

**Техническая архитектура
Системы управления API
«Гравити Апи Менеджмент Рус»**

Москва 2023

Содержание

1. Назначение документа	3
2. Описание архитектуры Гравити Апи Менеджмент Рус	3
2.1. Сетевая архитектура.....	3
2.1.1. Сетевые экраны	3
2.2. Верхнеуровневое описание компонентов.....	4
2.3. Функциональная схема	5
2.3.1. Компонент - API Gateway	6
2.3.2. Компонент - Management API.....	7
2.3.3. Компонент - Management UI	8
2.3.4. Компонент - Portal UI	9
2.3.5. Компонент - PostgreSQL.....	10
2.3.6. Компонент - Opensearch	11
2.3.7. Компонент - API Gateway	12
2.4. Отказоустойчивость и масштабирование.....	13
2.4.1. База данных	13
2.4.2. Серверы приложений.....	13
2.5. Программные интерфейсы.....	14

1. Назначение документа

Данный документ в первую очередь ориентирован на команду технических специалистов и архитекторов Заказчика. В документе описывается техническая архитектура Системы управления API «Гравити Апи Менеджмент Рус»

2. Описание архитектуры Гравити Апи Менеджмент Рус

2.1. Сетевая архитектура

На диаграмме «Сетевая архитектура» представлена общая схема расположения и взаимодействия компонентов ПО Гравити Апи Менеджмент Рус. Детальное описание каждого компонента приведено в последующих секциях документа.

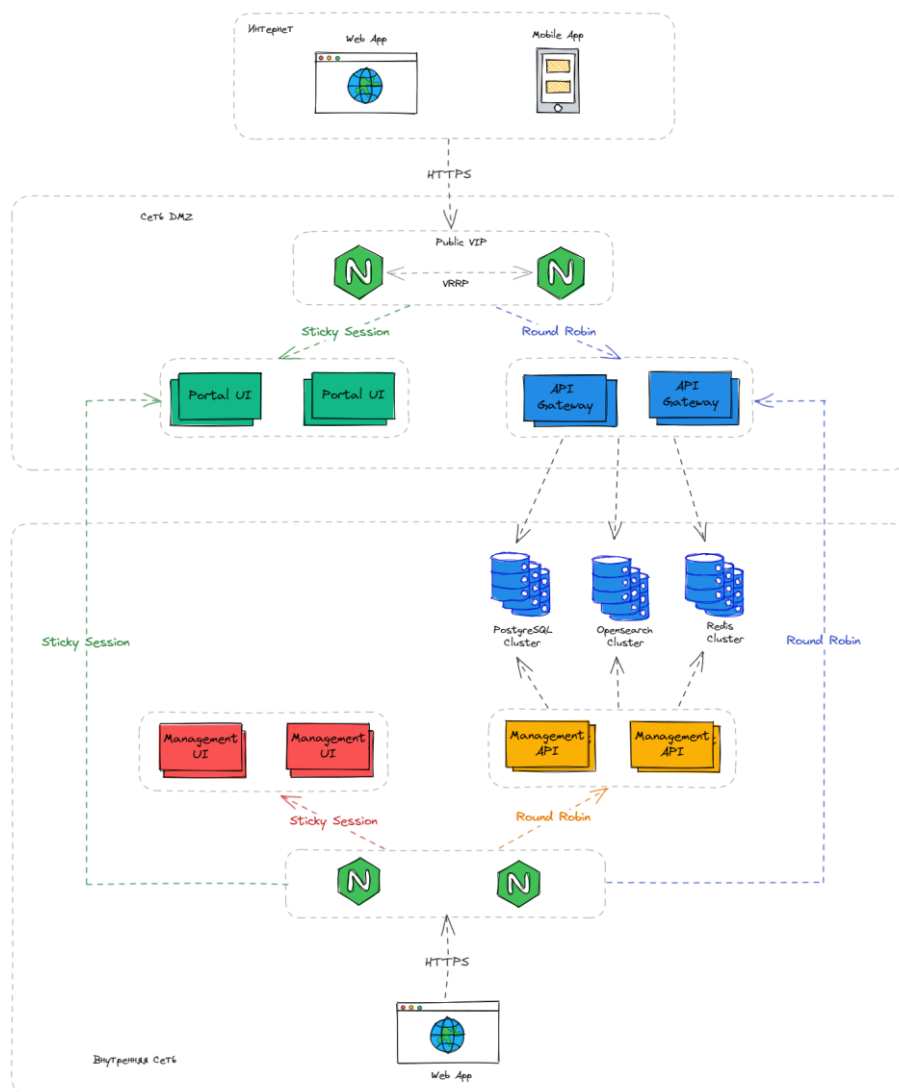


Рисунок 1 - Сетевая архитектура Гравити Апи Менеджмент Рус

2.1.1. Сетевые экраны

Сетевые экраны (firewall) находятся в зоне ответственности Заказчика; предполагается, что все необходимые порты будут открыты. Точные номера портов и IP адреса серверов каждого окружения могут меняться, поэтому не фиксируются в данной спецификации и будут уточняться в рабочем порядке в процессе конфигурации каждого окружения.

