



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

**Выпускная квалификационная работа бакалавра
Система авторизации инфраструктурных сервисов**

**Студент: Васильев А. И. ИУ7-82Б
Научный руководитель: Клорикьян П. В.**

Цели и задачи работы

Цель: реализация программно-алгоритмического комплекса для авторизации запросов в инфраструктурные сервисы.

Задачи:

1. Провести обзор предметной области, существующих решений, подходов к аутентификации и авторизации в микросервисной архитектуре.
2. Разработать и описать ключевые алгоритмы работы программно-алгоритмического комплекса, реализующего аутентификацию и авторизацию запросов в инфраструктурные сервисы.
3. Провести исследование влияния работы авторизации на время выполнения запросов в инфраструктурные сервисы.

Обзор предметной области

Микросервисная архитектура – архитектурный подход к разработке программного обеспечения, при котором оно состоит из небольших слабо связанных сервисов.

Инфраструктурный сервис предоставляет базовый функционал для работы сервисов с бизнес-логикой. Пример – база данных.

Решаемая проблема – запросы из сервисов с бизнес-логикой в инфраструктурные должны быть авторизованы во избежании несанкционированного доступа.

Kubernetes и сайдкар контейнер

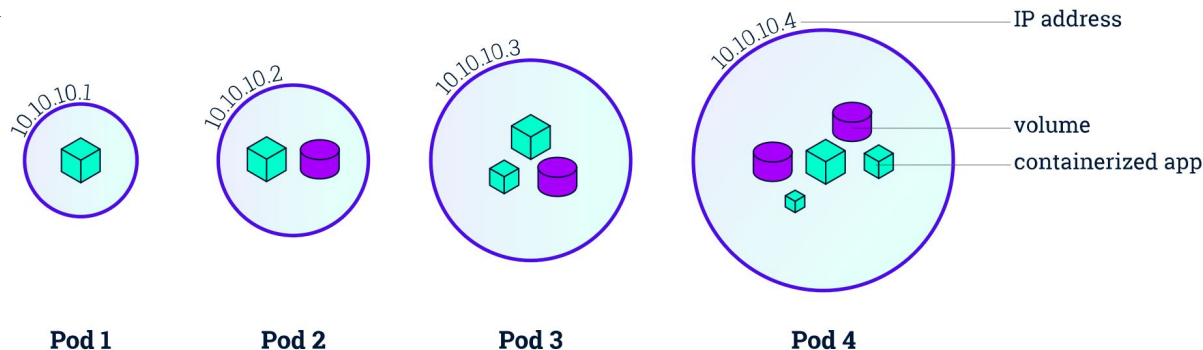
Kubernetes (k8s) – система для управления приложениями в изолированных друг от друга контейнерах.

Под – представляет собой группу из одного или нескольких контейнеров приложения, совместно использующие ресурсы.

Кластер – содержит набор таких рабочих машин.

Сайдкар – дополнительный контейнер, выполняющий вспомогательные для основной функции.

В работе использован как прокси-эндпоинт для внедрения авторизации входящих запросов.



Основные используемые понятия авторизации

Аутентификация – проверка личности сервиса.

Авторизация – проверка доступов сервиса.

OAuth 2.0 – открытый стандарт авторизации, позволяет приложению получать ограниченный доступ к ресурсам от сервиса.

OIDC – надстройка над OAuth 2.0 для аутентификации. Предоставляет стандартный способ аутентификации субъекта (ID Token в формате JWT).

JSON Web Token (JWT) – зашифрованный приватным ключом токен в формате json.

IdP (Identity Provider) – сервис, управляющий идентификацией и правами доступа, выпускающий токены с учетными данными.

