Инструкция по установке информационной системы

RT.MDM

Москва 2025

# 1. Общие сведения

Наименование системы: RT.MDM.

Документ содержит информацию об установке, настройке и обновлении RT.MDM.

## 1.1 Термины, определения и сокращения

В настоящем документе использованы и определены следующие термины и сокращения:

| **Термин/ Сокращение** | **Определение** |
| --- | --- |
| БД | База данных. |
| ПО | Программное обеспечение. |
| Docker | ПО для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации, контейнеризатор приложений. |
| Docker-compose | Инструментальное средство, входящее в состав Docker. Предназначено для решения задач, связанных с развёртыванием проектов. |
| Docker-образ(Docker-image) | Шаблон для создания docker-контейнеров. Представляет собой исполняемый пакет, содержащий все необходимое для запуска приложения: код, среду выполнения, библиотеки, переменные окружения и файлы конфигурации. |
| Docker-контейнер (Docker-container) | Автономный исполняемый пакет программного обеспечения, созданный на основе docker-образа. |
| Docker-реестр(Docker- registry) | Хранилище docker-образов. |

# 2. ТРЕБОВАНИЯ RT.MDM

В этом разделе описаны требования к RT.MDM и программному обеспечению.

## 2.1 Аппаратное обеспечение для работы RT.MDM

Для функционирования RT.MDM должны быть использованы следующие виды аппаратного обеспечения (Таблица 1):

Таблица 1 — Виды аппаратного обеспечения

| **Тип системы** | **CPU** | **Объем RAM, ГБ** | **Объем HDD, ГБ** | **Количество серверов** |
| --- | --- | --- | --- | --- |
| Сервер БД | 16 | 32-128 (в зависимости от объёма данных) | 150 | 1 |
| Сервер приложений | 16 | 32 | 100 | 1 |

## 2.2 Состав системного ПО RT.MDM

RT.MDM использует следующие программные компоненты:

1. Операционная система RED OS 7.3.
2. ПО Docker, Docker compose.
3. Сервер баз данных PostgreSQL 12.6 или выше.
4. Расширение PostGIS для сервера баз данных PostgreSQL.
5. Расширение btree\_gin для PostgreSQL.
6. Расширение btree\_gist для PostgreSQL.
7. Расширение dblink для PostgreSQL.
8. Расширение pg\_stat\_statements для PostgreSQL.
9. Расширение pg\_trgm для PostgreSQL.
10. Расширение pgstattuple для PostgreSQL.
11. Расширение plpgsql для PostgreSQL.
12. Расширение plpython3u для PostgreSQL.
13. Расширение postgres\_fdw для PostgreSQL.
14. Расширение tds\_fdw для PostgreSQL.
15. Расширение uuid-ossp для PostgreSQL.
16. Система поставляется в виде набора микро-сервисов, обменивающихся между собой информацией. Каждый из микро-сервисов обернут в отдельный docker-образ. Все docker-образы расположены в хранилище docker-registry.

Запуск приложения включает в себя развертывание нескольких docker-контейнеров, которые основаны на базовых docker-образах. Конфигурация для каждого сервиса, включая сетевые настройки и зависимости, задается в файле docker-compose.yml. Значения переменных окружения, необходимых для настройки контейнеров, указываются в файле .env.

# 3. Установка программных компонентов

Установка программных компонентов п.п. 1 - 4 из раздела «Состав системного ПО RT.MDM» является стандартной и в данную инструкцию не входит.

# 5. Настройка базы данных

1. Создайте роль БД с произвольным именем. Временно дайте этой роли права суперпользователя.
2. Создайте базу с произвольным именем и сделайте владельцем этой базы созданную в п.1 роль.
3. Скачайте файл дампа базы данных rt\_mdm\_dump из репозитория в произвольную папку.
4. Восстановите базу из дампа: pg\_restore -h localhost -p 5432 -U <имя созданной роли> --verbose --dbname= <имя созданной базы> <путь к дампу> &>> <имя файла для логов>
5. Проверьте, есть ли ошибки в файле для логов (на ошибки вида «функция pg\_catalog.\* не существует» не обращать внимание.
6. Уберите права суперпользователя у созданной в п.1 роли.

# 6. Установка и запуск системы RT.MDM

1. Скачайте из репозитория файлы архивов docker-образов.
2. Загрузите docker-образы в хранилище docker-registry при помощи команды docker load.
3. Проверьте наличие образов при помощи команды docker images.
4. Скачайте из репозитория файлы *docker-compose.yml* и .env в  папку opt/services/rt-mdm (Если название папки отличается от /opt/services/rt-mdm, нужно будет в файле .env заменить значение переменной VOLUME\_PATH).
5. В файле .env необходимо задать значения для переменных окружения. Обязательными являются переменные:
- DB\_MDM\_URL
- DB\_MDM\_PASSWORD
- VOLUME\_PATH (если название папки отличается от /opt/services/rt-mdm)
Значения остальных переменных можно заменить при необходимости позже.
6. Выполните команды
sudo chmod 777  /opt/services/rt-mdm/elasticsearch/data
sudo chmod 777  /opt/services/rt-mdm/elasticsearch/logs
Если название папки отличается от /opt/services/rt-mdm, нужно в командах исправить.
7. Перейдите в папку, содержащую файл docker-compose.yml  и выполните команду: docker-compose up –d.
8. Запустите команду docker ps чтобы убедиться, что все сервисы поднялись.
9. Скачайте из репозитория файл template.xlsm и скопируйте его в папку  /opt/services/rt-mdm/template (template.xlsm  - это шаблон Excel, необходимый для работы со справочником в Excel).
10. Зайдите в интерфейс системы RT.MDM. Для этого нужно в строке браузера набрать ip-адрес сервера, на котором запущены сервисы, двоеточие и порт 7001.
11. На появившейся форме авторизации ввести в полях «Электронная почта» и «Пароль» значение admin (admin - это пользователь с административными правами, созданный по умолчанию). Изменить пароль пользователя admin, а также добавить новых пользователей можно в разделе «Права пользователей».

# 7. Установка обновлений системы RT.MDM

1. Скачайте из репозитория файл «[*скрипт для накатки изменений в БД.txt*](https://repo.data.rt.ru/repository/rt.mdm/rt_mdm/%D1%81%D0%BA%D1%80%D0%B8%D0%BF%D1%82%20%D0%B4%D0%BB%D1%8F%20%D0%BD%D0%B0%D0%BA%D0%B0%D1%82%D0%BA%D0%B8%20%D0%B8%D0%B7%D0%BC%D0%B5%D0%BD%D0%B5%D0%BD%D0%B8%D0%B9%20%20%D0%B2%20%D0%91%D0%94.txt)*»* и выполните его, выбрав базу mdm (Например, в программе DBeaver).
2. . Скачайте из репозитория файлы архивов docker-образов со свежей датой.
3. Загрузите docker-образы в хранилище docker-registry при помощи команды docker load.
4. Если в репозитории  файл *docker-compose.yml* изменился (добавились новые переменные окружения или настройки), необходимо перенести их в свой файл *docker-compose.yml* либо заменить его, а в файл .env добавить недостающие переменные окружения.
5. Если в репозитории файл *template.xlsm* изменился, скачайте его в папку  /opt/services/rt-mdm/template, либо загрузите его через интерфейс в разделе «Администрирование»-> «Документация».
6. Перейдите в папку, содержащую файл *docker-compose.yml*  и выполните команду: docker-compose up –d.
7. Выполните команду docker ps чтобы убедиться, что все сервисы поднялись.