

KASSETA

Руководство по установке

Индекс	Cassette-IG
Конфиденциальность	Публичный - L0
Ревизия	1.0
Статус	Согласован

Содержание

1. Аннотация	3
2. Термины и сокращения	4
3. Минимальные системные требования	5
4. Установка и настройка системы	6
4.1. Развертывание Kubernetes кластера	6
4.2. Развёртывание внутри Kubernetes	6
4.2.1. Состав файлов репозитория	6
4.2.2. Создание новой среды	7
4.2.3. Gitlab runner	8

1. Аннотация

Документ предназначен для технических специалистов, занимающихся установкой, настройкой и поддержкой системы для организации совместной работы с цифровым медиаконтентом, его хранения, просмотра и согласования в рамках производственных процессов. Документ рассчитан на инженеров, обладающих специальными навыками и знаниями в области инсталляции программного обеспечения.



Данный документ опубликован исключительно с целью изучения системных требований для установки продукта, а также ознакомления с последовательностью и деталями процесса установки. Реальная установка продукта производится с использованием внутренних репозиториев ООО "Цифра", доступ к которым предоставляется заказчику по запросу.

2. Термины и сокращения

Термин	Определение
Пользователь	Человек, использующий веб-интерфейс KASSETA.
Проект	Сущность, создаваемая администратором и объединяющая набор загружаемых в систему файлов в единый блок данных.
Транскодирование	Преобразование видеофайла из одного цифрового формата в другой (преобразование формата файла, видео и аудио).

3. Минимальные системные требования

Для установки KASSETA как докер образа на один сервер (без Kubernetes) необходимо наличие одного сервера. Сервер должен удовлетворять следующим требованиям:

1. Операционная система Debian.
2. Многоядерный центральный процессор с тактовой частотой каждого ядра 2 ГГц (не менее 4 ядер).
3. Объем оперативной памяти 8 ГБ.
4. 1 жесткий диск или более емкостью не менее чем по 200 ГБ.
5. Интерфейса Ethernet 100 Base-T с поддерживаемой пропускной способностью 100 Мбит/сек соответственно.
6. Свободное место для папки временных файлов /tmp - 10 ГБ.

4. Установка и настройка системы

4.1. Развертывание Kubernetes кластера

Кластер развёртывается по официальной инструкции (<https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/high-availability/>).

Для установки сервиса в имеющийся настроенный кластер Kubernetes используется процесс CI/CD, настраиваемый с помощью GitLab. Весь процесс описан в документе, который предоставляется заказчику по требованию.

Все действия возможно производить на локальной машине или на любом Ubuntu-сервере с доступом через консоль от имени любого пользователя.

4.2. Развёртывание внутри Kubernetes

Конфигурация и утилиты helm для развёртывания KASSETA находятся в репозитории, ссылка на который предоставляется по требованию заказчика.

4.2.1. Состав файлов репозитория

1. additional

Структура файлов для additional:

```
additional
  dns-values.yaml
  probes-values.yaml
```

- `dns-values.yaml` - файл настройки спецификации pods, для секции containers, конфигурация добавляется в файл `/etc/resolv.conf` в самом контейнере.
- `probes-values.yaml` - файл настройки спецификации pods, для секции containers, конфигурация liveness и readiness probe.

2. limitations

В папке лежат файлы для управления ресурсами компонент продукта. Структура файлов для секции выглядит следующим образом:

```
limitations
  default-resource.yaml
```

- `default-resource.yaml` - файл настройки спецификации pods, для секции containers, конфигурация resources, задает значения ресурсов по-умолчанию.

3. values

В папке содержатся файлы для управления параметрами helm-чартов компонент. Структура файлов выглядит следующим образом:

```
values
  _global_values
  _env.yaml.gotmpl
  ingress.yaml.gotmpl
  mps.yaml.gotmpl
  vhs-ui.yaml.gotmpl
  vhs.yaml.gotmpl
  was-configs.yaml.gotmpl
  was.yaml.gotmpl
```

- a. `global_values` - директория с глобальными настройками для всех микросервисов. Состоит из файлов `_env.yaml.gotmpl` и `ingress.yaml.gotmpl`.
 - b. `_env.yaml.gotmpl` - файл конфигурации переменных окружения, которые добавляются в переменные окружения GITLAB.
 - c. `ingress.yaml.gotmpl` - файл конфигурации ингреса.
 - d. остальные файлы являются конфигурационными файлами с переменными для конкретных микросервисов.
4. `default.yaml`
Файл конфигурации переменных по умолчанию.
 5. `helmfile.yaml`
Файл конфигурации, который содержит несколько Releases. Один Release - это один Chart, развернутый в кластер Kubernetes. Chart - это стандартный набор манифестов (`deployment`, `ingress`, `service` и т.п.) + `values`.
 6. `version.yaml`
Файл с версиями микросервисов.

4.2.2. Создание новой среды

1. Создать отдельный проект в Gitlab.
2. Настроить данный проект как submodule.
3. В проекте среды создать `helmfile-master.yaml` с содержимым:

```
---
helmfiles:
- path: < >/helmfile.yaml
  values:
    - < >/default.yaml          #
      - < >/limitation/default-resource.yaml  #
    - production.yaml          #
```

4.2.3. Gitlab runner

1. Подключите раннер, машину, на которой будут выполняться команды CD и которая будет развертывать приложение. Перейдите в раздел Settings->CI/CD->Runners settings. Если в разделе "Runners activated for this project" есть раннеры с зелёным лейблом, нажмите кнопку "Enable" рядом с ним. Если раннеров нет, добавьте по инструкции, которая находится там же.
2. Добавьте актуальный kubeconfig кластера, в который будет устанавливаться продукт, в переменную kube_config окружения CD, предварительно преобразовав его в base64.
3. Добавьте свой персональный токен в переменную GITLAB_ACCESS_TOKEN и настройте доступ (All environments).
4. Персональный токен генерируется в профиле пользователя.
5. После этого следует выполнить `init` шаг из CI/CD (CI\CD -> Pipelines). На этом этапе создастся необходимый namespace и установится продукт с конфигурацией из репозитория.

© ООО "Цифра", 2026

Документация "KASSETA. Руководство по установке" является объектом авторского права. Воспроизведение всего произведения или любой его части воспрещается без письменного разрешения правообладателя.