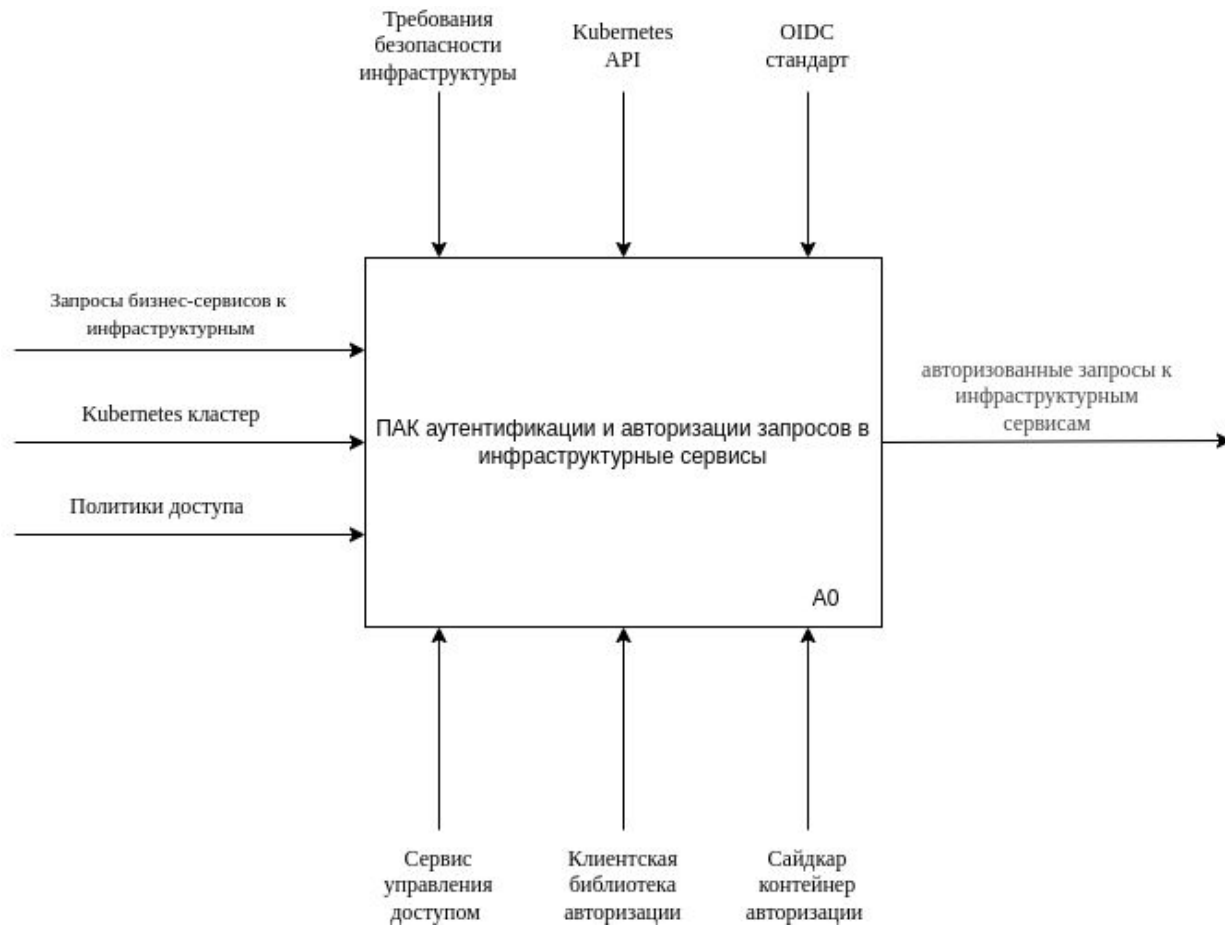
```
eyJhbGciOiJIUzI1NiIsImRlZCI6Imx0cS9mWVYtZC7N4YnFlD0ZqR0tKTjAwTDJicjd3eU8yZ25wYnHUNlR0TmsifQ.eyJhdGUiOiJsiaHR0bGM6LWY5dmlzClM5dGVzMlRlZmFlbHQC3ZJlNmNsdxXN0ZXIubG9yYWwiLCJrM3MiSw1ZXBhWjcxOnc2NjM4NDUwLCCyYXQ1QjE3NDgxMDI0NTAsImlzcyI6Imh0dHBZOi8va3ViZXJvZXRlc3kZWZhW0LnN2Y5bjBhVzdGVyLmxvY2FsIiwianRpIjo1ZnZyZDA5M2QtYzIZYS00NTJlTG10YTZDU3MTQzMzA1OTIxIiwia3ViZXJvZXRlc3pbyI6eyJuYW1lczBhY2U1OjIwb3N0Z2Jlc3Y1h1IiwibW9kZSI6I2I6eyJuYW1lIjoiazNkLWJtc3R1Y2x1c3Rlc1ciZXJkZXI0MCIsInVpZCI6IjJldjNjQ1NDMzLWQxYTMtNGZhc0ZmI2LWI3ZTI3ZmI4OGVLZCJ9LCCjw2Q1OnsibmFtZSI6Inb3RcnMcVzLWEtNjc5NGZgYjYyMyNy1iZmp3cSIsInVpZCI6IjY1NDBlMTM5LWNjUGUtNDgxMC1hNjNiLTRL1MQG3ODdc1MjhmcJ9LCjZzX2J2aWN1YWNjb3Vu dCI6eyJuYW1lIjo1ZGVmYXVsdcIsInVpZCI6ImEweYiWZTlRmLWQ2ZWYtNGFGOC04ODQ1LTgyMWJhZTMyZTJlZCJ9LCCjY3YXJvY20ZXI0ejE3NDgxMDYwNW Td9LCCjUuYmY1OjE3NDgxMDI0NTAsInN1YiI6InN5 c3RlbTpzZXJ2aWN1YWNjb3VudDpwbn3N0Z2Jlc3Y1 h0MRlZmFlbHQifQ.TKX3WQJueYR3f30ohF7adtSyYft6Us7C5csichRxFdOOBMSH4V5DLvWh6cDD45ggyQ0rQ1-1LKR0b19ZJ5Qyebhr_k_BK56hDrARks20ufNE RON0k-
```

