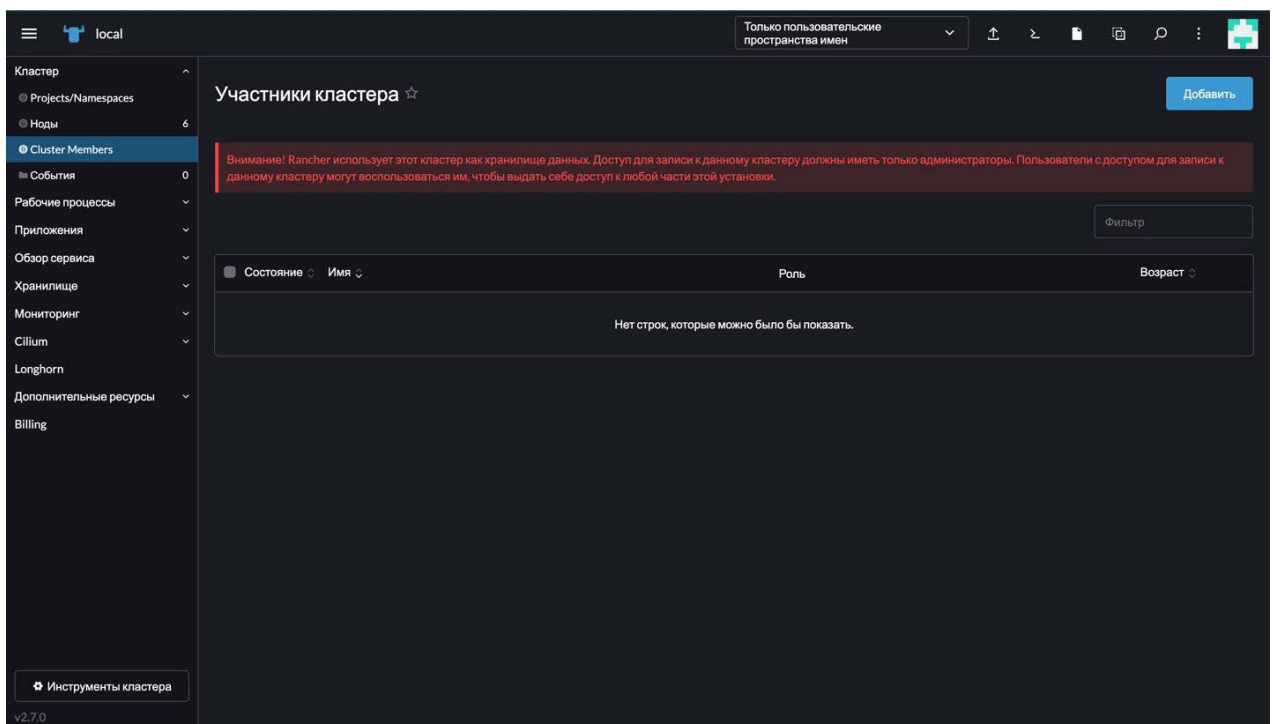


Рис. 11 Подпункт меню «Cluster members».

В подпункте меню «События» отражены важные события жизненного цикла кластера, а также, существует возможность добавить требуемые условия для отображения дополнительных событий.

4.4 РАБОЧИЕ ПРОЦЕССЫ

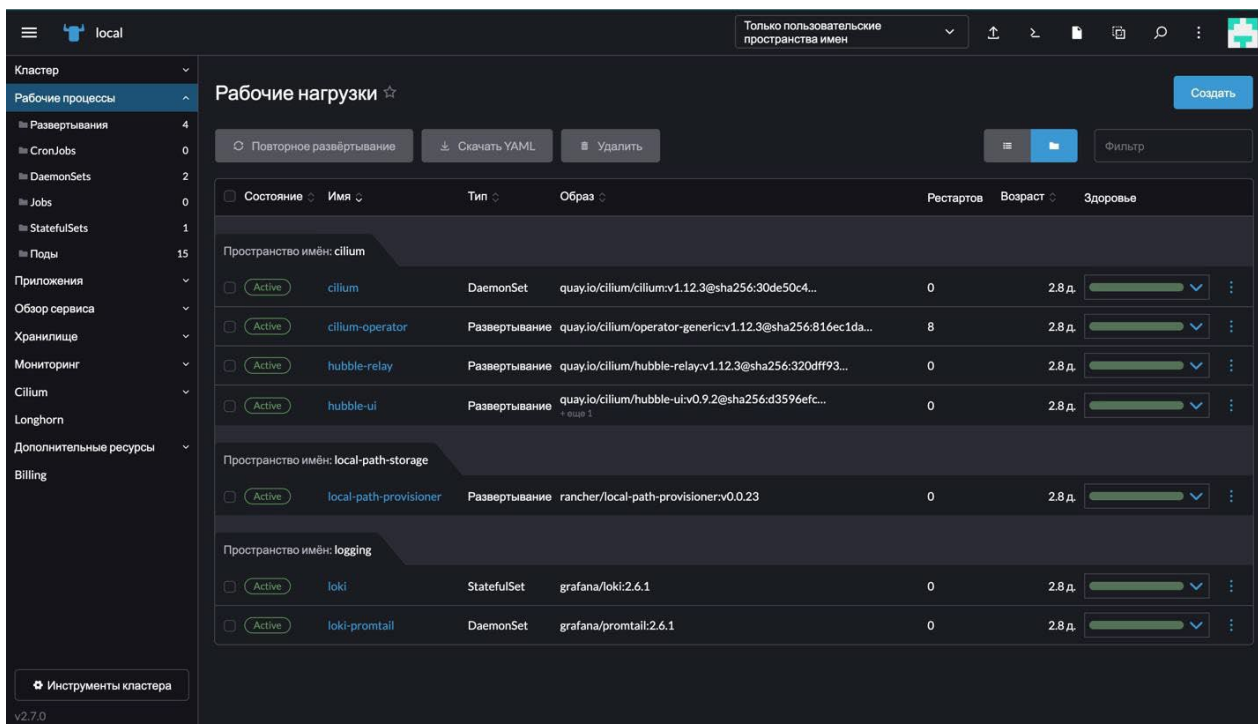


Рис. 12 Пункт меню «Окно рабочих процессов» включает себя подпункты меню по типу состояния.

По кнопке «Создать» доступно меню создания контейнеров с указанием обязательных и опциональных полей.

Дополнительно, создание контейнеров поддерживается через IaC-нотификации.

При создании допустимо указание квот ресурсов, параметров условий жизнеспособности приложения, указание политик безопасности, томов хранилищ.

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

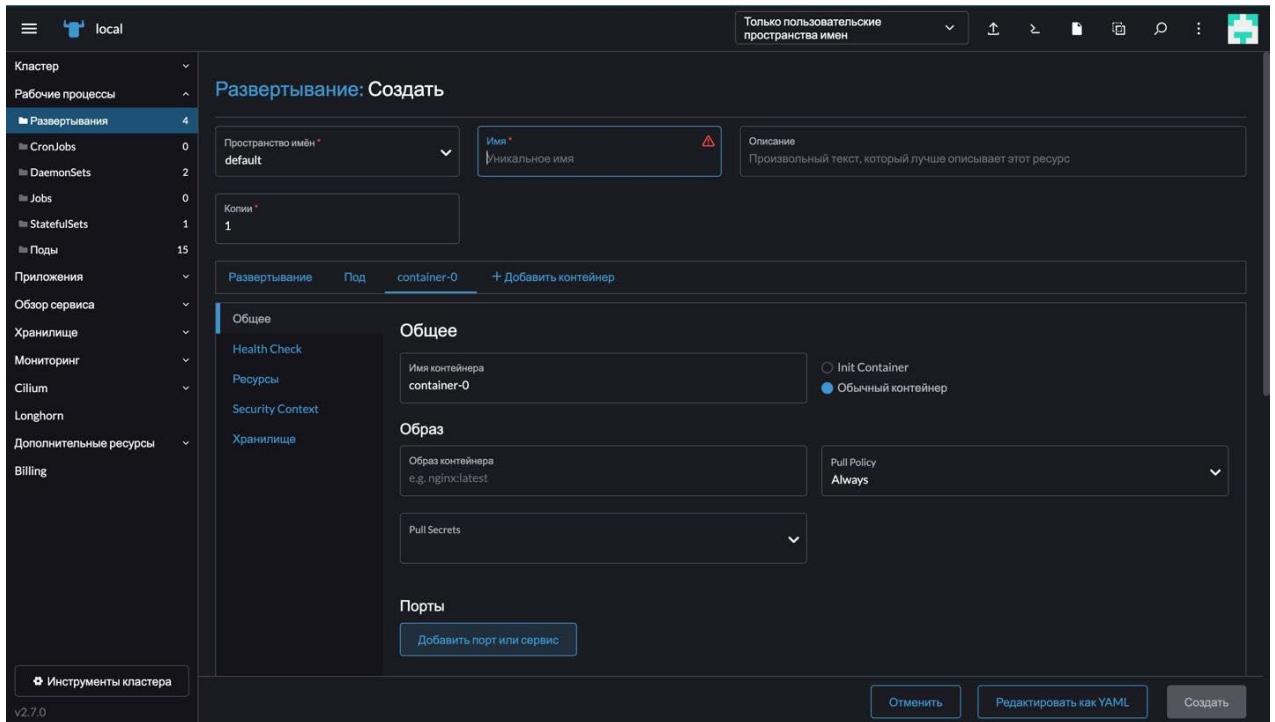


Рис. 13 Меню создания контейнера.

При переходе по ссылкам-именам отображается детальная информация о количестве подов, наименовании, полные адреса образов контейнеров, ip-адреса и прочая сопроводительная информация.

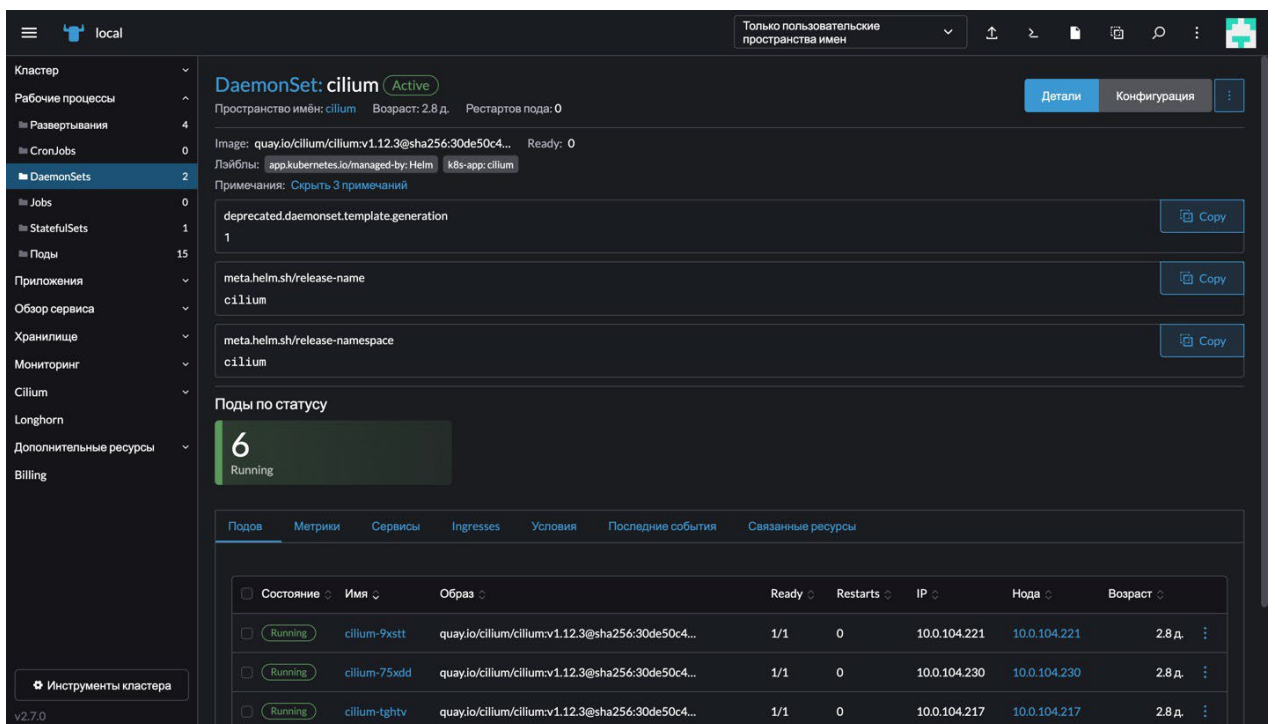


Рис. 14 Подпункт меню «Детали».

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

В подпункте «Конфигурация» для отображения приведены ключевые метрики параметров запущенных подов/контейнеров.

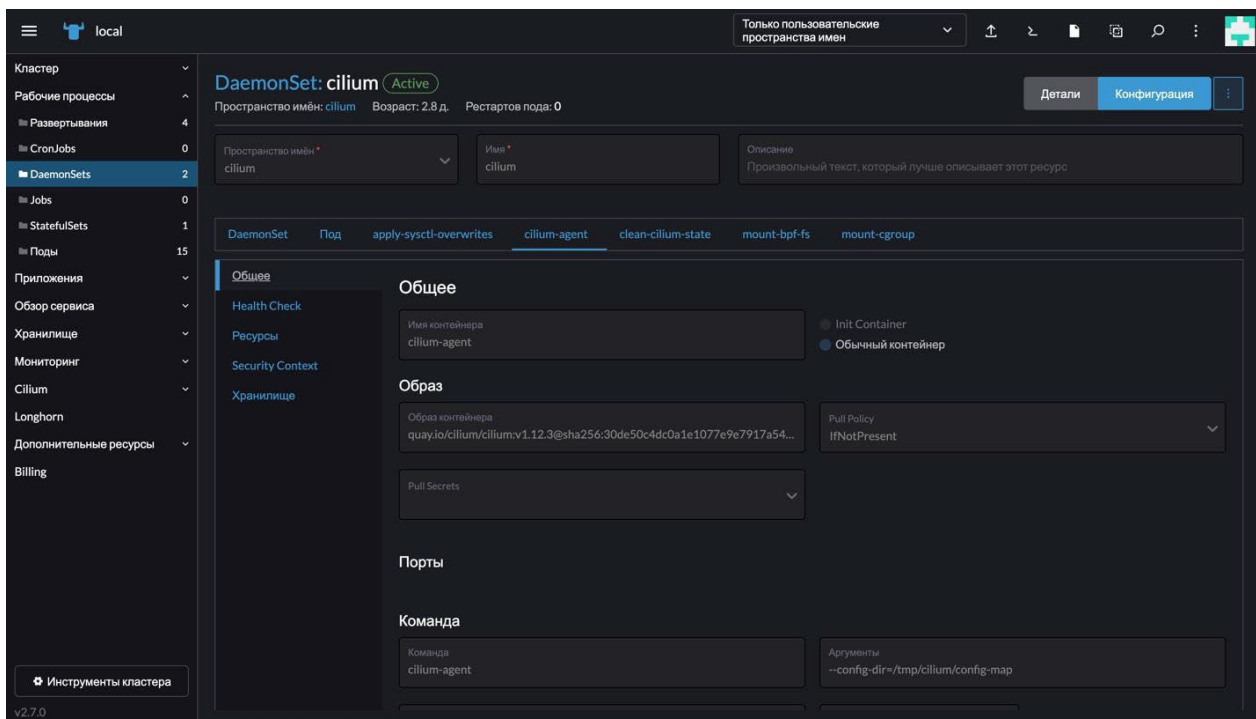


Рис. 15 Подпункт меню «Конфигурация».

Контекстное меню блока «Детали/Конфигурация» на страницах при переходе по ссылкам-именам позволит оперативно управлять подами.

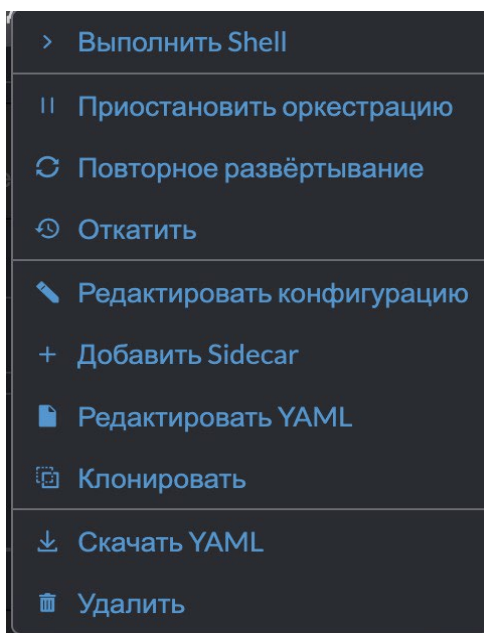


Рис. 16 Контекстное меню подпункта «Детали/Конфигурация»

4.5 ПРИЛОЖЕНИЯ

Пункт меню «Приложения» содержит в себе подпункты «Charts», «Установленные приложения», «Репозитории» и «Последние действия».

