	ООО "ИТ ВЕКТУРА"
Версия 1.3	Инструкция по удаленному доступу к инфраструктуре системы автоматизации логистики IT Vectura Transportation Managemet System

УТВЕРЖДЕНО
Решением генерального директора ООО "ИТ ВЕКТУРА"
от «01» сентября 2022 г.
(протокол No__)

Инструкция по удаленному доступу к инфраструктуре системы автоматизации логистики IT Vectura Transportation Management System

г. Москва, 2023 г.

Версионность

Дата	Автор	Версия	Описание
01.05.2023	Источникова Е.С.	1.0	Первая версия документа
14.06.2023	Источникова Е.С.	1.1	Корректировка раздела 1. Системные требования
10.07.2023	Источникова Е.С.	1.2	Корректировка раздела 7. Дополнительно
04.03.2024	Источникова Е.С.	1.3	Обновление разделов 2. Удаленный доступ к инфраструктуре с развернутым экземпляром, 6. Дополнительная инфраструктура в Yandex, 8. Контакты технических специалистов

Оглавление

1. Системные требования

Для доступа к инфраструктуре с программным обеспечением “IT Vectura” необходимо ПО для подключения ssh:

Для доступа из под windows рекомендуется установить Putty.

https://en.wikipedia.org/wiki/PuTTY;

Для доступа из под Linux рекомендуется использовать OpenSSH:

https://en.wikipedia.org/wiki/OpenSSH;

2. Удаленный доступ к инфраструктуре с развернутым экземпляром

Шаги для доступа к инфраструктуре:

1. Выдайте ваш публичный SSH ключ администратору

2. Администратор выдаст доступ вашему ключу к удаленному хосту ubuntu@stage2.itvectura.com

3. Откройте терминал и выполните команду ssh для подключения к хосту.

3. Описание используемых программ на хосте

3.1 Операционная система

На хосте установлена операционная система Ubuntu 22.04.2 LTS, которая является долгосрочной поддерживаемой версией ОС Ubuntu и обеспечивает обновления безопасности и исправления ошибок в течение 5 лет.

3.2 Kubernetes

Kubernetes - это платформа управления контейнеризованными приложениями, разработанная Google и сейчас поддерживаемая общиной. Kubernetes предоставляет инструменты для автоматического развертывания, масштабирования и управления контейнеризованными приложениями, а также для управления ресурсами, сетью и хранилищем данных.

Kubernetes использует концепцию подов (Pods), которые являются наименьшей единицей развертывания приложений и содержат один или несколько контейнеров. Поды объединяются в службы (Services), которые предоставляют стабильное имя и IP-адрес для доступа к приложению, независимо от того, на какой физической машине оно запущено. Kubernetes также позволяет масштабировать приложение горизонтально, добавляя или удаляя экземпляры подов в зависимости от нагрузки.Он предоставляет возможность управления конфигурацией приложений и автоматическое восстановление после сбоев. Он также позволяет управлять сетевыми настройками и хранилищем данных для приложения. На хосте установлен пакет MicroK8s, который представляет собой минимальную, быструю и легковесную версию Kubernetes. MicroK8s поддерживает различные плагины, в том числе cert-manager, который упрощает управление и автоматизацию сертификатов TLS (Transport Layer Security) для приложений, работающих на Kubernetes.

Установленные плагины Microk8s:

Название	Описание
cert-manager	Управление сертификатами TLS. Интегрируется с Kubernetes для автоматического выдачи и управления сертификатами.

dns	CoreDNS – сервер DNS для Kubernetes, который упрощает доступ к сервисам и приложениям в кластере
ha-cluster	Предоставляет настройку высокой доступности для кластера на текущем узле
helm	Пакетный менеджер для Kubernetes
helm3	Версия 3 пакетного менеджера Helm для Kubernetes
hostpath-storage	Плагин предоставляет хранилище для Kubernetes в виде директории на хост машине
ingress	Предоставляет контроллер Ingress для внешнего доступа к сервисам в Kubernetes
storage	Устаревший плагин, является алиасом для hostpath-storage.

Эти плагины предоставляют дополнительные функциональные возможности для кластера Kubernetes, такие как управление сертификатами, хранилищем, контроллеры входа и маршрутизации, пакетный менеджер для Kubernetes и т.д.