Пример k8s ID Token в формате JWT

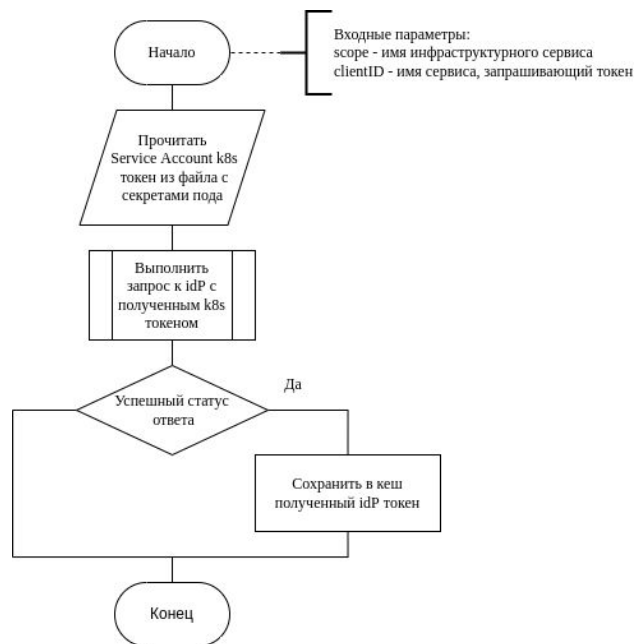
Обзор существующих решений

Решение	Централизация	Локальная валидация	Ротация секретов	Универсальность
Service Mesh	нет	нет	да	нет
SPIFFE/SPIRE	нет	да	да	да
Встроенные RBAC БД	нет	да	нет	нет
OAuth2 Proxy с OPA	да	нет	нет	нет
Предлагаемое решение	да	да	да	да