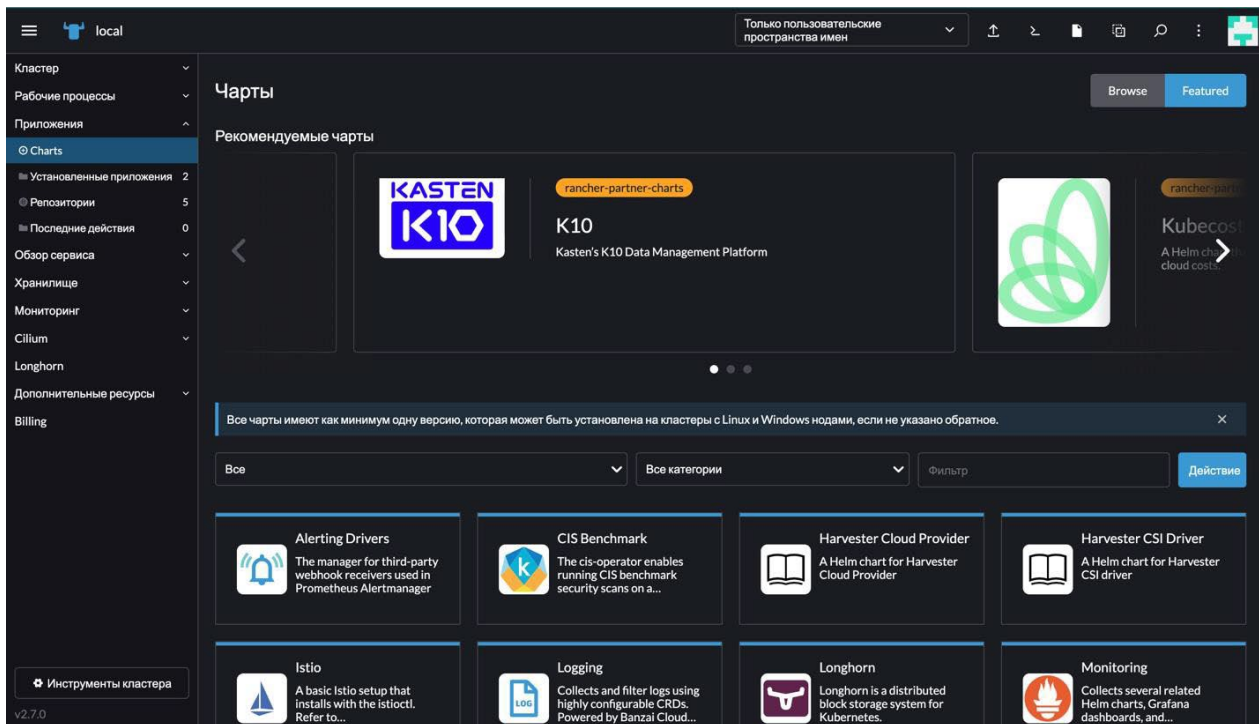


Рис. 17 Окно подпункта «Charts»

В подпункте «Charts» отображаются готовые к установке предварительно сконфигурированные и нативные приложения, доступные для быстрой установки в кластер.

Установленные приложения - функционирующие на текущий момент приложения в кластере

Репозитории - каталоги приложений, к которым обращается кластер.

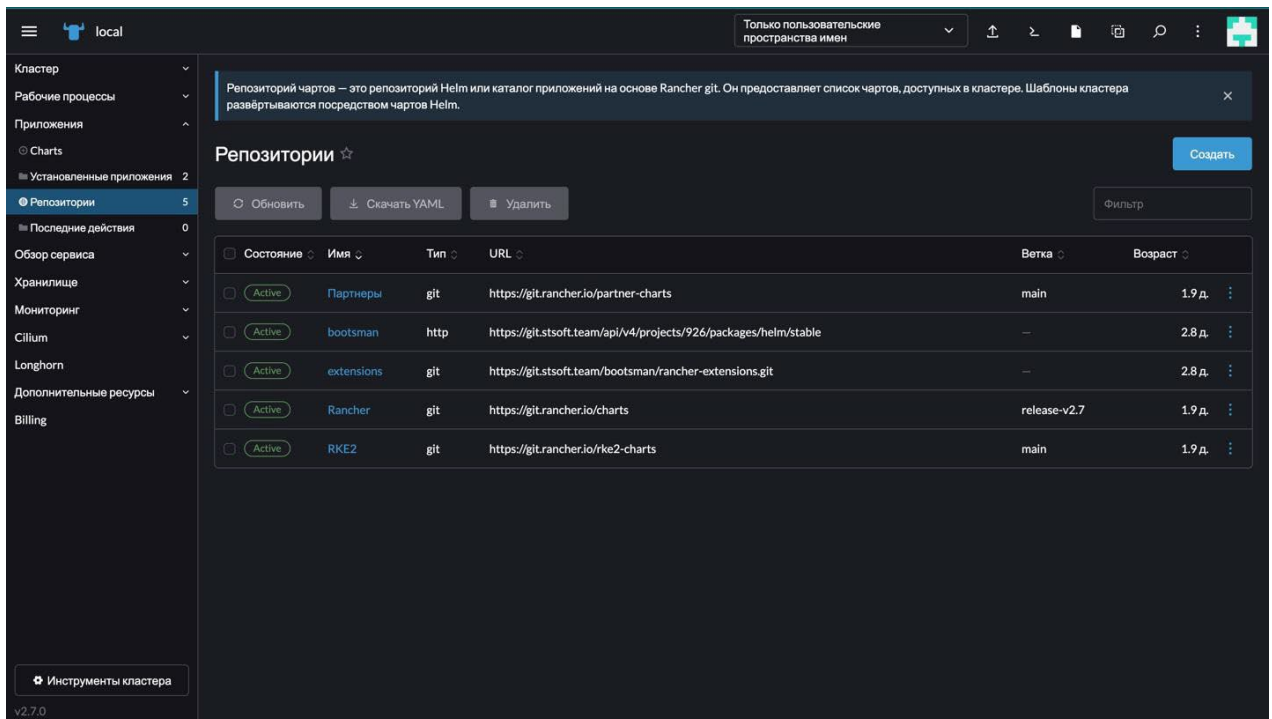


Рис. 18 Подпункт меню «Репозитории»

Предустановленный перечень репозитория не ограничен и может дополняться.

Подпункт меню «Последние действия» - содержит в себе историю последних операции блока «Приложения».

4.6 ОБЗОР СЕРВИСА

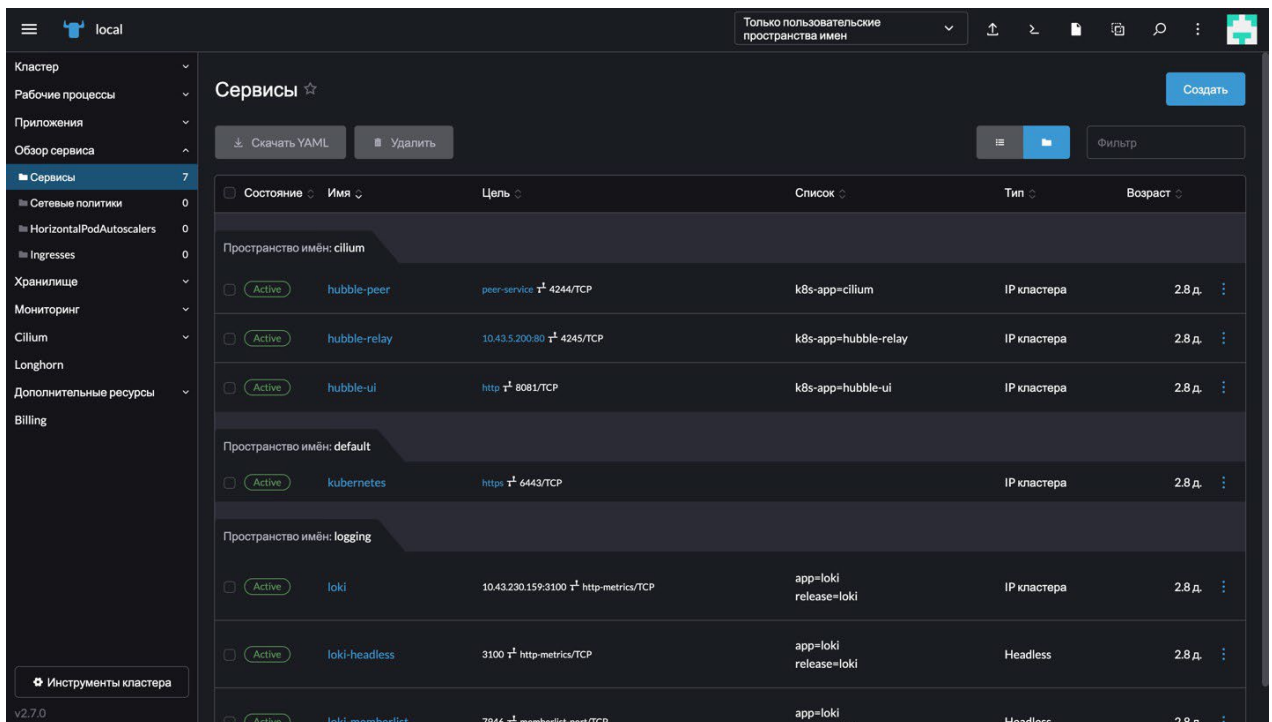


Рис. 19 Пункт меню «Сервисы»

Пункт меню для обзора и создания сервисов в контексте пространств имен и ряда дополнительных настроек.

Подпункты отвечают за функциональность по наименованию.

Сетевые политики позволяют просматривать и создавать правила управление трафиком.

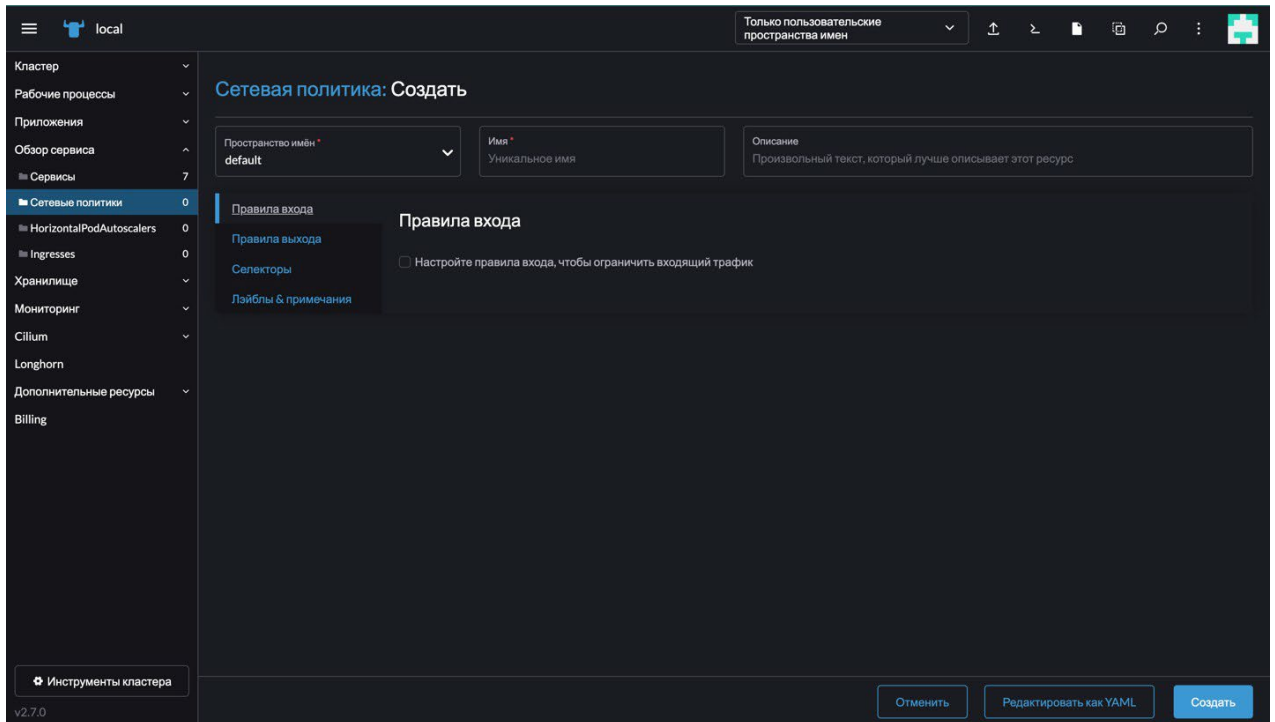


Рис. 20 Подпункт меню «Сетевые политики».

HorizontalPodAutoscalers позволяет просматривать и управлять условиями горизонтального масштабирования контейнеров.

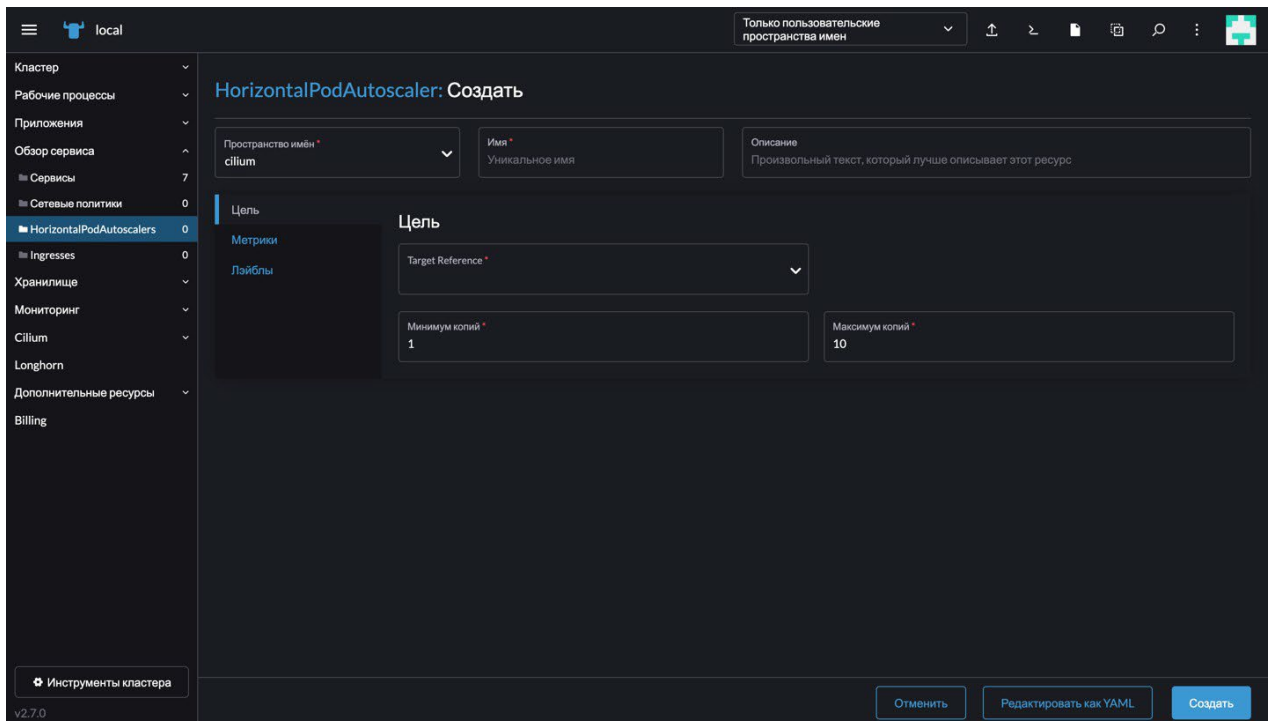


Рис. 21 Подпункт меню «HorizontalPodAutoscalers».

Подпункт меню «Ingresses» - позволяет просматривать и создавать политики маршрутизации, настраивать маршруты.

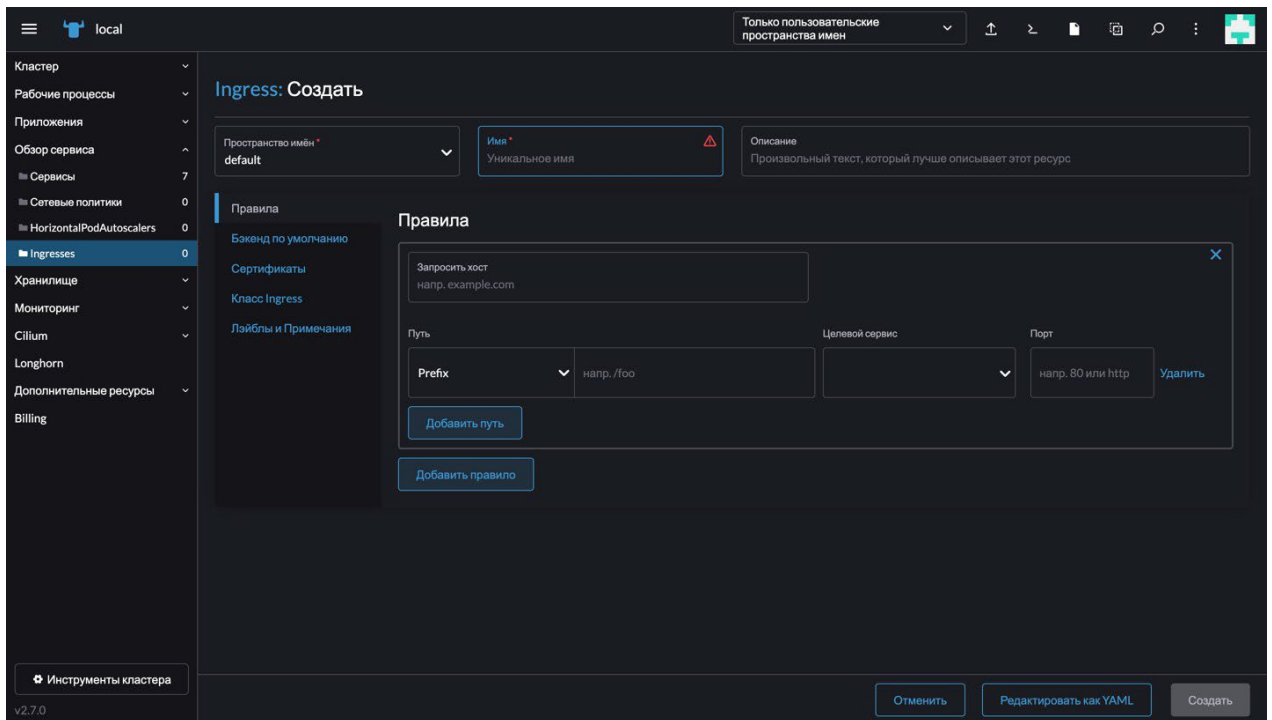


Рис. 22 Подпункт меню «Ingresses».