Расположение папок баз данных/приложений на хосте:

- PostgreSQL: /mnt/postgres/
- PostgreSQL: /mnt/itv-data/

При развертывании MicroK8s был создан Центр сертификации (Certificate Authority), подписанный сертификат сервера и ключевой файл учетной записи службы. Эти файлы хранятся в каталоге /var/snap/microk8s/current/certs/. Kubelet и API-сервер используют тот же самый Центр сертификации, и поэтому подписанный сертификат сервера используется API-сервером для аутентификации с kubelet (--kubelet-client-certificate). Сейчас серверные сертификаты выданы для следующих адресов:

stage2.itvectura.com

studio.stage2.itvectura.com

Для доступа к API-серверу через Интернет и реальное доменное имя, в файл /var/snap/microk8s/current/certs/csr.conf.template добавлены данные строки: [alt_names]

DNS.0 = kubernetes

DNS.1 = kubernetes.default

DNS.2 = kubernetes.default.svc

DNS.3 = kubernetes.default.svc.cluster

DNS.4 = kubernetes.default.svc.cluster.local

DNS.5 = stage3.itvectura.com

IP.0 = 127.0.0.1

IP.1 = 10.152.183.1

IP.2 = 89.208.209.171

3.3 Конфигурационные файлы Kubernetes

Файлы конфигурации Kubernetes - это конфигурационные файлы, которые используются клиентом Kubernetes для подключения к кластеру и выполнения операций в нем. Они хранятся в каталоге /root/.kube

Файлы в каталоге .kube включают:

Название	Описание
config	файл конфигурации Kubectl, используемый для определения контекстов, кластеров и пользовательских учетных записей.
kubelet.conf	файл конфигурации Kubelet, используемый для определения параметров запуска узла Kubelet.
controller manager.conf	файл конфигурации контроллера менеджера, используемый для настройки параметров контроллера-менеджера Kubernetes.
scheduler.conf	файл конфигурации планировщика, используемый для настройки параметров планировщика Kubernetes.

4. Описание используемых программ на хосте. Установка Kubernetes CLI (kubectl) на различные операционные системы для получения прямого доступа к кластерам (опционально, можно заходить с хоста вместо этого)

Установка Kubernetes CLI (kubectl) может отличаться в зависимости от операционной системы, на которой вы работаете. Ниже приведены инструкции для установки kubectl Инструкция по установке:

https://kubernetes.io/ru/docs/tasks/tools/install-kubectl/

Для добавления кластера в настройки kubectl используйте следующие шаги:

1. Попросите администратора предоставить вам файл конфигурации kubeconfig.

2. Сохраните конфигурационный файл в ~/.kube/config на вашем компьютере. Если файл уже существует, сначала сохраните его копию, например, так:

$ cp ~/.kube/config ~/.kube/config-backup

3. Затем скопируйте конфигурационный файл в ~/.kube/config: $ cp /path/to/kubeconfig ~/.kube/config

Теперь вы готовы использовать kubectl для доступа к кластеру Kubernetes!

5. Подробное описание элементов нашего кластера которые используются в Kubernetes

В Kubernetes (k8s) используется множество различных элементов, которые взаимодействуют друг с другом для управления и масштабирования контейнерных приложений.

5.1 Поды (Pods)

Поды (Pods) - это минимальная единица развертывания в Kubernetes, включающая один или несколько контейнеров. Чтобы увидеть все поды, выполните команду kubectl get pods -- all-namespaces.

Название (Вместо многоточия динамически сгенерированный присвоенный индикатор)	Описание
postgres-statefulset-0	Postgresql база данных
cert-manager-...	Управление сертификатами в Kubernetes с помощью Let's Encrypt
cert-manager-cainjector-...	Контроллер сертификатов для cert manager
camunda-deployment-...	BPM-платформа Camunda CE 7.0
camunda-tasks-v1-deployment-...	Сервис, связанный с BPM платформой Camunda CE 7.0. Осуществляет взаимодействие с другими сервисами через REST API
frontend-v3-deployment-...	Фронтенд-приложение на Node.js
nginx-ingress-microk8s controller-...	Контроллер для управления входящим трафиком
cert-manager-webhook-...	Webhook для управления сертификатами с помощью cert manager
coredns-...	Система DNS-серверов в Kubernetes
k8s-agent-stage2-gitlab agent-...	Агент GitLab для запуска сборок и тестирования