2.2. Верхнеуровневое описание компонентов

Система «Гравити Апи Менеджмент Рус» состоит из кластера баз данных PostgreSQL, кластера Opensearch, кластера Redis и кластера серверов приложений (API Gateway, Management API, Management UI, Portal UI). Все сервера в текущей архитектуре – виртуальные.

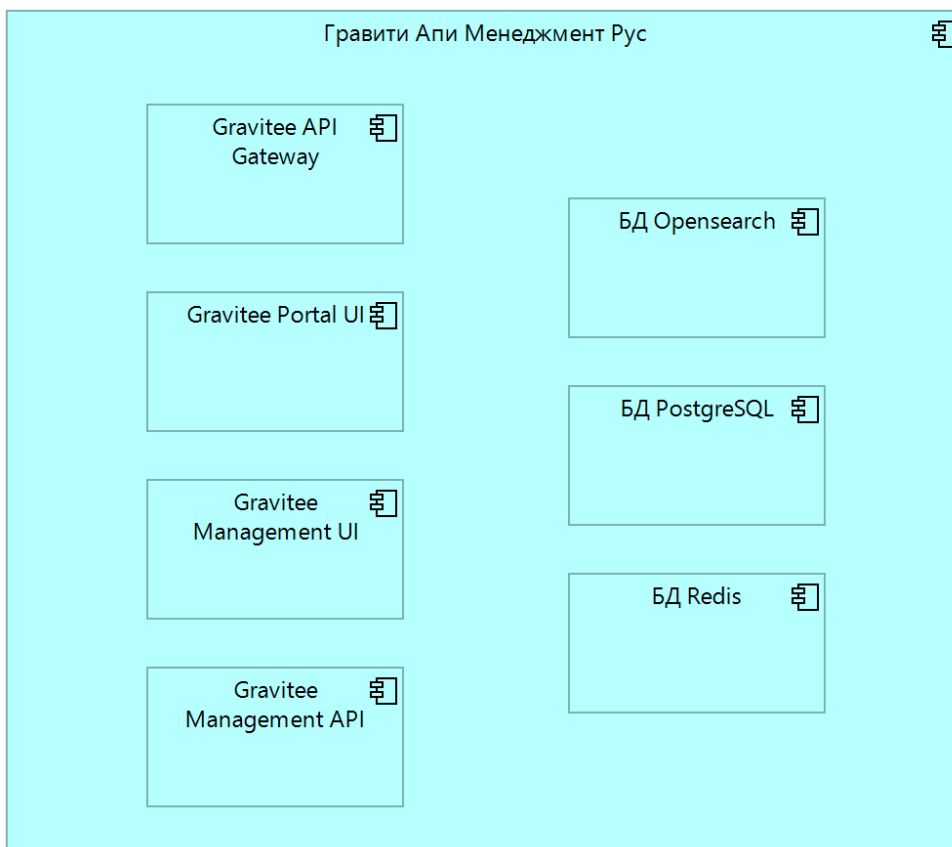


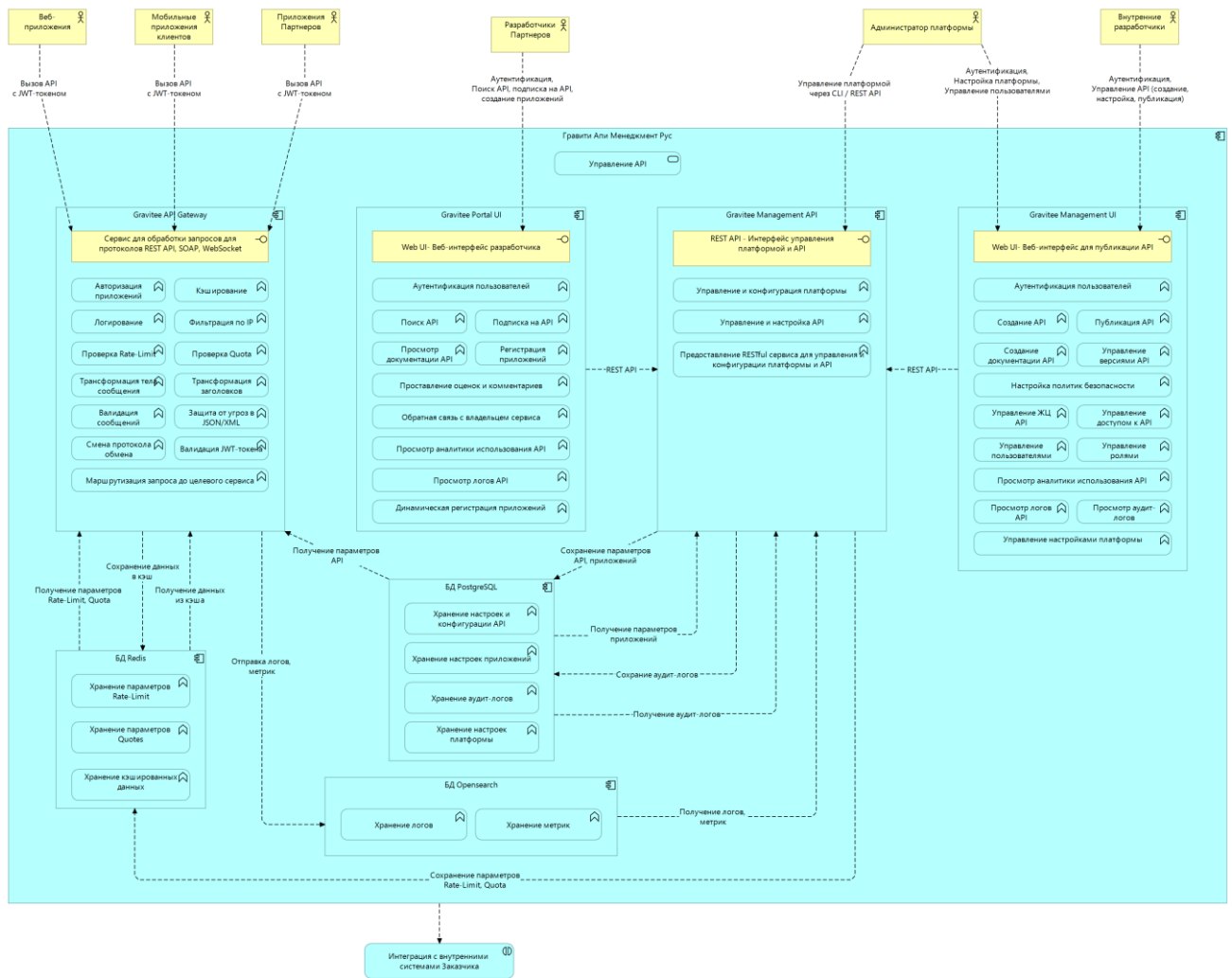
Рисунок 2 - Состав системы Гравити Апи Менеджмент Рус

Хранение информации реализуется разными компонентами, в зависимости от типа хранимых данных.

Таблица 1 - Информация по хранению разных типов данных

Тип хранилища / Тип данных	PostgreSQL	Opensearch	Redis
Management Хранение настроек платформы, конфигурации API, настройки приложений, аудит-логи	X		
Analytics Хранение данных для аналитики/статистики (метрики, логи)		X	
Rate-Limit Хранение настроек ограничения трафика (Rate-Limit, Quota) по каждому API, кэширование данных			X

2.3. Функциональная схема



2.3.1. Компонент - API Gateway

2.3.1.1. Назначение

API Gateway – это основной компонент Системы API Management, который обеспечивает выполнение политик (правила) безопасности и управляет преобразованием входящих и исходящих запросов, проходящих через API Gateway.

2.3.1.2. Функциональные возможности

Основные функции компонента API Gateway:

- Авторизация приложений
- Валидация JWT-токена
- Валидация сообщений/запросов
- Защита от угроз в JSON/XML
- Логирование
- Получение данных из кэша
- Маршрутизация запроса до целевого сервиса
- Проверка Rate-Limit
- Проверка Quota
- Смена протокола обмена
- Трансформация тела сообщения
- Трансформация заголовков
- Фильтрация по IP

2.3.2. Компонент - Management API

2.3.2.1. Назначение

Компонент Management API - является обычным RESTful API, который предоставляет сервисы для администрирования (управления и настройки) Системы API Management. Ко всем сервисам доступ ограничен правилами аутентификации и авторизации. Данный компонент используется компонентами Management UI и Portal UI, но также может использоваться для управления Системой через инструменты CI/CD с помощью скриптов автоматизации.