4.7 ХРАНИЛИЩЕ

Пункт меню «Хранилище» позволяет просматривать, добавлять и удалять системы хранения данных, создавать и управлять томами.

PersistentVolumes - стандартная сущность в контексте K8s, позволяющая определить место хранения данных.

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

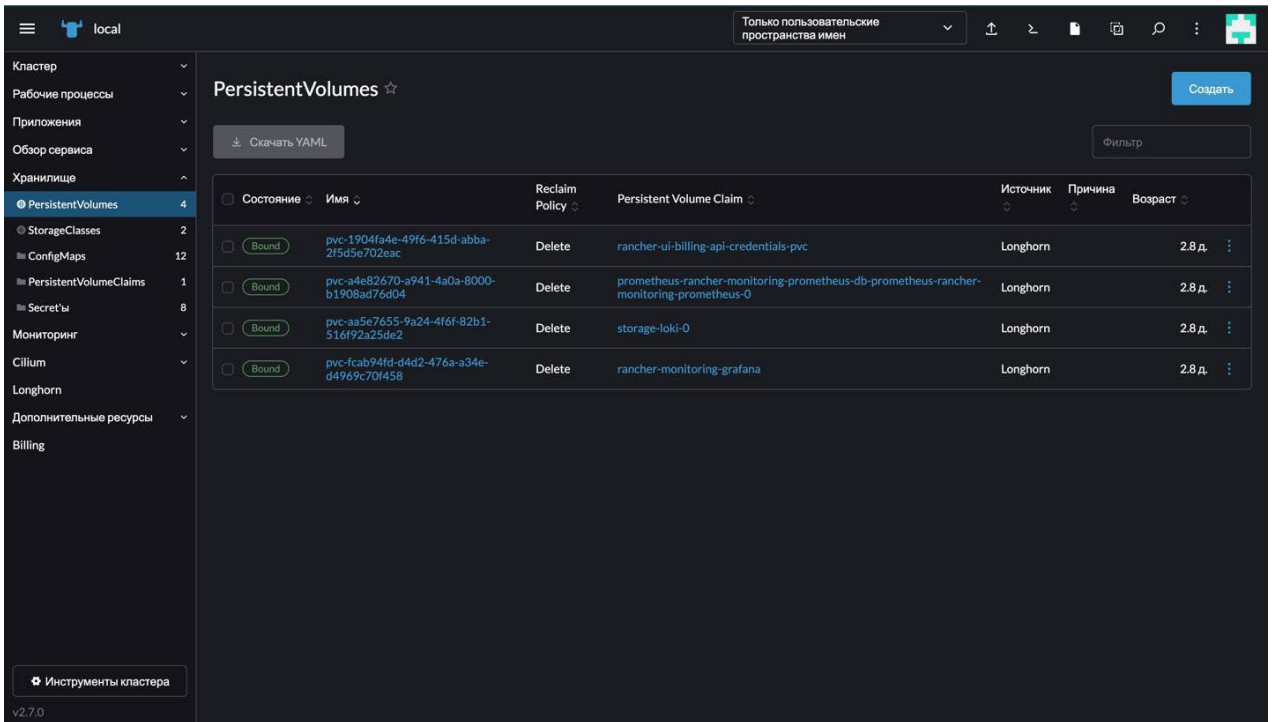


Рис. 23 Основное окно PersistentVolumes.

StorageClasses - описывает хранение параметров подключения к системе хранения данных.

ConfigMaps - однозначно дифференцирует данные образа от конфигурационных переменных, сертификатов

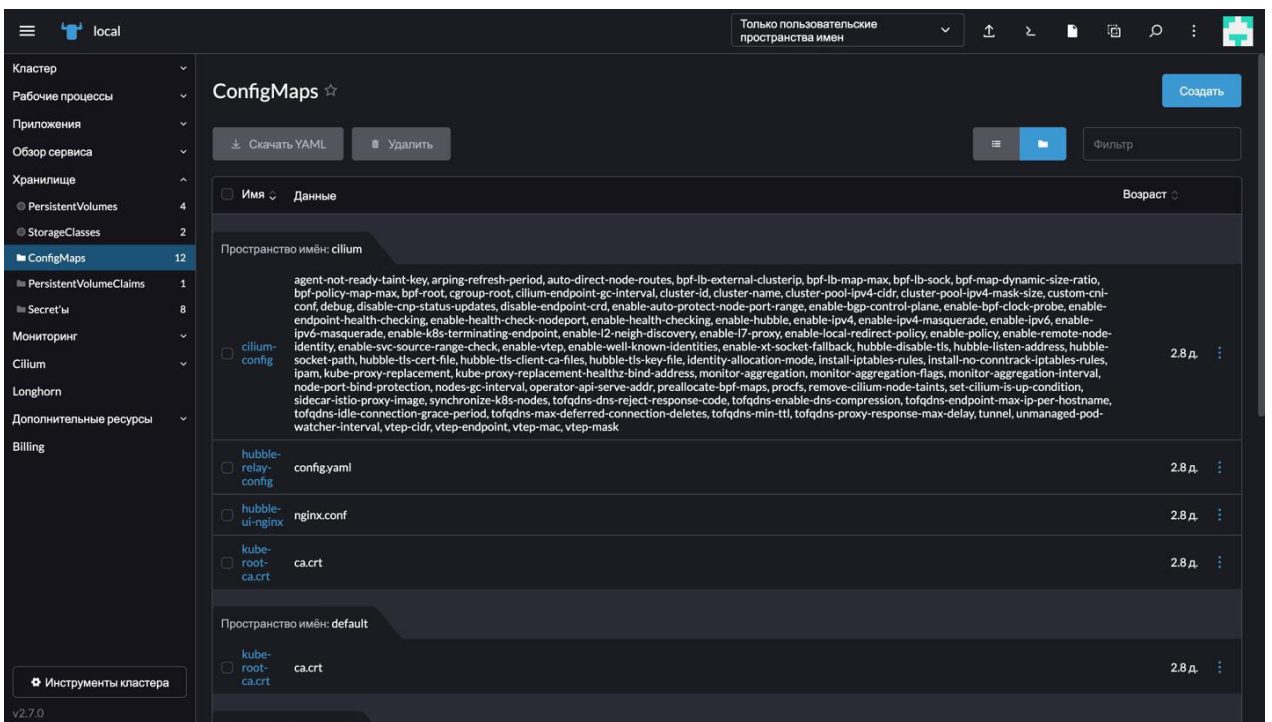


Рис. 24 Основное окно ConfigMaps.

PersistentVolumeClaims - отображение запросов на выделение ресурсов хранилища

Secret's - просмотр и управление конфиденциальной информацией кластера

4.8 МОНИТОРИНГ

Пункт меню, позволяющий просматривать и настраивать метрики мониторинга средствами встроенного в платформу ПО.

В качестве стандарта де-факто используется prometheus, grafana, alertmanager.

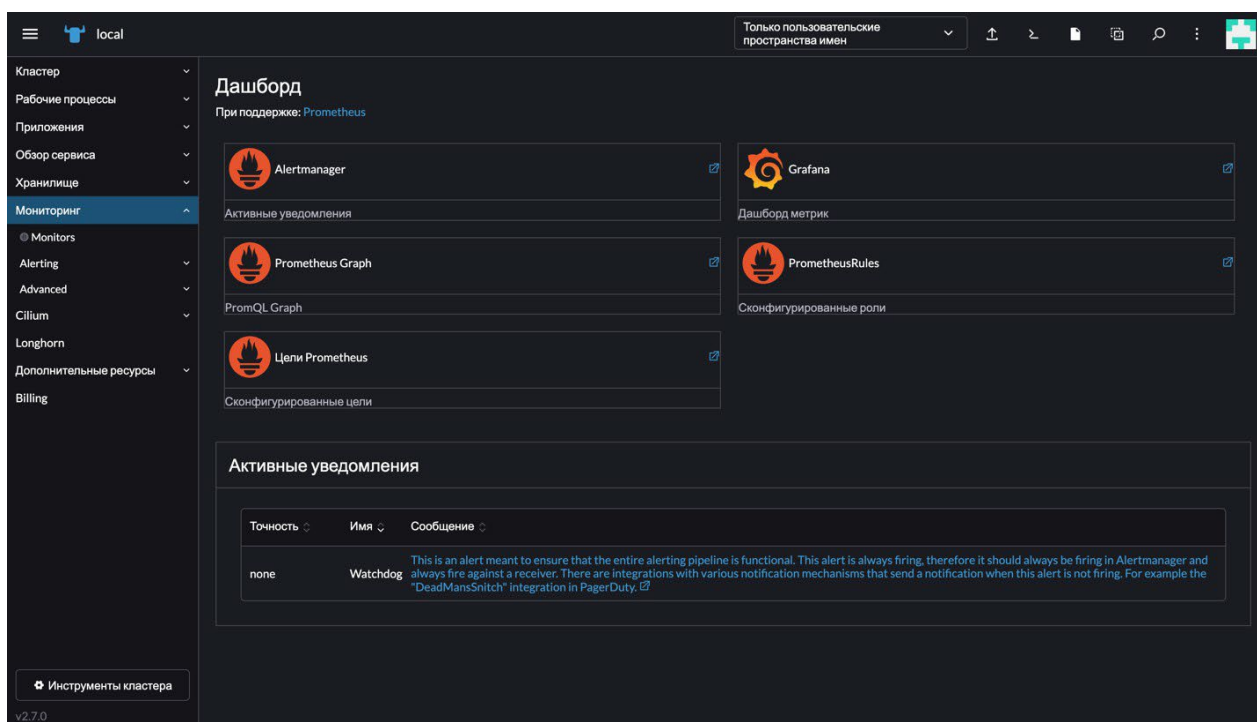


Рис. 25 Пункт меню «Мониторинг». Переход в окна управления осуществляется по клику по ссылке.

Подпункты меню позволят тонко настроить метрики сбора журналов событий, систему оповещений и необходимые дашборды.

4.9 CILLIUM

Проект с открытым исходным кодом, включенный в пакет установки платформы, который обеспечивает сетевое взаимодействие, безопасность и доступность облачных сред.

Cillium имеет удобный интуитивно-понятный графический интерфейс настроек политик сетевого трафика.

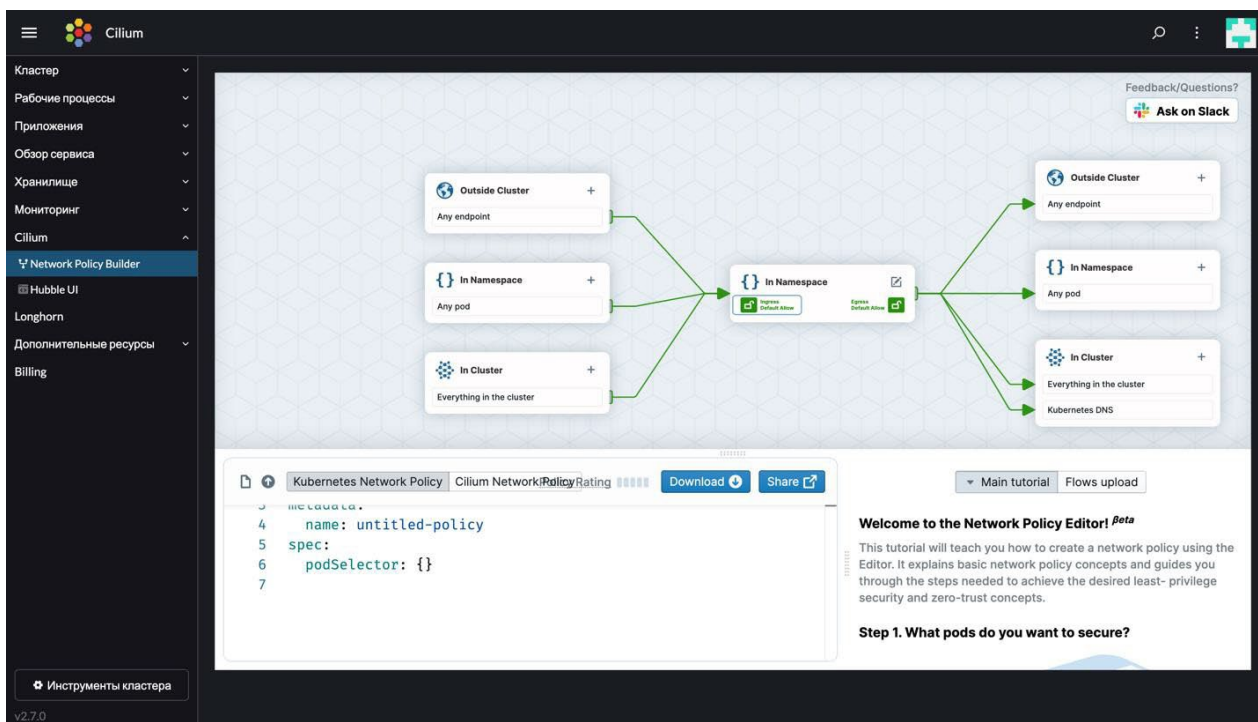


Рис. 26 Основное окно Network Policy Builder в графическом формате представляет маршрут входящего и исходящего трафика.

Изменение политик формирует YAML файл, готовый к применению.

Применение созданных политик осуществляется посредством применения файла через консоль kubectl, загрузкой готового файла через элемент интерфейса.

Пример использования приведен в разделе далее.

Подпункт Hubble UI позволяет отслеживать и управлять сетевым взаимодействием и потоками данных в разрезе контейнеров/сервисов.

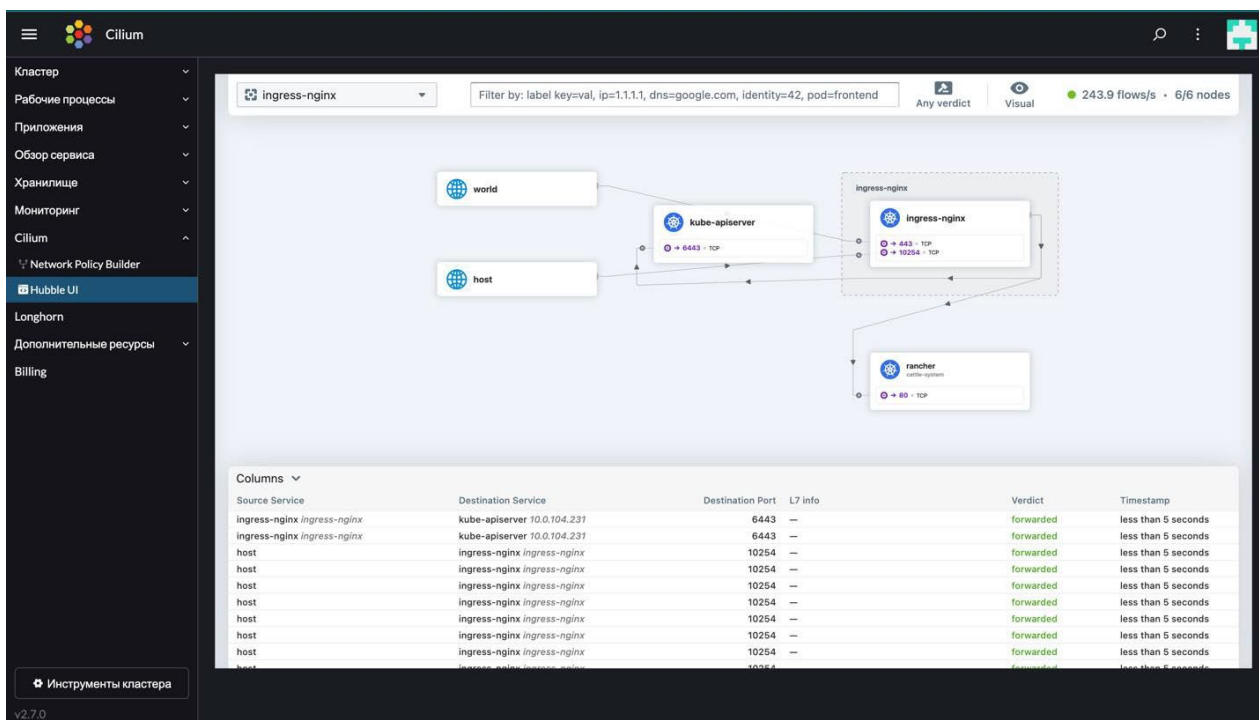


Рис. 27 Hubble UI. Пример отображения потока данных трафика ingress-nginx.

4.10 LONGHORN

Преднастроенное в ходе установки распределенное, отказоустойчивое хранилище.