Название (Вместо многоточия динамически сгенерированный присвоенный индикатор)	Описание
metrics-server-...	Кластерный агрегатор данных об использовании ресурсов
kubernetes-dashboard-...	Универсальный веб-интерфейс для кластеров Kubernetes
ovn-node-...	Подсистема сетевой политики и маршрутизации
ovn-kube-controllers-...	Контроллер ovn для управления политикой сети в Kubernetes
backend-functions-v1-deployment-...	Деплоймент бэкенд-функций версии 1.0
chatbot-v1-deployment-...	Деплоймент чатбота версии 1.0
dashboard-metrics-scraper-...	Сборщик метрик для панели управления
data-import-from-1c-v1-deployment-...	Под модуля импорта данных из системы 1С версии 1.0
data-import-from-etran-v1-deployment-...	Под модуля импорта данных из системы ETRAN версии 1.0
data-import-mapper-v1-deployment-...	Под модуля сопоставления данных версии 1.0
dbmate-migrations-...	Миграции базы данных с использованием инструмента dbmate
itv-supabase-auth-...	Аутентификация и управление пользователями в Supabase
itv-supabase-db-...	База данных в Supabase
itv-supabase-kong-...	Управление API и маршрутизацией в Supabase
itv-supabase-meta-...	Метаданные и управление внутренними настройками в Supabase
itv-supabase-realtime-...	Обработка реального времени в Supabase

Название (Вместо многоточия динамически сгенерированный присвоенный индикатор)	Описание
itv-supabase-rest-...	REST API и обработка запросов в Supabase
itv-supabase-storage-...	Управление хранилищем и файлами в Supabase
itv-supabase-studio-...	Визуальная среда разработки и управления проектами в Supabase

5.2 Сервисы (Services)

Сервисы (Services) - это механизм для обеспечения доступности и маршрутизации трафика к подам. Чтобы увидеть все сервисы, выполните команду kubectl get services --all-namespaces.

Название (Вместо многоточия динамически сгенерированный присвоенный индикатор)	Тип	Порт	Описание
kubernetes	ClusterIP	443/TCP	Встроенный сервис, используемый для управления кластером Kubernetes.
camunda-service	NodePort	8080:31754/TCP	Служба Camunda CE 7, используемая для управления процессами
frontend-v3-service	NodePort	80:30828/TCP	Служба, предоставляющая пользовательский интерфейс для взаимодействия с Node.js приложением
kube-dns	ClusterIP	53/UDP,53/TCP, 9153/TCP	Служба DNS для кластера Kubernetes, редоставляющая имя и IP-адрес для всех объектов в кластере

Название (Вместо многоточия динамически сгенерированный присвоенный индикатор)	Тип	Порт	Описание
postgres-service	NodePort	5432:30991/TCP	Служба базы данных PostgreSQL, используемая для хранения данных приложения
cert-manager	ClusterIP	9402/TCP	Служба cert-manager, используемая для автоматического получения и обновления SSL сертификатов для приложения
cert-manager webhook	ClusterIP	443/TCP	Служба cert-manager, используемая для взаимодействия с внешними сервисами, такими как Let's Encrypt
backend-functions v1-service	NodePort	9000:31325/TCP	Сервис бэкенд функций версии 1.0
dashboard-metrics scraper	ClusterIP	8000/TCP	Сервис сборщика метрик для панели управления
itv-supabase-auth	ClusterIP	9999/TCP	Сервис аутентификации и управления пользователями в Supabase
itv-supabase-db	ClusterIP	5432/TCP	Сервис базы данных в Supabase
itv-supabase-kong	ClusterIP	8000/TCP	Сервис управления API и маршрутизацией в Supabase
itv-supabase-meta	ClusterIP	8080/TCP	Сервис метаданных и управления внутренними настройками в Supabase

Название (Вместо многоточия динамически сгенерированный присвоенный индикатор)	Тип	Порт	Описание
itv-supabase-rest	ClusterIP	3000/TCP	Сервис REST API и обработки запросов в Supabase
itv-supabase-storage	ClusterIP	5000/TCP	Сервис управления хранилищем и файлами в Supabase
itv-supabase-studio	ClusterIP	3000/TCP	Сервис визуальной среды разработки и управления проектами в Supabase
realtime-dev	ClusterIP	4000/TCP	Сервис для обработки данных в реальном времени
kubernetes dashboard	ClusterIP	443:8443/TCP	Сервис панели управления Kubernetes
metrics-server	ClusterIP	443:https/TCP	Сервис метрик для Kubernetes

5.3 Деплойменты (Deployments)

Деплойменты (Deployments) - это объекты, которые управляют созданием и обновлением репликасетов и подов. Можно ознакомиться с помощью данной команды: kubectl get deployments --all-namespaces.

Название (Вместо многоточия динамически сгенерированный присвоенный индикатор)	Описание
gitlab-agent-k8s-agent-stage	Развертывание GitLab Runner агента, который запускает GitLab CI/CD задания.
coredns	Развертывание CoreDNS, сервера DNS, который заменяет kube-dns в Kubernetes.