2.3.2.2. Функциональные возможности

Основные функции компонента Management API:

- Управление и конфигурация платформы
- Управление и настройка API
- Предоставление RESTful сервиса для управления и конфигурации платформы и API

2.3.3. Компонент - Management UI

2.3.3.1. Назначение

Компонент Management UI отвечает за администрирование Системы API Management и управление жизненным циклом API, развернутых в соответствующих API Gateway. Является основным веб-интерфейсом для пользователей Системы с ролями «Администратор» и «Публикатор/Издатель».

2.3.3.2. Функциональные возможности

Основные функции компонента Management UI:

- Аутентификация пользователей
- Создание API
- Создание документации API
- Управление доступом к API
- Настройка политик безопасности
- Публикация API
- Просмотр логов API
- Просмотр аналитики использования API
- Просмотр аудит-логов
- Управление версиями API
- Управление настройками платформы
- Управление пользователями
- Управление ролями

2.3.4. Компонент - Portal UI

2.3.4.1. Назначение

Компонент Portal UI отвечает за поиск API, просмотр документации API, тестирование API и подписки на опубликованный API. Является основным веб-интерфейсом для пользователей Системы с ролями «Подписчик». Пользователи могут использовать этот компонент для управления своими приложениями (регистрация приложения, просмотр аналитики и статистики использования API в разрезе приложения).

2.3.4.2. Функциональные возможности

Основные функции компонента Portal UI:

- Аутентификация пользователей
- Поиск API
- Подписка на API
- Просмотр документации API
- Регистрация приложений
- Просмотр логов API
- Просмотр аналитики использования API
- Проставление оценок и комментариев
- Обратная связь с владельцем сервиса

2.3.5. Компонент - PostgreSQL

2.3.5.1. Назначение

Хранение настроек платформы, конфигурации API, настройки приложений, аудит-логи.

2.3.5.2. Функциональные возможности

Основные функции компонента PostgreSQL:

- Хранение настроек платформы
- Хранение настроек и конфигурации API
- Хранение настроек приложений
- Хранение аудит-логов

2.3.6. Компонент - Opensearch

2.3.6.1. Назначение

Хранение данных для аналитики/статистики (метрики, логи).

2.3.6.2. Функциональные возможности

Основные функции компонента Opensearch:

- Хранение логов
- Хранение метрик

2.3.7. Компонент - API Gateway

2.3.7.1. Назначение

Хранение настроек ограничения трафика (Rate-Limit, Quota) по каждому API, кэширование данных.

2.3.7.2. Функциональные возможности

Основные функции компонента Redis:

- Хранение параметров Rate-Limit для API
- Хранение параметров Quota для API
- Хранение кэшированных данных для API

2.4. Отказоустойчивость и масштабирование

2.4.1. База данных

Отказоустойчивость СУБД Postgres должна быть обеспечена одной из стандартных технологий, поддерживаемых Postgres (например, Patroni + ETCD). Горизонтальное масштабирование для такого класса систем как правило дополнительно не требуется.

Отказоустойчивость **Opensearch** должна быть обеспечена репликацией данных между тремя (минимальная конфигурация) серверами в кластере, которые должны быть развернуты в разных ЦОД. При необходимости возможно горизонтальное масштабирование добавлением нового сервера в кластер.

Отказоустойчивость **Redis** должна быть обеспечена репликацией данных между тремя (минимальная конфигурация) серверами в кластере, которые должны быть развернуты в разных ЦОД. При необходимости возможно горизонтальное масштабирование добавлением нового сервера в кластер.

2.4.2. Серверы приложений

Отказоустойчивость и горизонтальное масштабирование серверов приложений обеспечена Active-Active кластеризацией с распределением нагрузки. В случае выхода из строя одного или нескольких серверов не произойдет потери данных или отказа сервиса в целом, только пропорционально снизится производительность системы в целом.

2.5. Программные интерфейсы

Система Гравити Апи Менеджмент Рус реализует несколько интерфейсов для пользователя в соответствии с [функциональной схемой](#):

- Gravitee Portal UI предоставляет веб-интерфейс для разработчиков приложений;
- Gravitee Management UI предоставляет веб-интерфейс по управлению системой для администраторов;
- Gravitee API Gateway предоставляет веб-сервис для обработки HTTP-запросов от клиентов;
- Gravitee Management API предоставляет RESTful API для управления системой;