В пункте меню содержится ссылка на веб-интерфейс, отражающий текущее состояние, просмотр и настройка нод, томов, планировщик заданий, настройка резервного копирования и общие настройки:

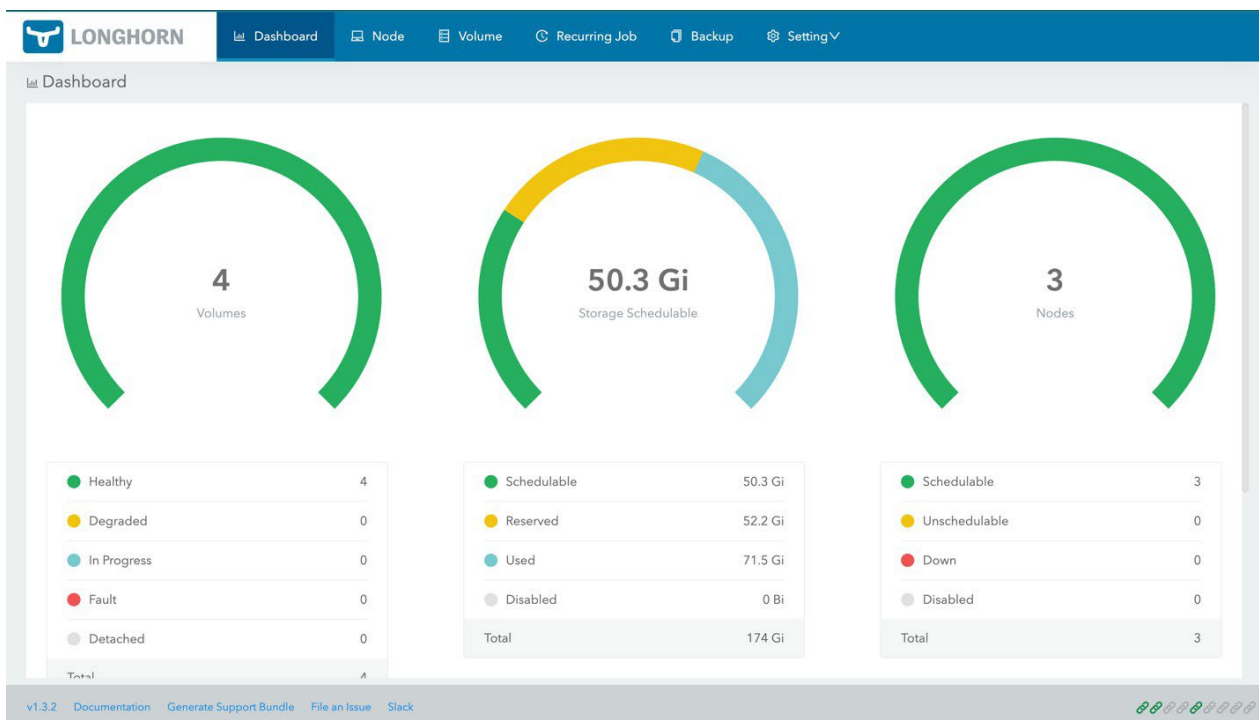


Рис. 28 Основное окно модуля Longhorn.

В рамках настоящего руководства администрирование модуля не рассматривается, исчерпывающая документация по работе с модулем приведена в документации на сайте <https://longhorn.io/docs>

4.11 ДОПОЛНИТЕЛЬНЫЕ РЕСУРСЫ

Пункт меню «Дополнительные ресурсы» включает в себя перечень не вошедших в основные блоки элементы управления кластером модулей, направленных на тонкую настройку кластера и его элементов.

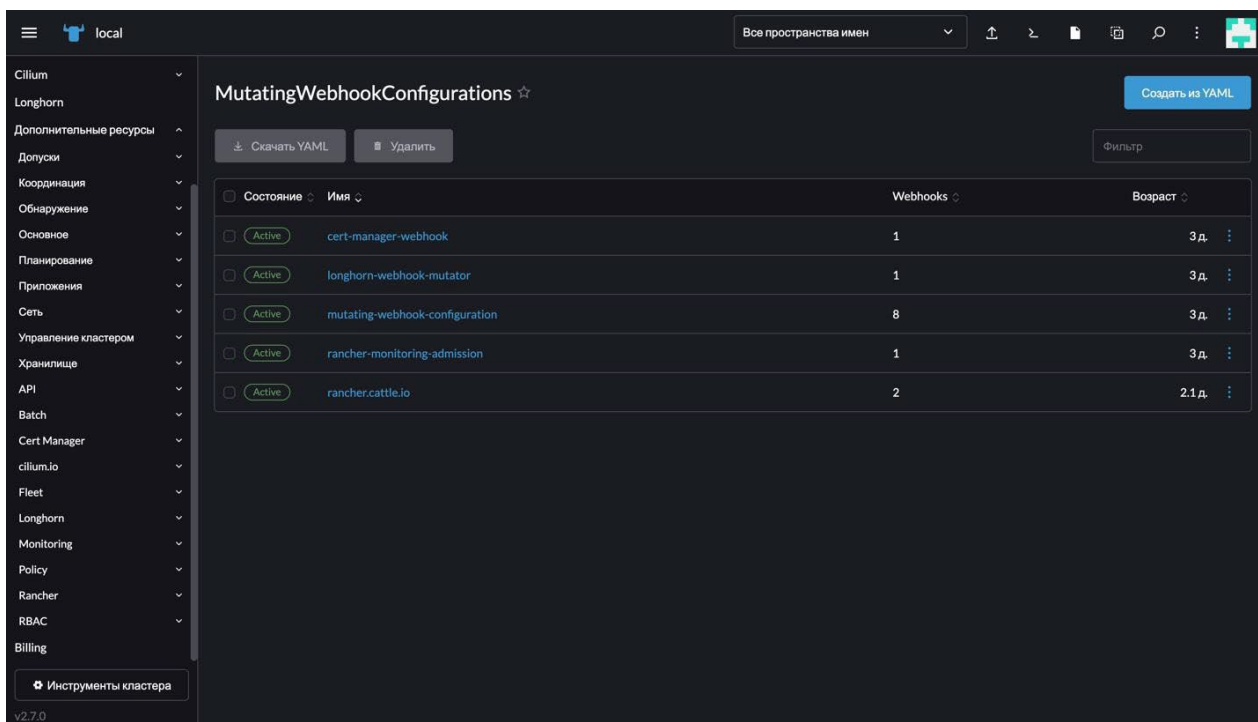


Рис. 29 Перечень подпунктов пункта меню «Дополнительные настройки»

4.12 BILLING

Раздел позволяет визуализировать интенсивность потребления вычислительных ресурсов кластером, выстраивая графики потребления в разрезах процессорных мощностей, оперативной памяти, количестве операций/транзакций за заданную единицу времени.

4.13 ИНСТРУМЕНТЫ КЛАСТЕРА

В разделе содержатся как уже установленные, так и готовые к установке дополнительные компоненты организации функционирования кластера.

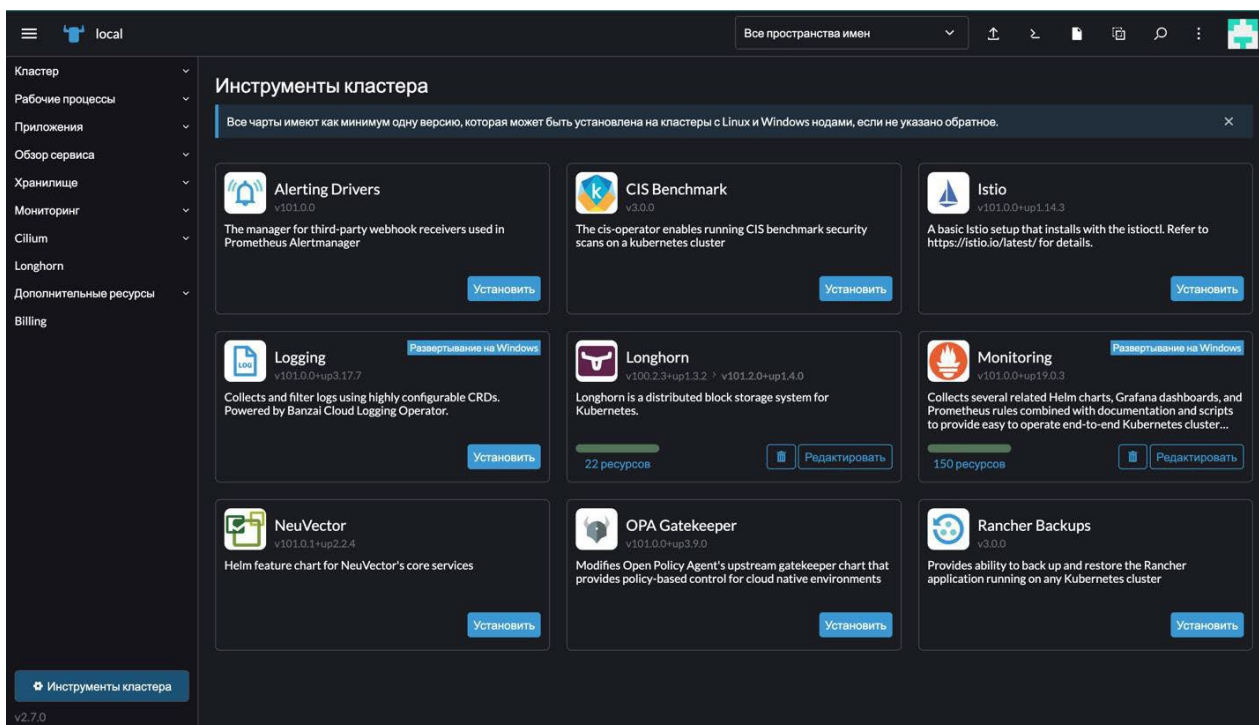


Рис. 30 «Инструменты кластера»

Пул компонентов представляет собой набор специализированного ПО, использование которого регламентировано процессами на стороне конечного пользователя: информационная безопасность, сохранность данных, контроль инфраструктуры и т.п.

5 ИНТЕРФЕЙС УПРАВЛЕНИЯ. ГЛОБАЛЬНЫЕ ПРИЛОЖЕНИЯ

В блоке меню "Глобальные приложения" объединены элементы управления отвечающие за процессы непрерывной доставки (Continuous delivery(CD), Управления кластером и Управления виртуализацией.

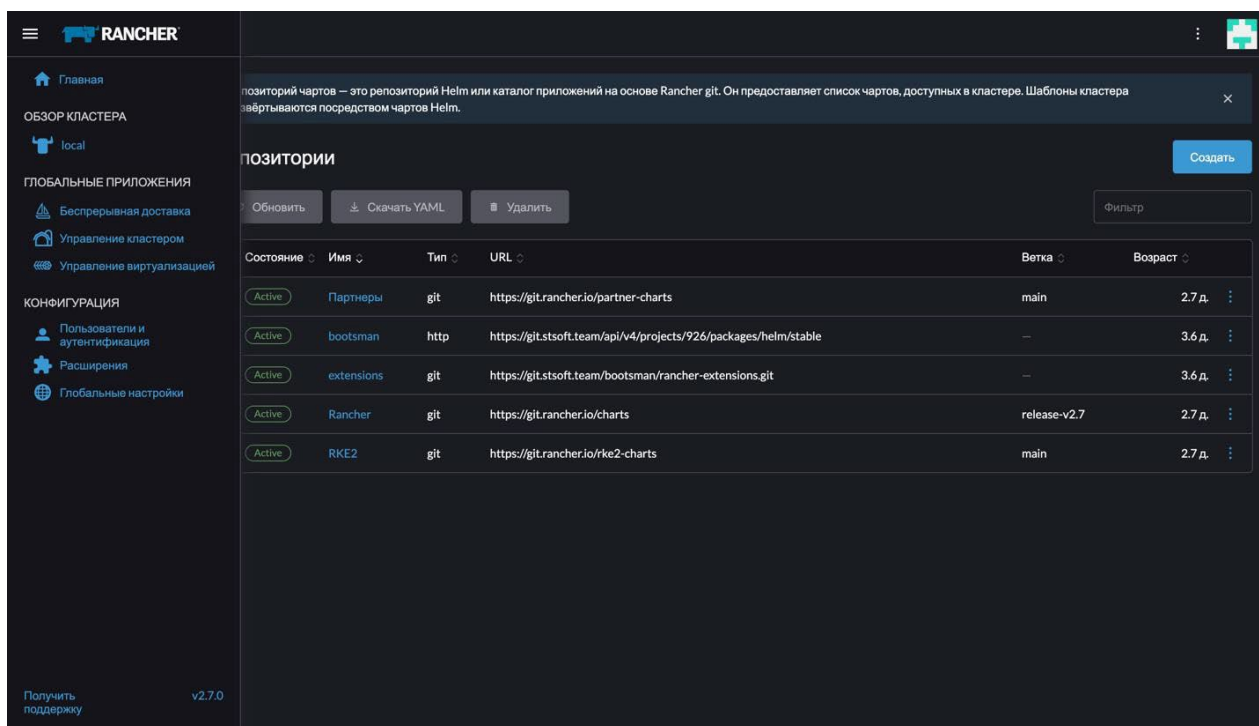


Рис. 31 Блок меню "Глобальные приложения"

5.1 БЕСПРЕРЫВНАЯ ДОСТАВКА

Пункт меню объединяет в себе настройки Continuous Delivery, и позволяет их конфигурировать в случае необходимости.

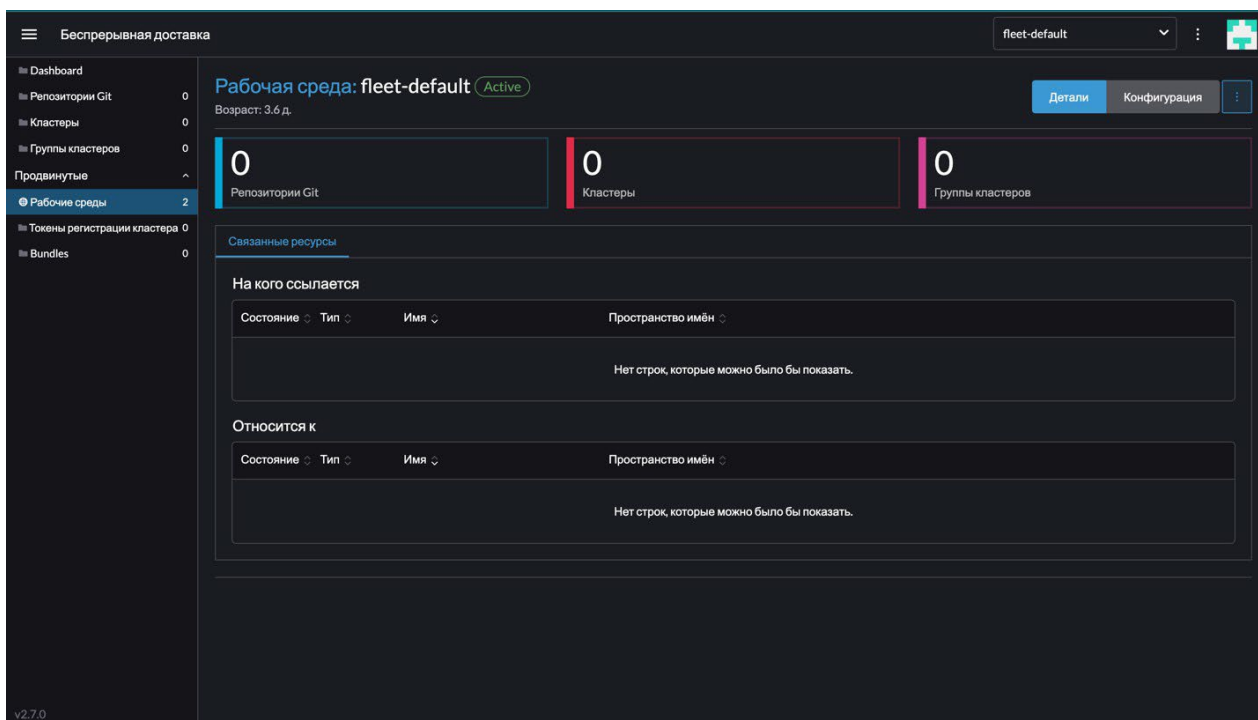


Рис. 32 Пункт меню «Бесперывная доставка»

5.2 УПРАВЛЕНИЕ КЛАСТЕРОМ

Пункт меню объединяет в себе настройки управления кластером в разрезе размещения, учетных данных облачных провайдеров, используемых драйверов облачных провайдеров и настроек репозитория.