Название (Вместо многоточия динамически сгенерированный присвоенный индикатор)	Описание
cert-manager-cainjector	Развертывание CA инжектора для Cert-Manager, который генерирует сертификаты для Kubernetes ресурсов.
cert-manager	Развертывание Cert-Manager, который обеспечивает автоматическое управление сертификатами.
cert-manager-webhook	Развертывание Cert-Manager вебхука, который обрабатывает запросы на подпись сертификатов.
camunda-deployment	Развертывание Camunda CE 7 BPM, открытой платформы управления бизнес процессами, которая позволяет моделировать, автоматизировать и оптимизировать бизнес процессы.
camunda-tasks-v1-deployment	Развертывание Node.js приложения, которое взаимодействует с Camunda BPM через REST API, чтобы управлять задачами бизнес процессов.
frontend-v3-deployment	Деплоймент фронтенда Node.js, который предоставляет веб-интерфейс
ovn-kube-controllers	Обеспечивает управление и настройку сетевой политики Kubernetes с помощью ovn
backend-functions-v1-deployment	Развертывание бэкенд функций версии 1.0
data-import-from-1c-v1-deployment	Развертывание модуля импорта данных из системы 1С версии 1.0
data-import-from-etran-v1-deployment	Развертывание модуля импорта данных из системы ETRAN версии 1.0
data-import-mapper-v1-deployment	Развертывание модуля сопоставления данных версии 1.0

Название (Вместо многоточия динамически сгенерированный присвоенный индикатор)	Описание
kubernetes-dashboard	Развертывание панели управления Kubernetes, которая обеспечивает визуальный интерфейс для управления кластером
dashboard-metrics-scraper	Развертывание сборщика метрик для панели управления
metrics-server	Развертывание сервера метрик для Kubernetes, обеспечивающего сбор и предоставление метрик
chatbot-v1-deployment	Развертывание чатбота версии 1.0
itv-supabase-db	Развертывание базы данных в Supabase
itv-supabase-rest	Развертывание REST API и обработки запросов в Supabase
itv-supabase-kong	Развертывание управления API и маршрутизации в Supabase
itv-supabase-meta	Развертывание метаданных и управления внутренними настройками в Supabase
itv-supabase-realtime	Развертывание обработки реального времени в Supabase
itv-supabase-studio	Развертывание визуальной среды разработки и управления проектами в Supabase
itv-supabase-auth	Развертывание аутентификации и управления пользователями в Supabase
itv-supabase-storage	Развертывание управления хранилищем и файлами в Supabase

5.4 Конфигурационные карты (ConfigMaps) и Секреты (Secrets)

Конфигурационные карты (ConfigMaps) и Секреты (Secrets) - это объекты, которые используются для хранения конфигурационных данных и секретной информации соответственно. Со списком ConfigMaps и Secrets можно ознакомится с помощью данной команды:

kubectl get configmaps --all-namespaces.

kubectl get secrets --all-namespaces.

Название (Вместо многоточия динамически сгенерированный присвоенный индикатор)	Описание
extension-apiserver authentication	Расширение аутентификации API сервера
ovn-config	Конфигурация сетевого решения ovn
local-registry-hosting	Локальный реестр для хостинга образов
kube-root-ca.crt	Корневой сертификат для узлов кластера Kubernetes
postgres-config	Конфигурация СУБД PostgreSQL
k8s-agent-stage2-gitlab agent-token	Агент GitLab для связи с Kubernetes на этапе Stage
coredns	Деплоймент CoreDNS для DNS-сервиса в кластере
nginx-load-balancer microk8s-conf	Конфигурация Nginx для балансировки нагрузки в кластере MicroK8s
nginx-ingress-tcp microk8s-conf	Конфигурация Nginx для управления трафиком в кластере MicroK8s
cert-manager-webhook	Вебхук для обновления сертификатов в кластере
ingress-controller-leader	Лидер-кандидат для Ingress контроллера
itv-camunda-tasks	ConfigMap с конфигурацией для задач в Camunda.
itv-supabase-auth	ConfigMap с конфигурацией для аутентификации и управления пользователями в Supabase.

