

Инструкция по запуску и тестированию

Предварительные требования

- Docker и Docker Compose
- Postman для тестирования API

Запуск системы

Перейдите в папку \deployment и запустите файл start.bat.

Тестирование функциональности

Коллекция Postman доступна по адресу <https://www.postman.com/orange-satellite-666437/otus-highload/collection/8uv29vb/otus-highload>

Выполненные работы

1. Развертывание инфраструктуры мониторинга

- **Prometheus** - для сбора бизнес-метрик по принципу RED
- **Zabbix** - для сбора технических метрик инфраструктуры
- **Grafana** - для визуализации и создания дашбордов
- Все сервисы контейнеризованы с использованием Docker

2. Интеграция мониторинга в C# приложение

- Добавлены NuGet пакеты для метрик Prometheus
- Реализован сервис Metrics для бизнес-метрик:
 - Счетчики операций CRUD пользователей
 - Гистограмма времени выполнения запросов
 - Счетчик ошибок запросов
- Настроен эндпоинт /metrics для Prometheus

3. Бизнес-метрики по принципу RED

- **Rate** - количество запросов в секунду
- **Errors** - процент ошибок и счетчик неудачных запросов
- **Duration** - время выполнения запросов (p50, p95, p99)

4. Технические метрики через Zabbix

- Мониторинг системных ресурсов (CPU, память, диск)
 - Сетевые метрики и активные подключения
- #### 5. Дашборды в Grafana

Дашборд Prometheus (бизнес-метрики):

- RED метрики в реальном времени
- Операции с сообщениями
- Статус коды HTTP запросов

Дашборд Zabbix (технические метрики):

- Утилизация ресурсов сервера
- Производительность системы

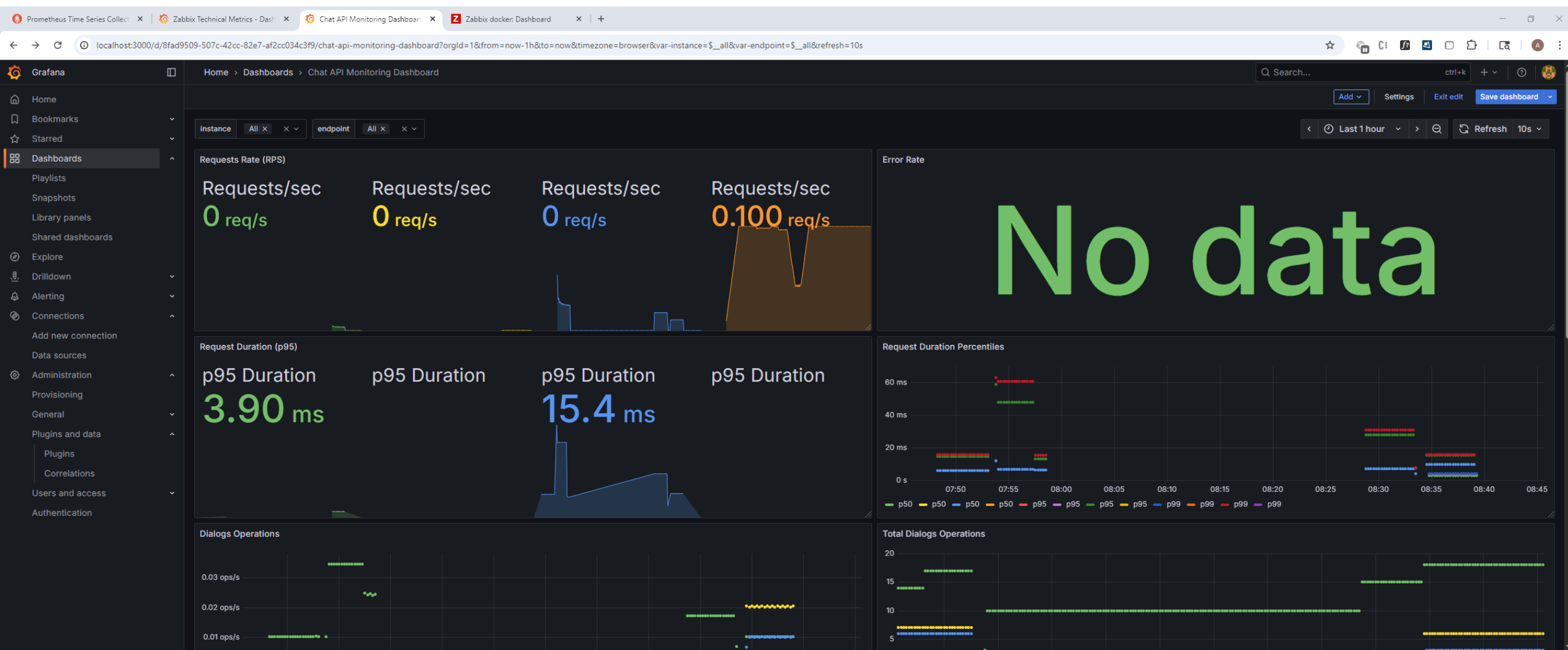
Конфигурация системы

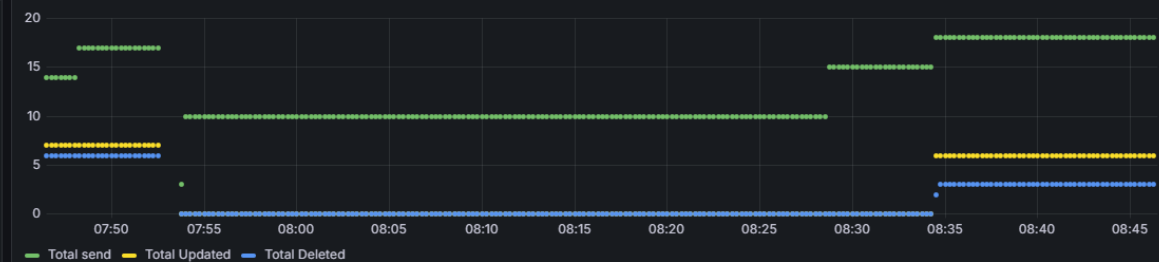
Docker-контейнеры:

- **Grafana:** порт 3000
- **Prometheus:** порт 9091
- **Zabbix Web:** порт 8080
- **Zabbix Server:** порт 10051
- **Node Exporter:** порт 9100

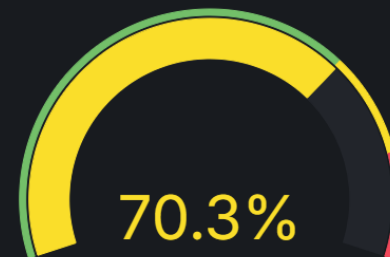
Источники данных:

- Prometheus: <http://prometheus:9090>
- Zabbix: http://zabbix-web:8080/api_jsonrpc.php





Memory Usage



Value



Disk Usage