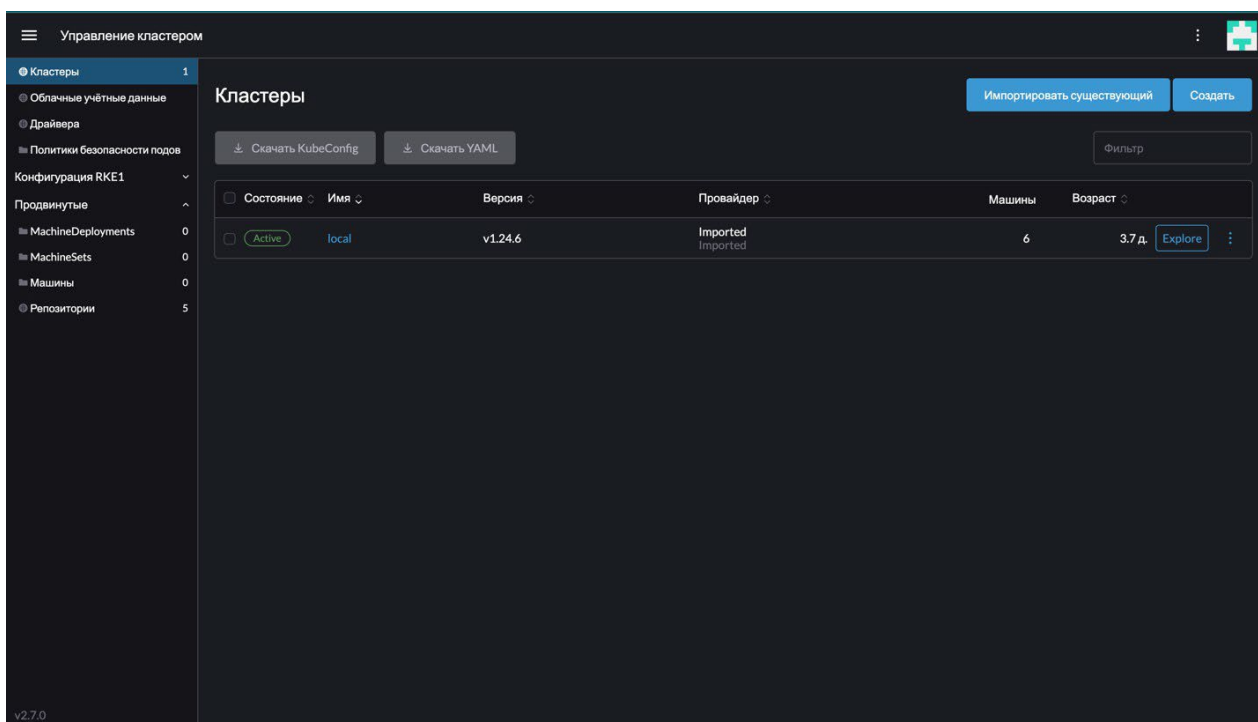


Рис. 33 Управление кластером.

5.3 УПРАВЛЕНИЕ ВИРТУАЛИЗАЦИЕЙ

Пункт меню отвечает за управление виртуализацией в случае размещения кластера на физических серверах «Bare metal» под управлением Harvester.

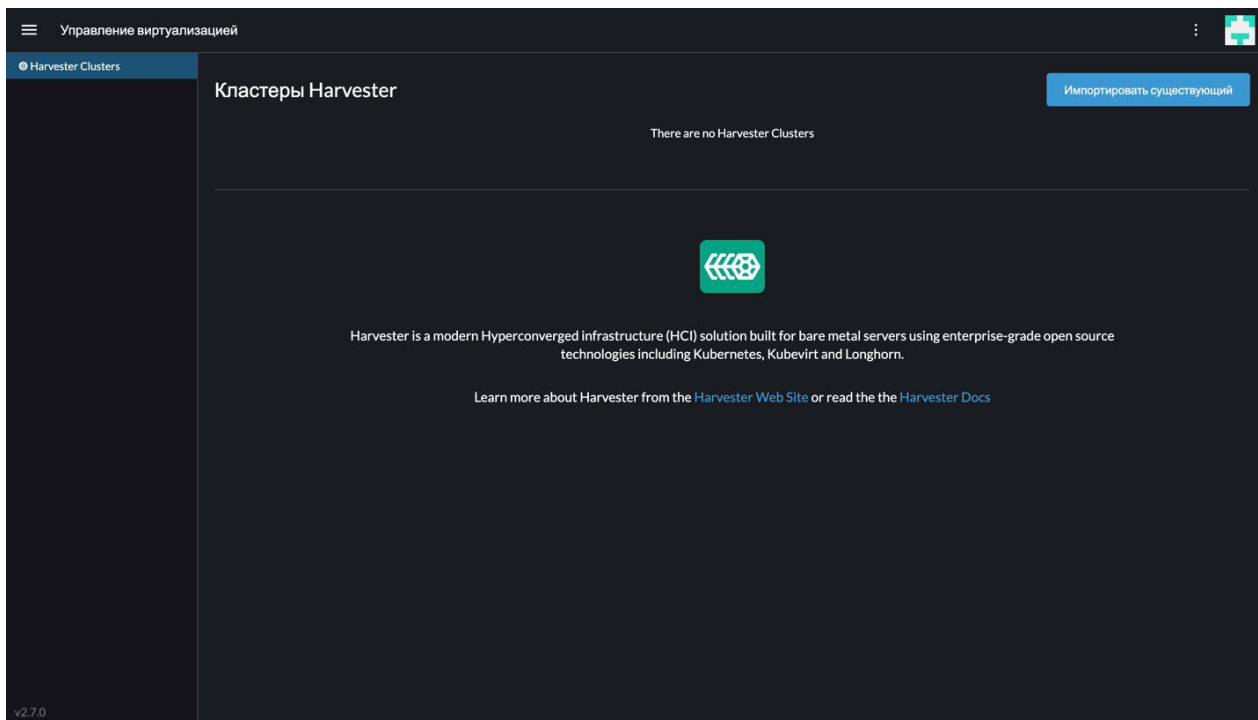


Рис. 34 Управление виртуализацией.

6 ИНТЕРФЕЙС УПРАВЛЕНИЯ. КОНФИГУРАЦИЯ

Блок меню объединяет в себе настройки авторизации и аутентификации пользователей, подключения провайдеров службы каталогов, установки расширений на веб-интерфейс и глобальные настройки веб-интерфейса.

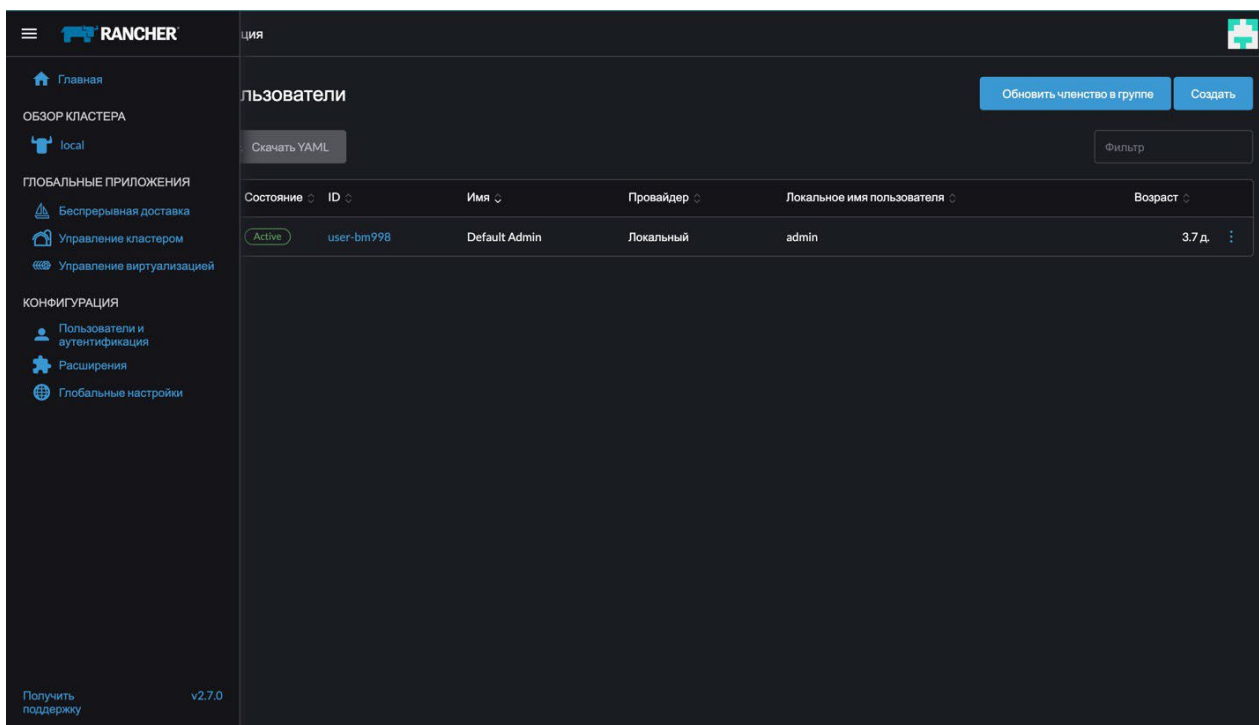


Рис. 35 Блок меню "Конфигурация"

6.1 ПОЛЬЗОВАТЕЛИ И АУТЕНТИФИКАЦИЯ

Блок меню позволяет управлять политикой доступа пользователей и групп пользователей, выдачей прав и контроля доступа как локальным форматом, так и посредством провайдера службы каталогов.

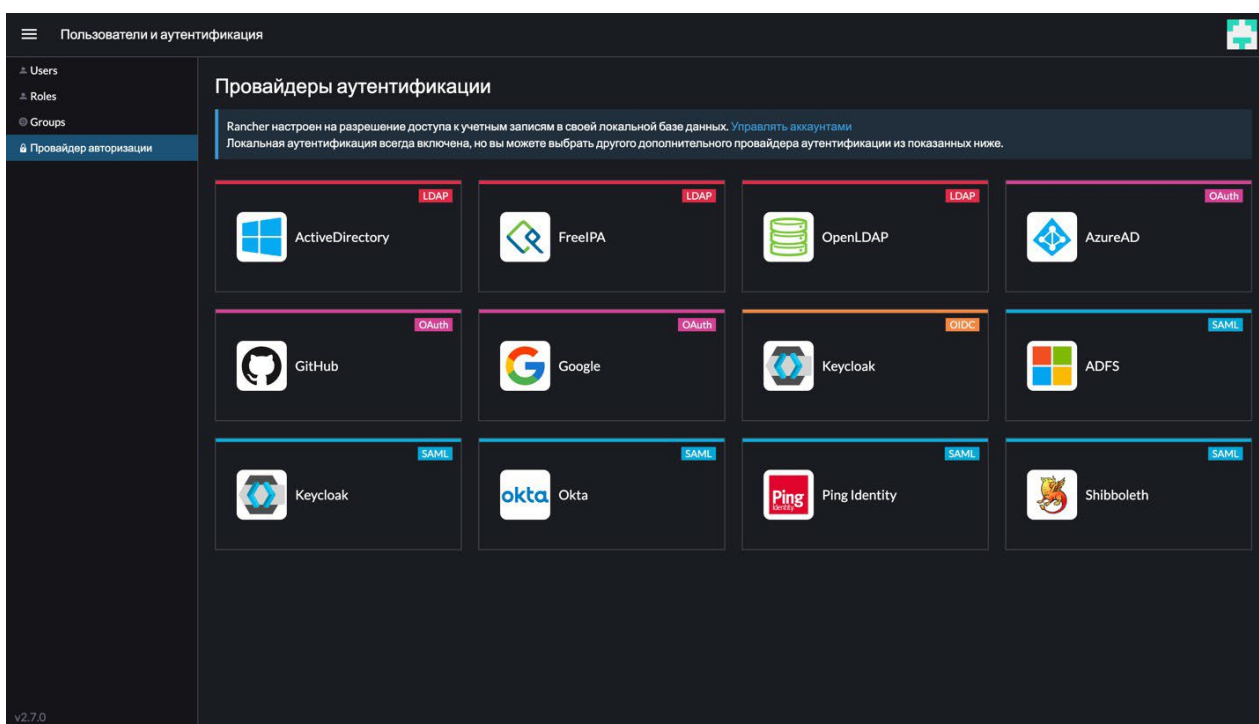


Рис. 36 Блок меню «Пользователи и аутентификация»

6.2 РАСШИРЕНИЯ

Блок меню «Расширения» позволяет управлять расширениями веб-интерфейса.

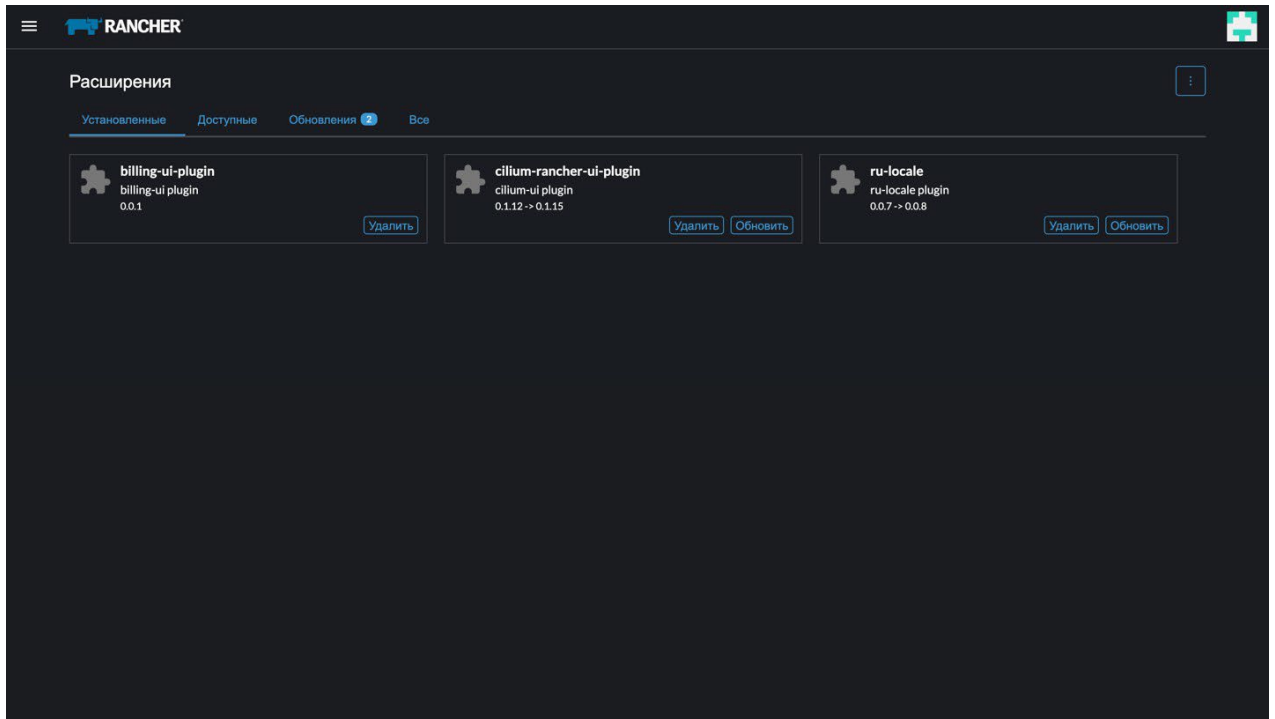


Рис. 37 Блок меню "Расширения"

6.3 ГЛОБАЛЬНЫЕ НАСТРОЙКИ

Блок меню "Глобальные настройки" объединяет в себе тонкую настройку веб-интерфейса, такие как настройки сессии, сертификатов, указание ссылок и т.п.

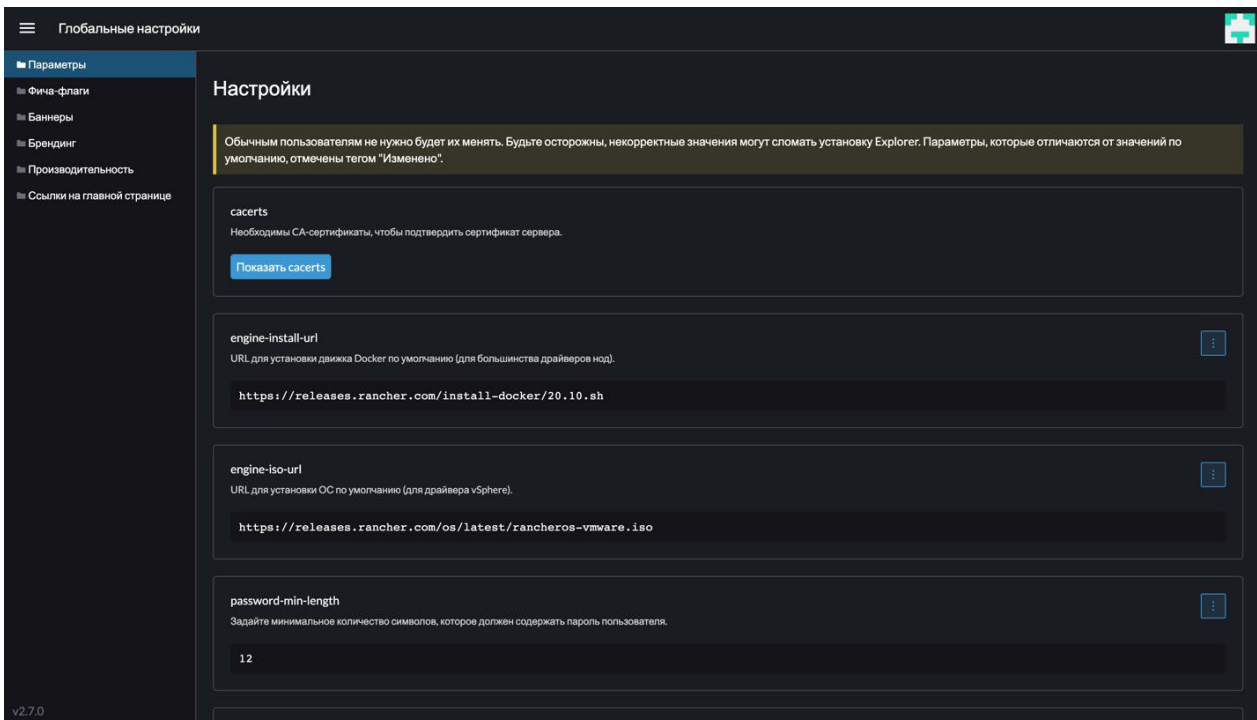


Рис. 38 Блок меню "Глобальные настройки"

6.4 НАСТРОЙКИ ПРОФИЛЯ ПОЛЬЗОВАТЕЛЯ

По клику на элемент профиля в правом верхнем углу рабочей области открывается контекстное меню управления профилем пользователя.