Название (Вместо многоточия динамически сгенерированный присвоенный индикатор)	Описание
itv-supabase-kong	ConfigMap с конфигурацией для управления API и маршрутизации в Supabase.
itv-supabase-meta	ConfigMap с метаданными и конфигурацией внутренних настроек в Supabase.
itv-supabase-realtime	ConfigMap с конфигурацией для обработки данных в реальном времени в Supabase.
itv-supabase-rest	ConfigMap с конфигурацией для REST API и обработки запросов в Supabase.
itv-supabase-storage	ConfigMap с конфигурацией для управления хранилищем и файлами в Supabase.
itv-supabase-studio	ConfigMap с конфигурацией для визуальной среды разработки и управления проектами в Supabase.
itv-backend-functions	ConfigMap с конфигурацией для бэкенд-функций.
itv-supabase-db	Секрет с данными для подключения к базе данных в Supabase.
itv-supabase-jwt	Секрет с JSON Web Token (JWT) для аутентификации в Supabase.
itv-supabase-smtp	Секрет с данными для настройки и подключения к SMTP-серверу для отправки электронной почты в Supabase.
itv-auth-studio-secret	Секрет с секретным ключом для аутентификации в студии разработки (Auth Studio).
docker-registry-secret	Секрет с учетными данными для подключения к реестру Docker.
itv-camunda-tasks	Секрет с данными для задач в Camunda.

5.5 Хранилища данных (Persistent Volumes) и Запросы на хранилища (Persistent Volume Claims)

Хранилища данных (Persistent Volumes) и Запросы на хранилища (Persistent Volume Claims) - это объекты, которые используются для управления доступом к постоянному хранилищу данных в Kubernetes. Со списком ConfigMaps можно ознакомится с помощью данной команды:

kubectl get PersistentVolumeClaims --all-namespaces.

Название	Volume	Размер	Access Modes	Storage Class
postgres-claim	postgres volume	8Gi	RWO	microk8s hostpath
itv-supabase db-pvc	itv-supabase db-pv	50Gi	RWO	standard

5.6 Ингрессы (Ingresses)

Ингрессы (Ingresses) - это объекты, которые используются для управления доступом к приложениям извне кластера Kubernetes. Со списком Ingresses можно ознакомится с помощью данной команды:

kubectl get Ingresses --all-namespaces.

Название	Хост	ip	ports
frontend-v3-ingress	stage2.itvectura.com	127.0.0.1	80, 443
camunda ingress	stage2.itvectura.com	127.0.0.1	80
itv-supabase kong	stage2.itvectura.com	127.0.0.1	80, 443
itv-supabase studio	stage2.itvectura.com	127.0.0.1	80, 443
backend functions-v1-ingress	stage2.itvectura.com	127.0.0.1	80, 443

6. Дополнительная инфраструктура в Yandex Cloud

Yandex Cloud (YC) - это облачная платформа, предоставляемая Яндексом. Она позволяет создавать, запускать и масштабировать приложения и сервисы в облаке, используя виртуальные машины, контейнеры и серверы функций. Для получения доступа обратитесь к администратору и предоставьте учетную запись Яндекс.

6.1 Реестр контейнеров (Registry)

Реестр контейнеров (Registry) - сервис, который позволяет хранить и управлять Docker образами. Он может использоваться для создания своих собственных образов или для получения доступа к общедоступным образам из различных репозиториев.

6.2 Object Storage

Object Storage: Сервис для хранения и управления объектами, который позволяет загружать, скачивать и удалять файлы и директории из облака. YC предоставляет различные уровни доступности и хранения данных, а также инструменты для управления доступом.

6.3 DNS-сервер (DNS)

DNS-сервер (DNS) - сервис, который позволяет переводить доменные имена в IP-адреса и наоборот. Он используется для определения расположения ресурсов в сети и облегчения доступа к ним.

6.4 Gitlab Manage Service

Gitlab Manage Service - это процесс автоматизации сборки, тестирования и развертывания приложений. Он используется для создания, тестирования и запуска приложений в автоматическом режиме.

Связь с реестром контейнеров (Registry) заключается в том, что после сборки и тестирования приложения в pipeline, его Docker-образ может быть загружен в реестр контейнеров для последующего развертывания на серверах.

7. Дополнительно

7.1 Бэкапы

В подах Camunda CE 7 задействованы два скрипта, которые выполняют команды pg_dump для создания резервных копий PostgreSQL.

Для хранения этих резервных копий используется сервис Object Storage, предоставляемый Яндекс.Облаком. В конфигурации приложения задаются параметры доступа к сервису Object Storage, такие как ключ доступа, секретный ключ, название бакета и т.д. Скрипты выполняют команды для сохранения файлов резервных копий в указанный бакет Object Storage.

После сохранения резервных копий в Object Storage, их можно использовать для восстановления данных в случае потери или повреждения исходных данных в PostgreSQL.

8. Контакты технических специалистов

konstantin.trunin@itvectura.com – Константин Трунин

konstantin.karpov@itvectura.com – Константин Карпов

aleksandr.zaburdaev@itvectura.com – Александр Забурдаев