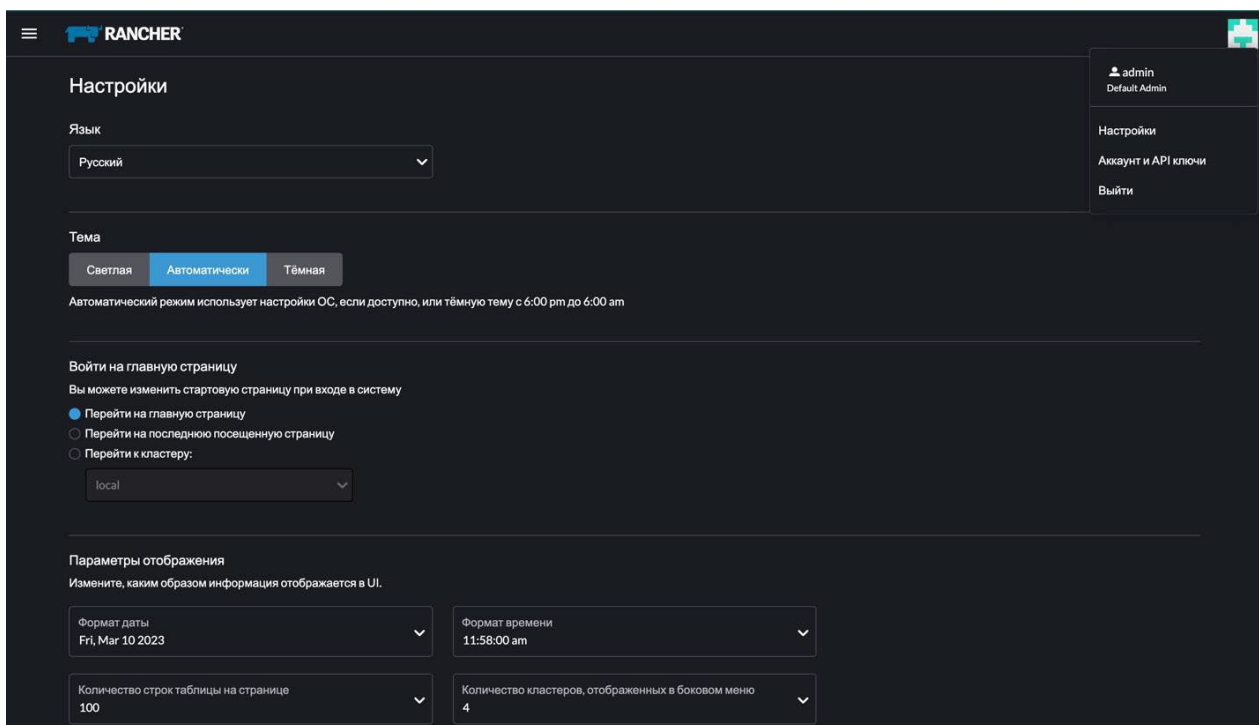


Рис. 39 Блок меню «Настройки» позволяет настроить тему отображения, точку входа при авторизации, настройки даты-времени и т.п.

Подпункт «Аккаунт и API ключи» позволяет сменить пароль на текущем аккаунте и настроить API для взаимодействия.

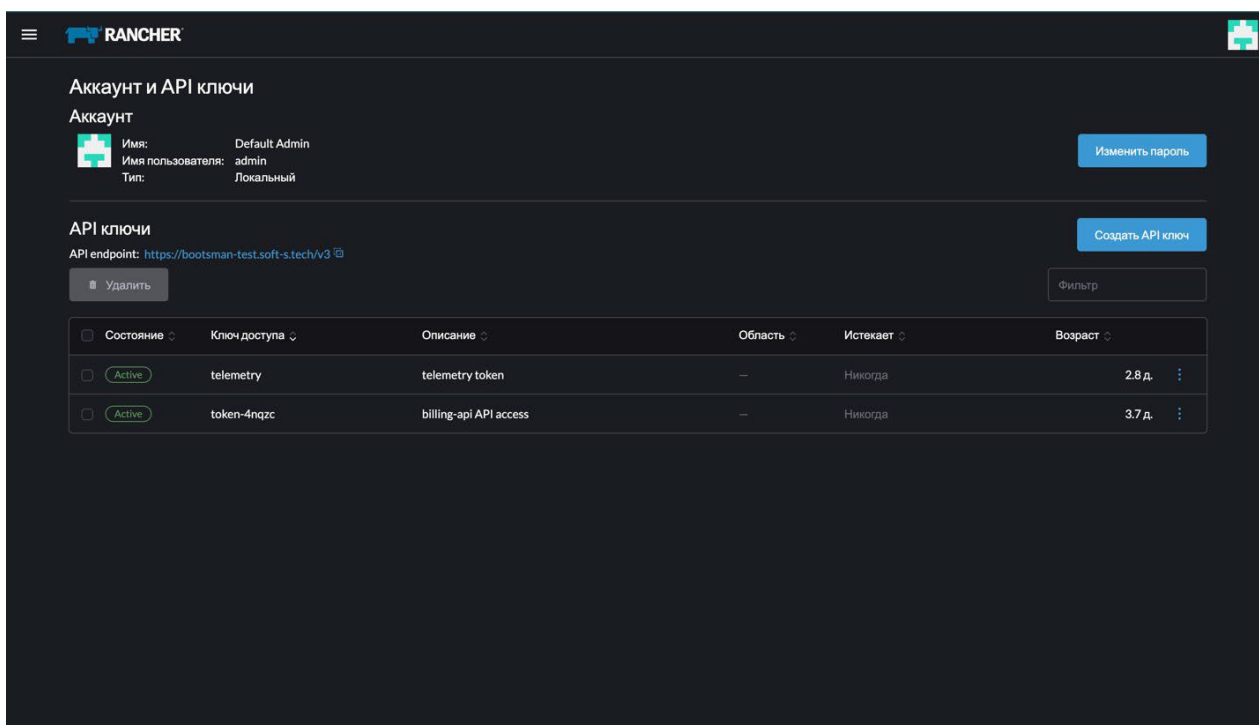


Рис. 40 Блок меню «Аккаунт и API ключи».

7 ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ

7.1 УСТАНОВКА ПРИЛОЖЕНИЯ ИЗ ЧАРТА

Для установки приложения из чарта необходимо перейти в пункт меню "Обзор кластера"→[Имя кластера]→ Приложения→ Charts.

В выпадающем списке "Все" приведены доступные репозитории, из которых возможно выполнить установку приложений (на скриншоте ниже выбраны все доступные).

Для установки предустановленных тестовых экземпляров необходимо выбрать репозиторий «bootzman».

Следует отметить, что установка приложений из данного меню не требует углубленных знаний синтаксиса и логики написания YAML-конфигураций.

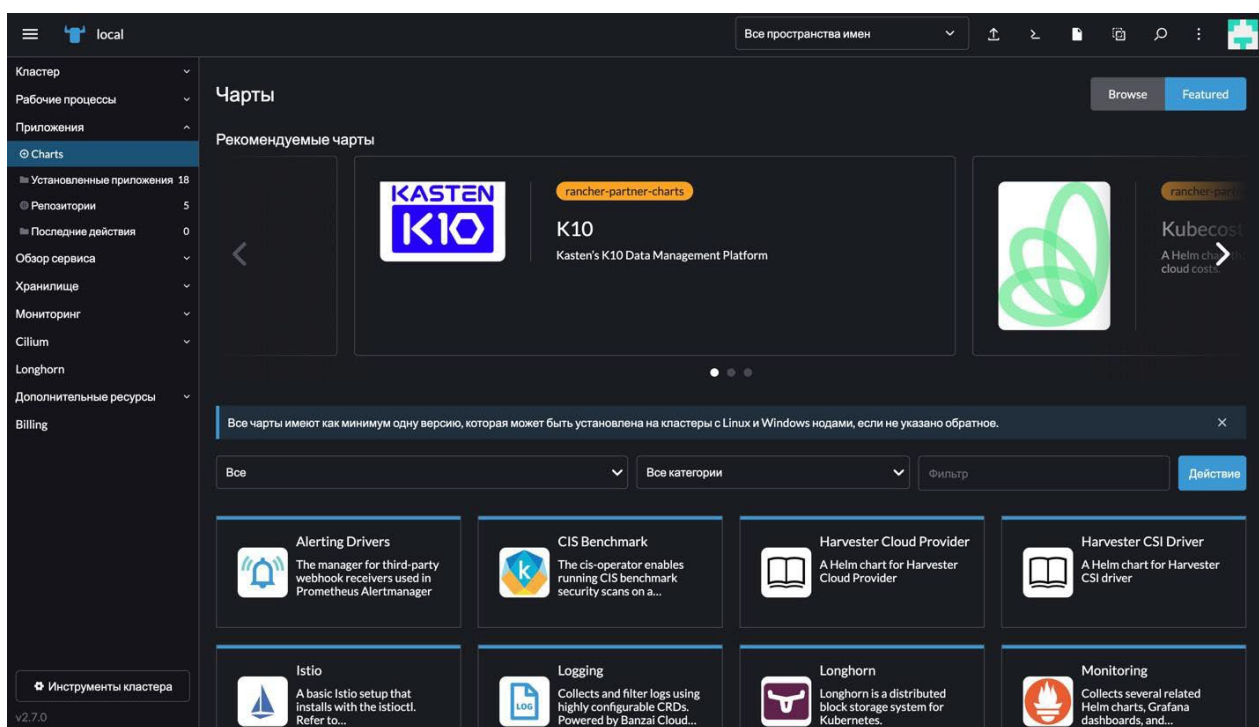


Рис. 41 Пункт меню «Charts»

Далее - выбрать приложение для установки (в качестве примера будет использован PostgreSQL).

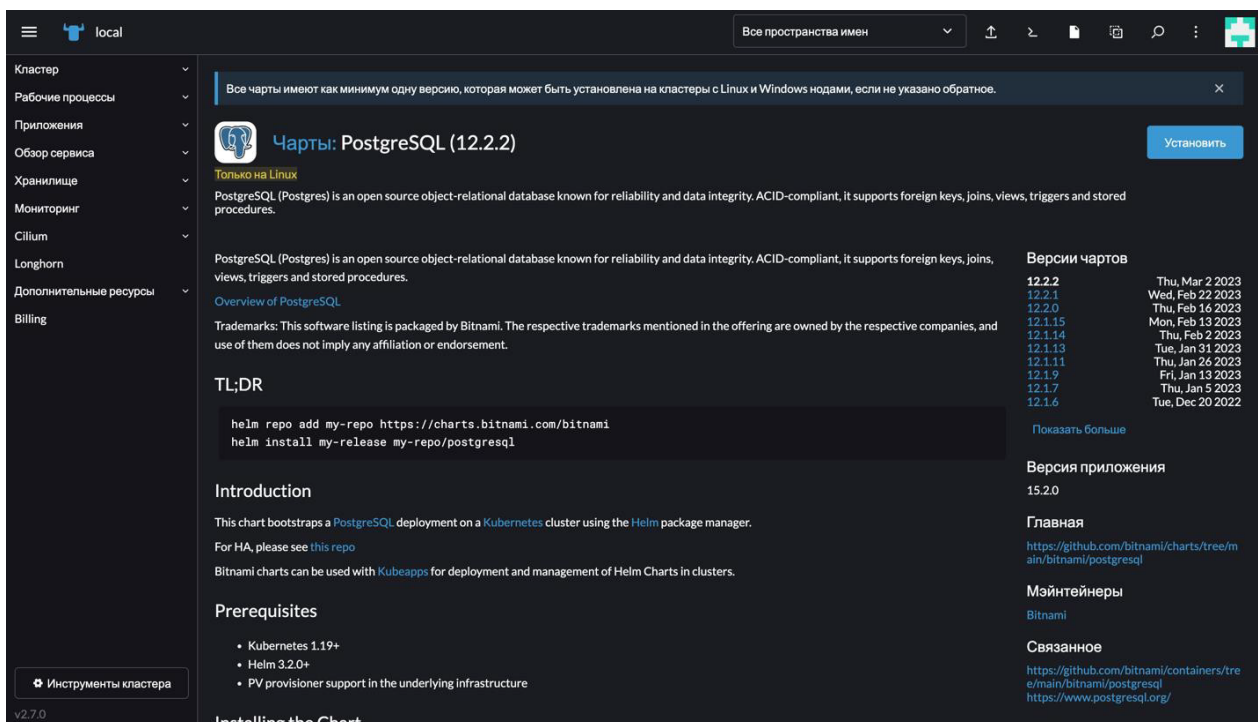


Рис. 43 Выбор приложения.

До момента установки можно ознакомиться с описанием, доступным из чарта Helm, выбрать версию и начать процесс установки по кнопке «Установить»

На первом окне установки необходимо выбрать пространство имен, которое будет использовать приложение

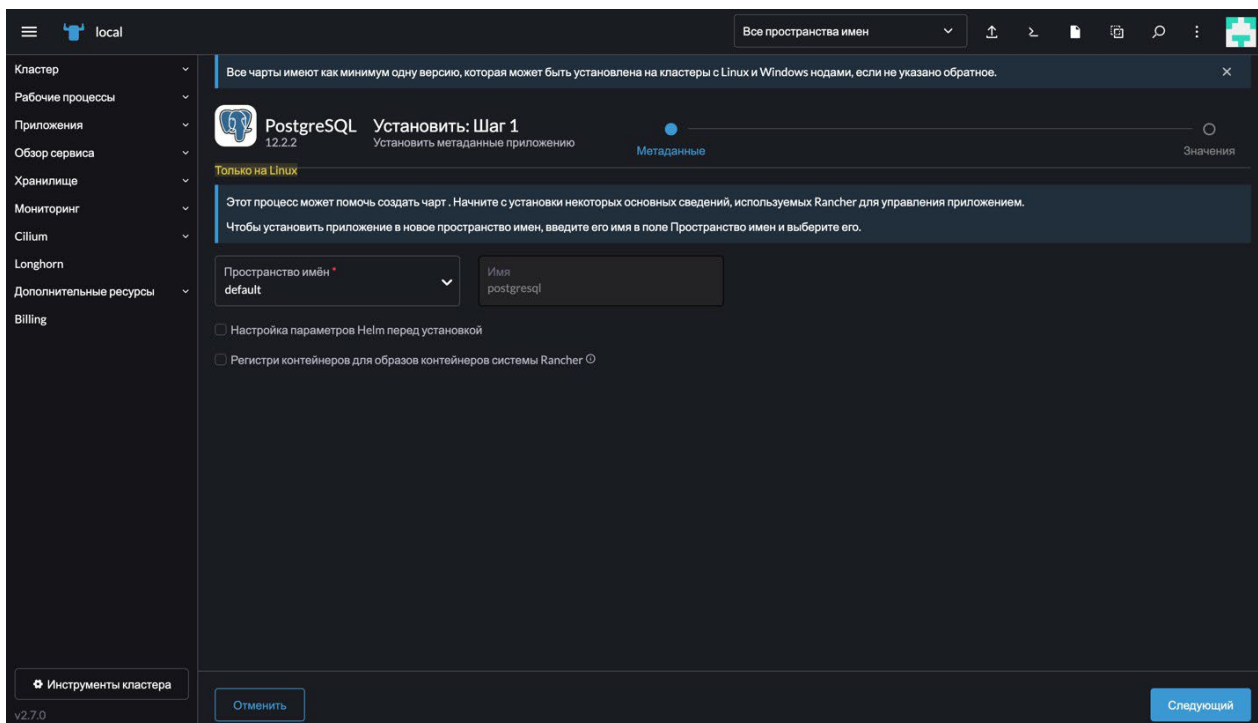


Рис. 44 Окно установки.

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Для соблюдения консистентности, разграничения данных и безопасности рекомендуется использовать отдельное пространство имен.

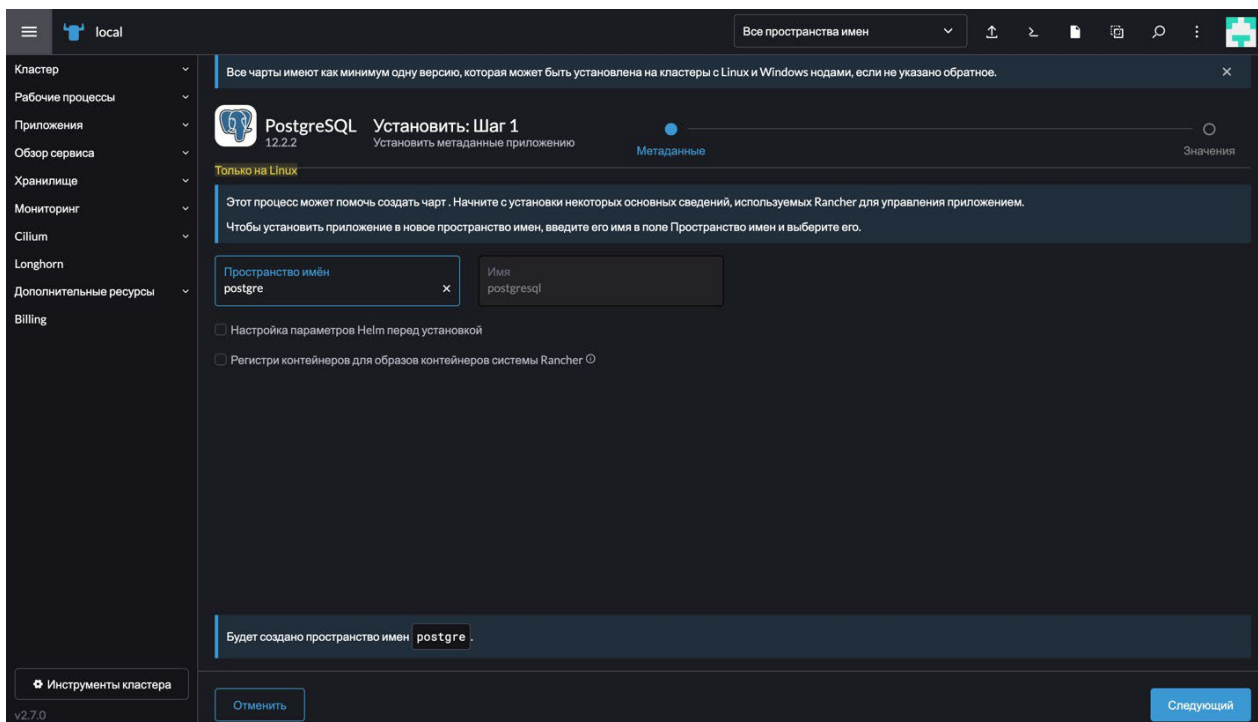


Рис. 45 Создание пространства имен доступно в том же окне.

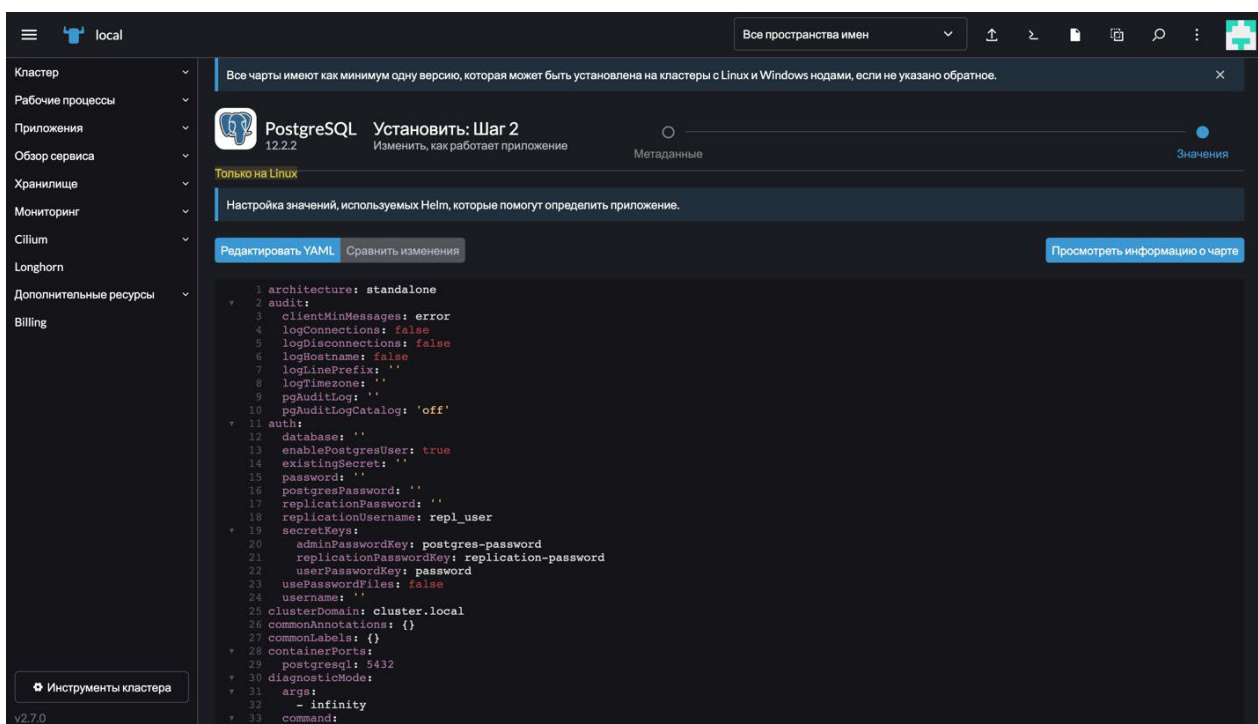
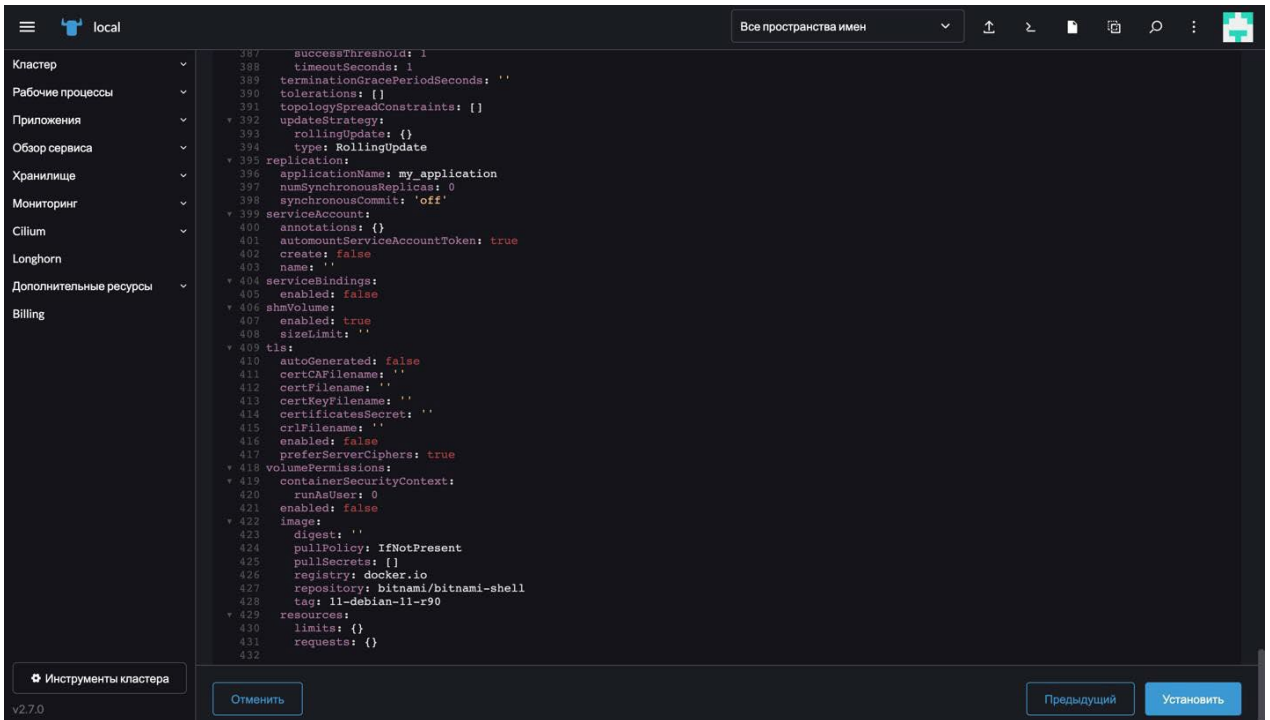


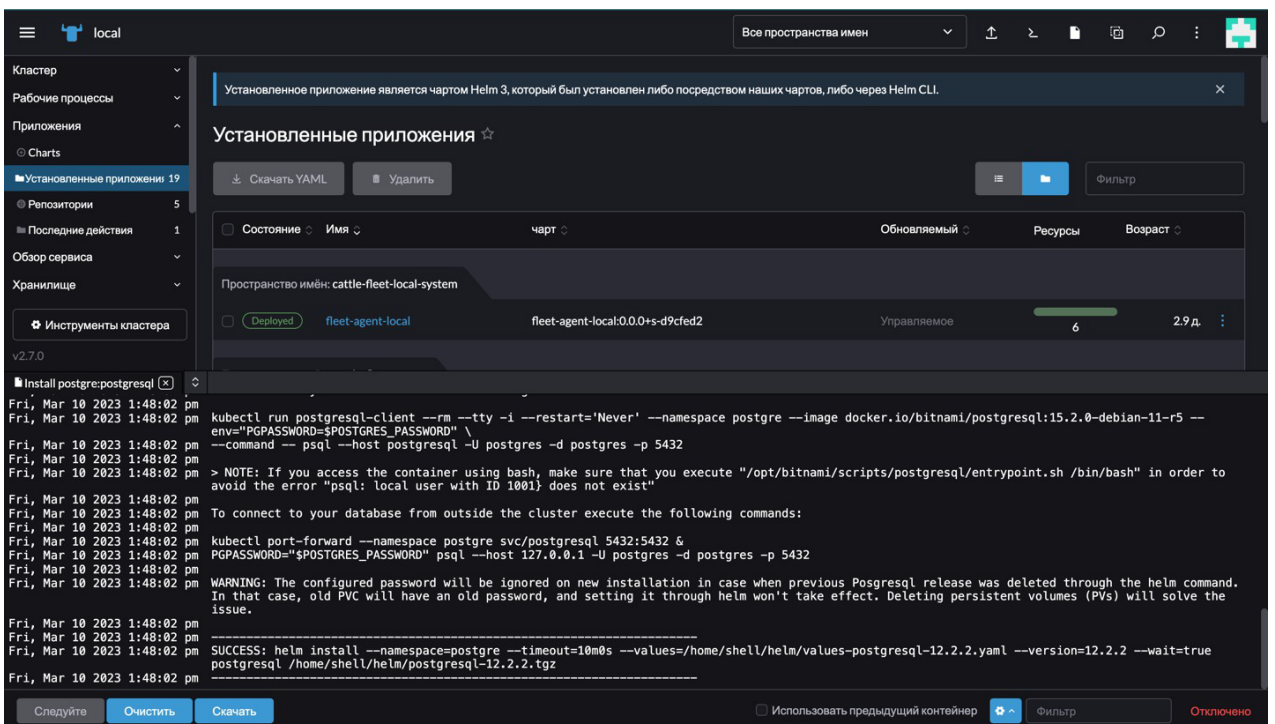
Рис. 46 При переходе на следующий шаг платформа сформирует YAML-файл, готового к установке приложения. При необходимости файл можно отредактировать.

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ



Установка проходит в полу-интерактивном режиме.

В нижней части экрана открывается окно с журналом установки, содержащее результат выполнения.



ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

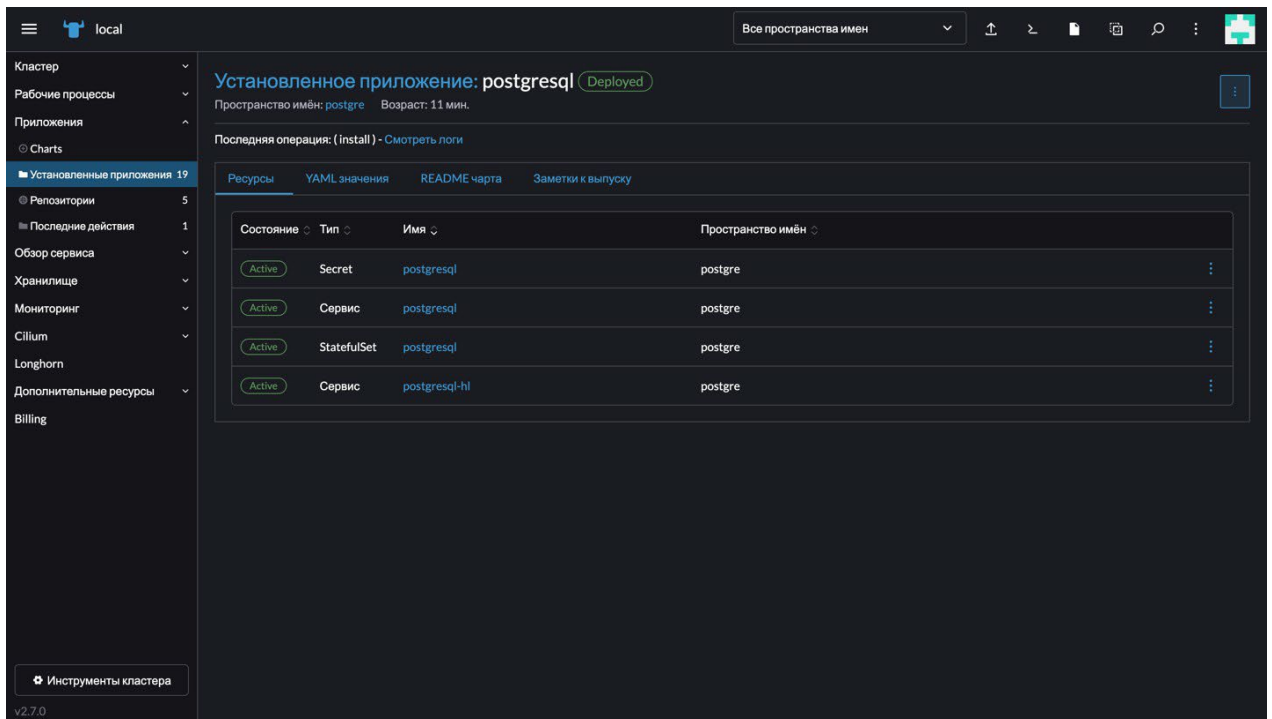


Рис. 49 Отображение приложения в установленных.

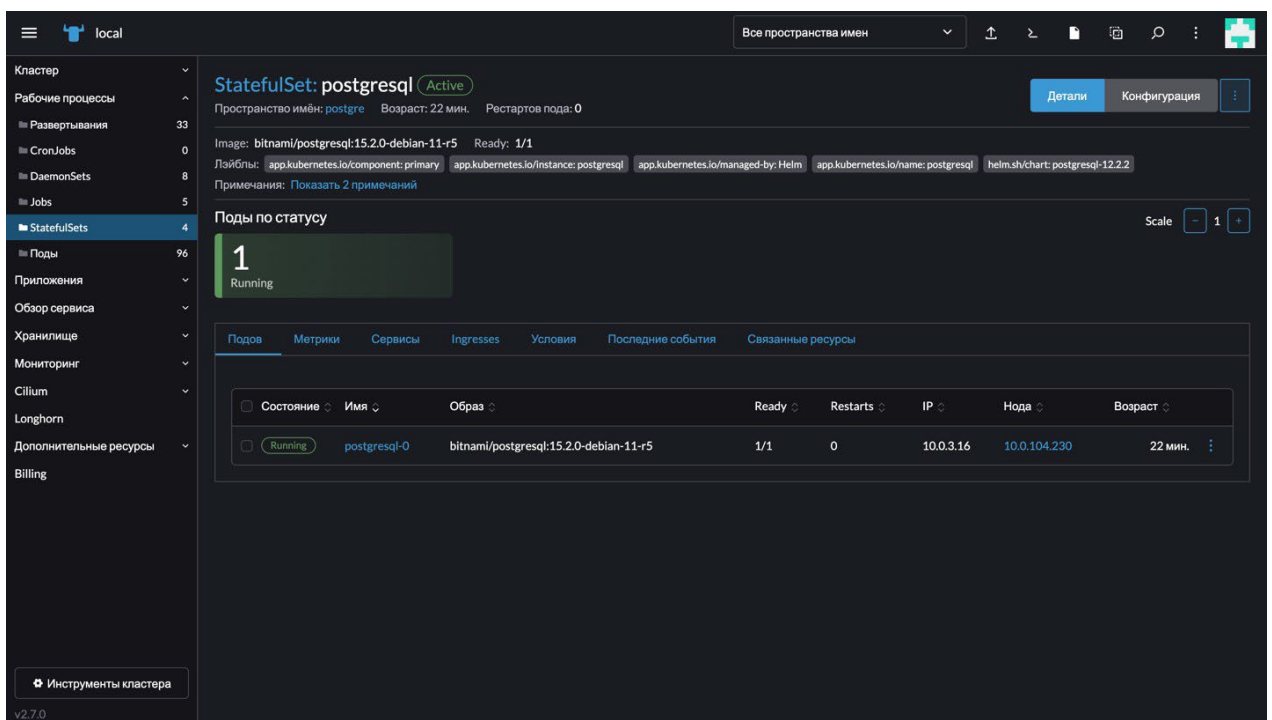


Рис. 50 Отображение запущенного приложения в пункте меню «Рабочие процессы»

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Через контекстное меню доступны опции выполнения команд в оболочке окружения приложения, просмотра журналов событий, редактирование параметров конфигурации, клонирование и удаление приложения.

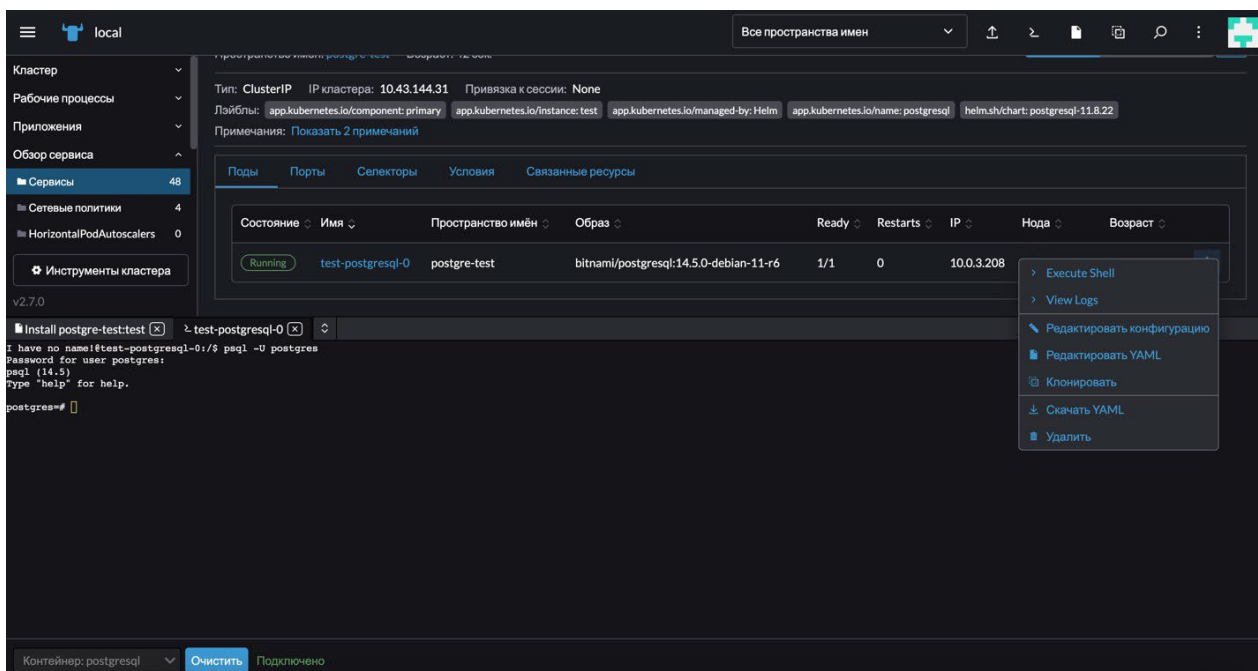


Рис. 51 Контекстное меню приложения.

7.2 УСТАНОВКА ПРИЛОЖЕНИЯ С ИСПОЛЬЗОВАНИЕМ KUBECTL

Ремарка: установка данного типа подразумевает наличие у оператора знаний синтаксиса и структуры YAML-конфигураций и общего понимания функционирования kubernetes.

Установка приложения с использованием kubectl доступна с помощью импорта YAML-файла по кнопке с пиктограммой «Импортировать из YAML».

Предварительно, перед установкой необходимо создать секрет для хранения конфиденциальной информации(пароль) пространство имен, том который будет использоваться приложением.

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

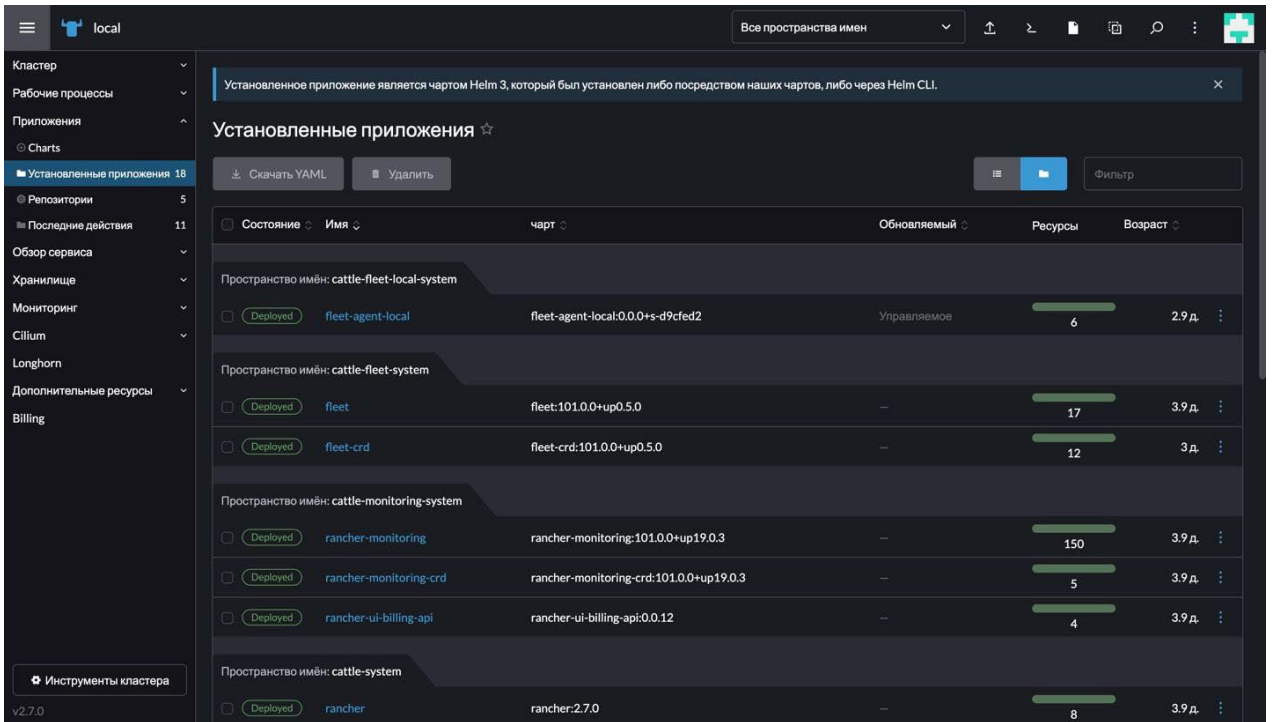


Рис. 52 Импорт из YAML.

Платформа поддерживает как самостоятельное написание файла, так и импорт из существующего.

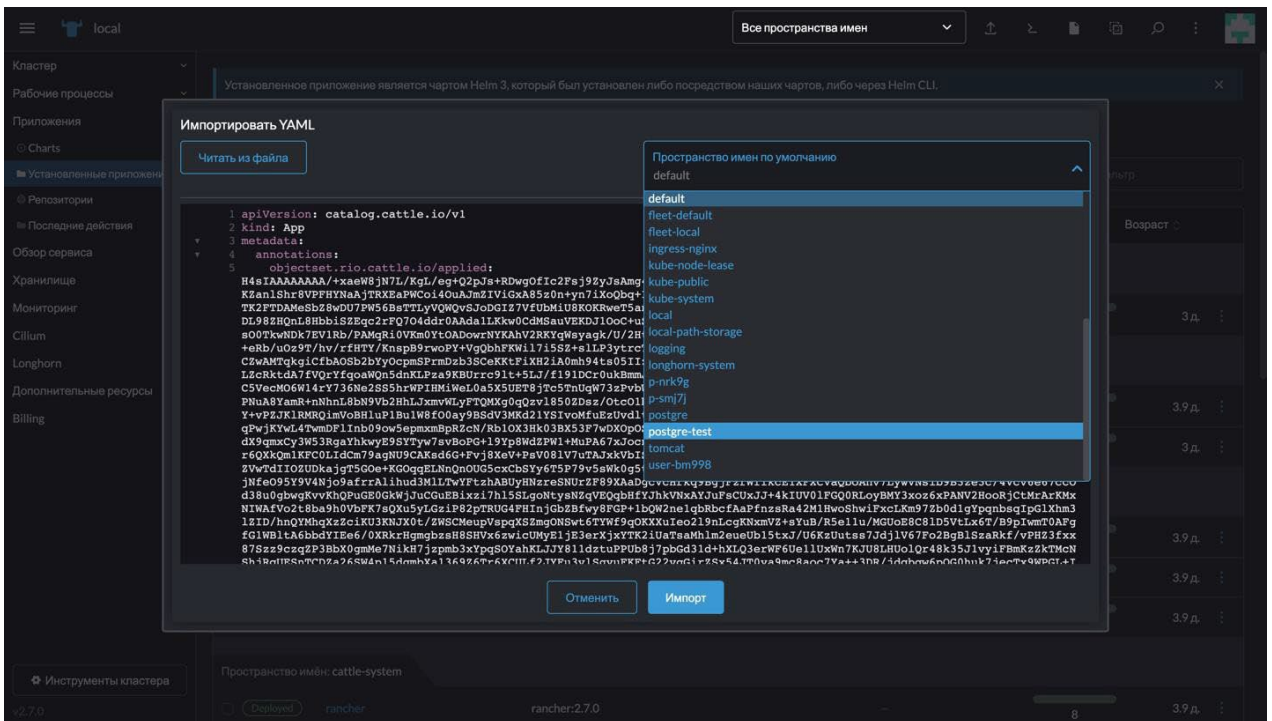


Рис. 53 Импорт из YAML. Выбор пространства имен.

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

После заполнения формы и нажатия на кнопку "Импорт" происходит синтаксическая проверка и запуск установки.

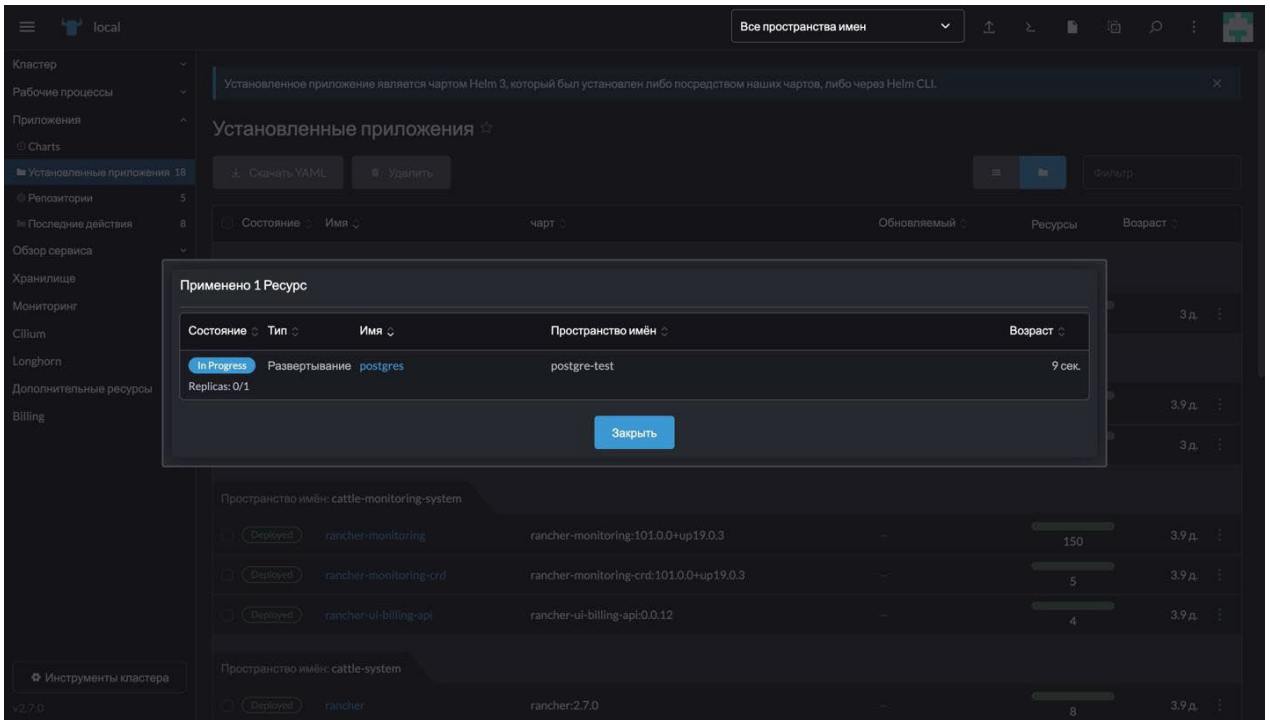


Рис. 54 Запуск установки приложения.

7.3 КОНФИГУРАЦИЯ СЕТЕВЫХ ПОЛИТИК CILLIUM

Встроенный графический интерфейс позволяет редактировать сетевые политики в визуальном формате, формируя файл конфигурации готовый для применения.

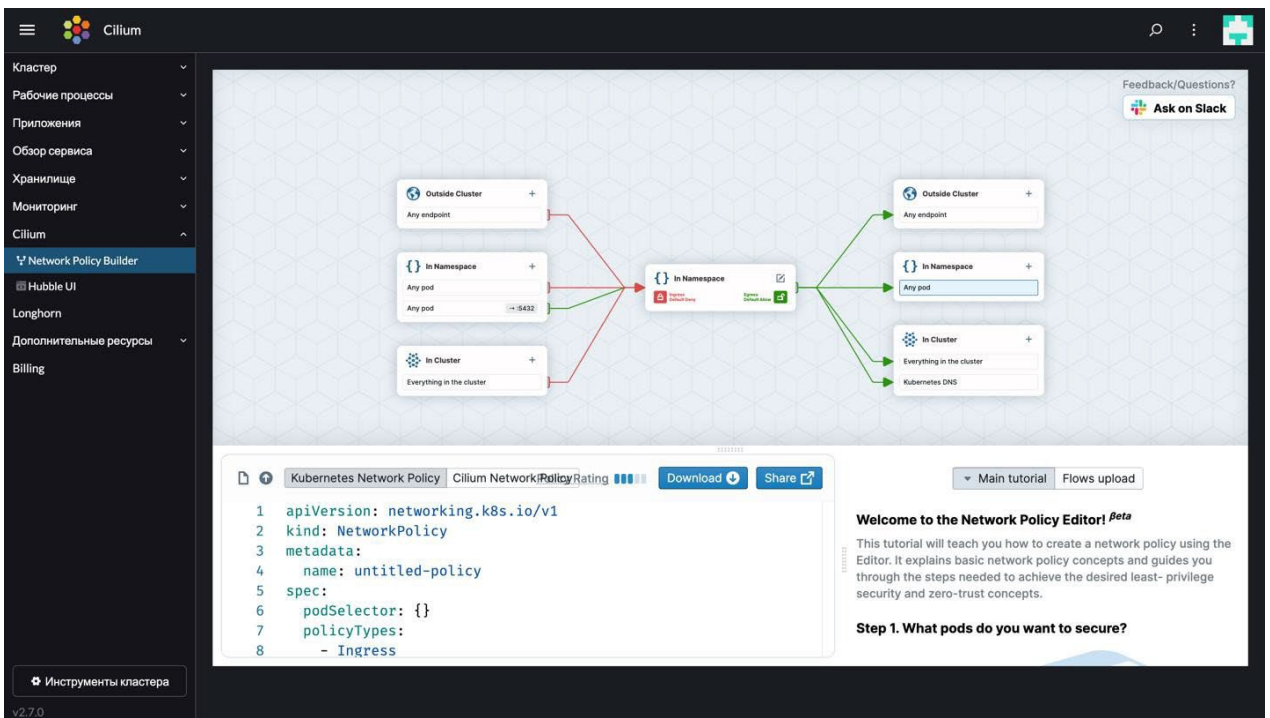


Рис. 55 Редактирование сетевых политик Cilium.

ИНФОРМАЦИЯ, НЕОБХОДИМАЯ ДЛЯ ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

В рабочей области слева внизу формируется файл конфигурации, применение которого осуществляется аналогично предыдущему пункту, через блок «Импортировать из YAML».

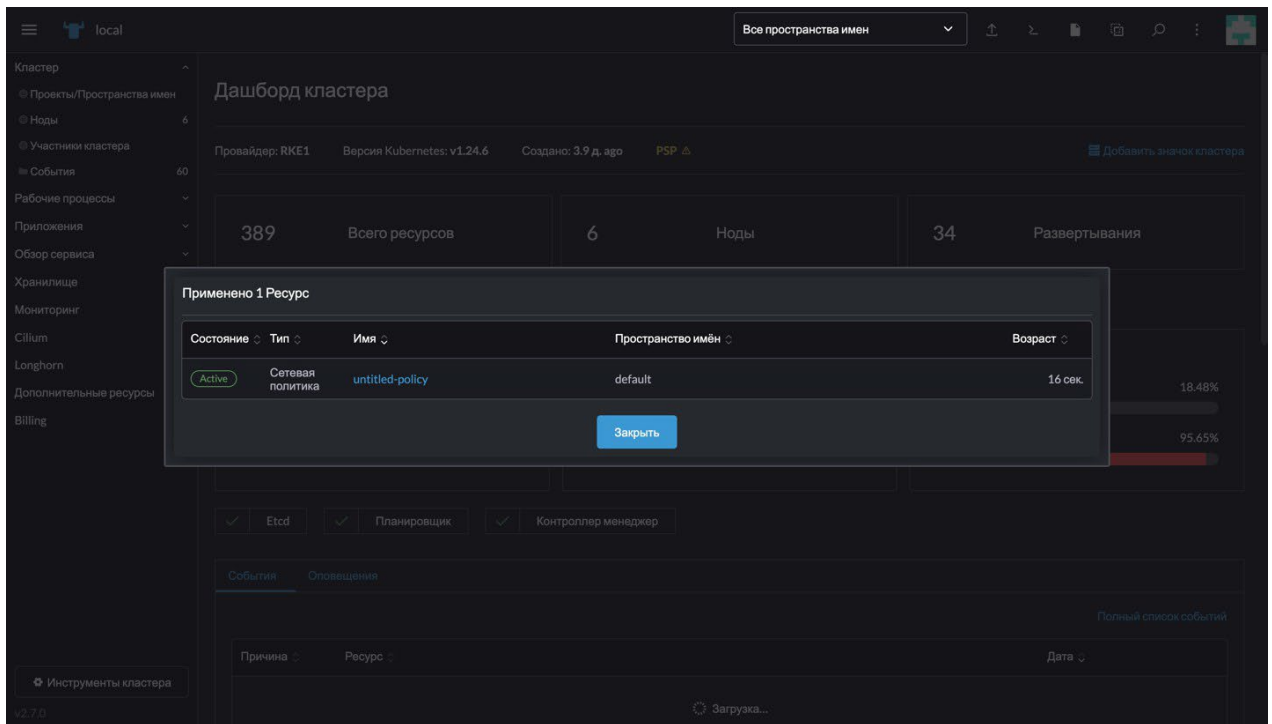


Рис. 56 Применение сетевых политик