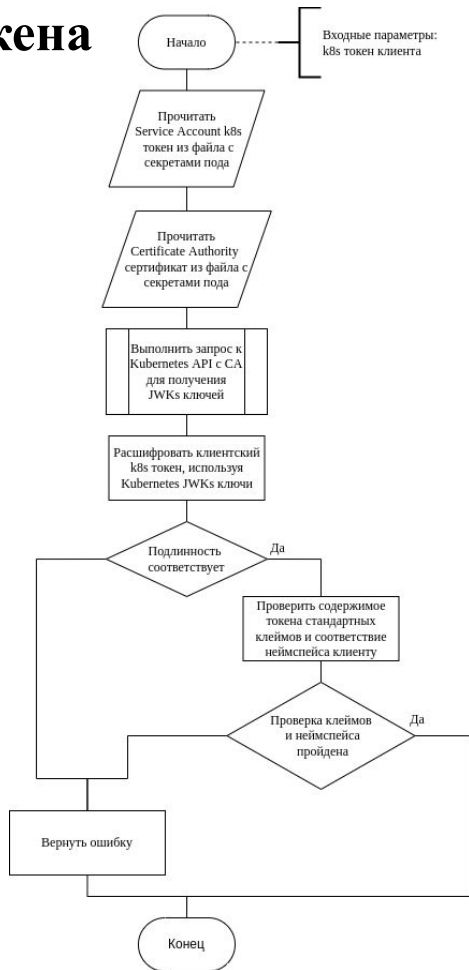
Формальная постановка задачи



Алгоритм выпуска IdP токена



Алгоритм проверки k8s токена



Алгоритм авторизации входящего запроса

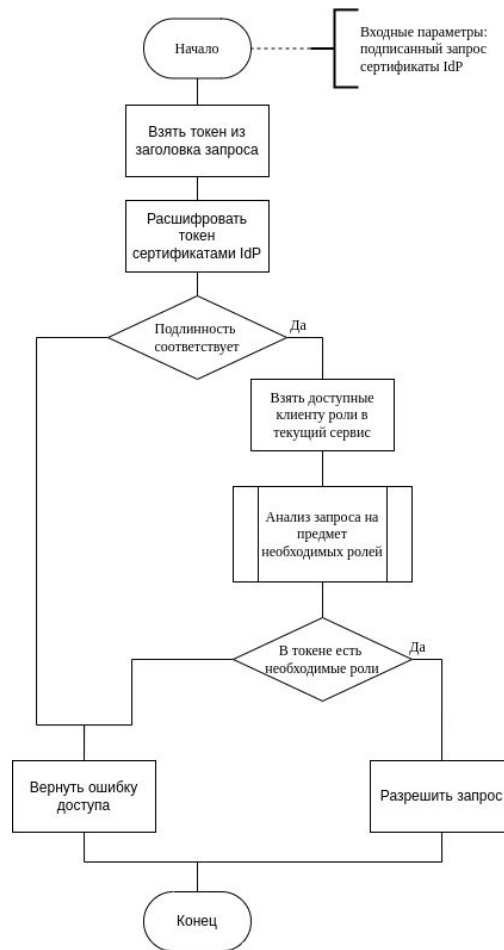
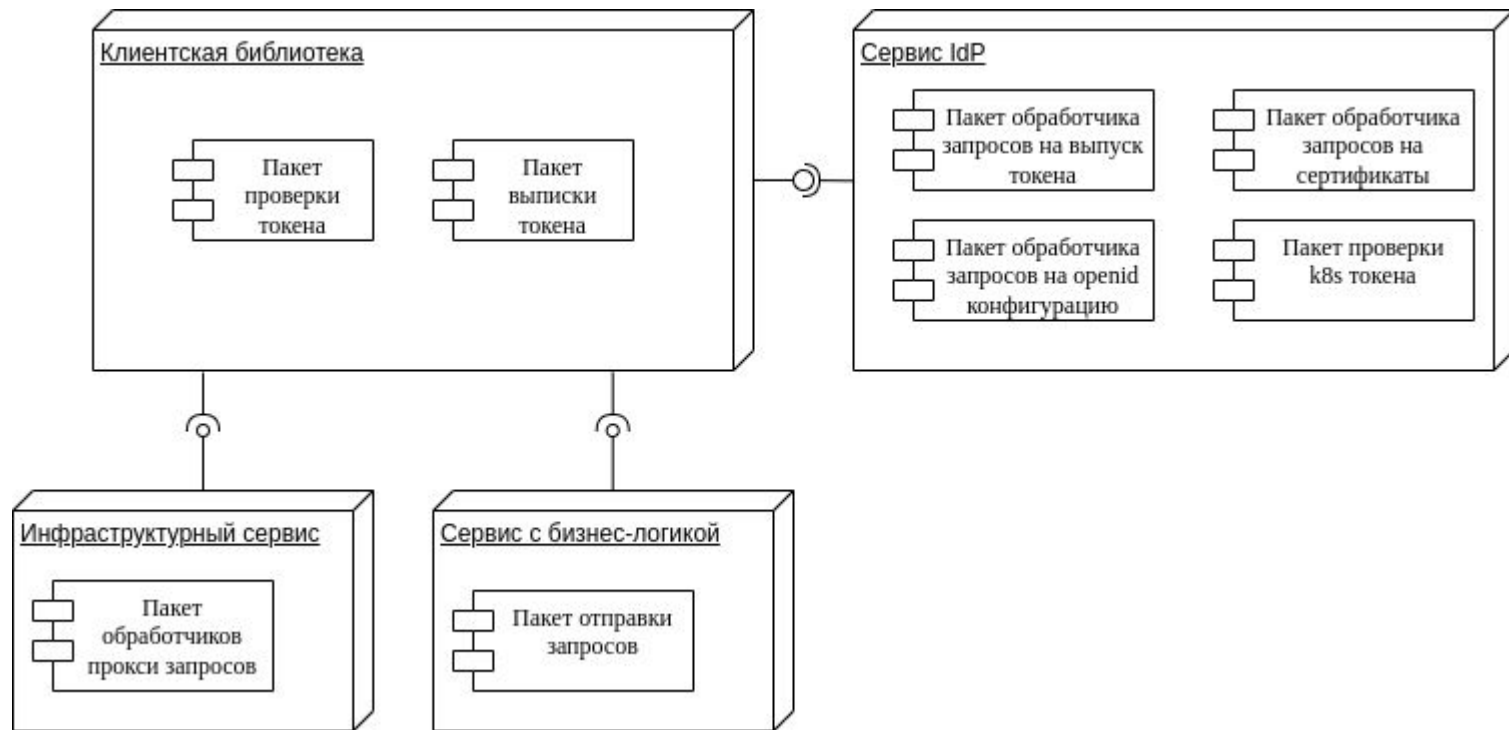


Диаграмма компонентов разработанного ПО



Пример интерфейса ПО

Административная панель авторизации

Настройки сервиса

Выберите сервис:

service-a ▼

☒ Подпись запросов включена

☒ Проверка подлинности запросов включена

Применить Обновить токены

Глобальные настройки

☒ Подпись запросов включена (все сервисы)

☒ Проверка подлинности запросов включена (все сервисы)

Применить для всех сервисов

Управление правами

Просмотр текущих прав

Client:

service-a

Scope:

postgres-a

Посмотреть текущие права

Текущие права: RW

Добавление новых прав

Client:

service-a

Scope:

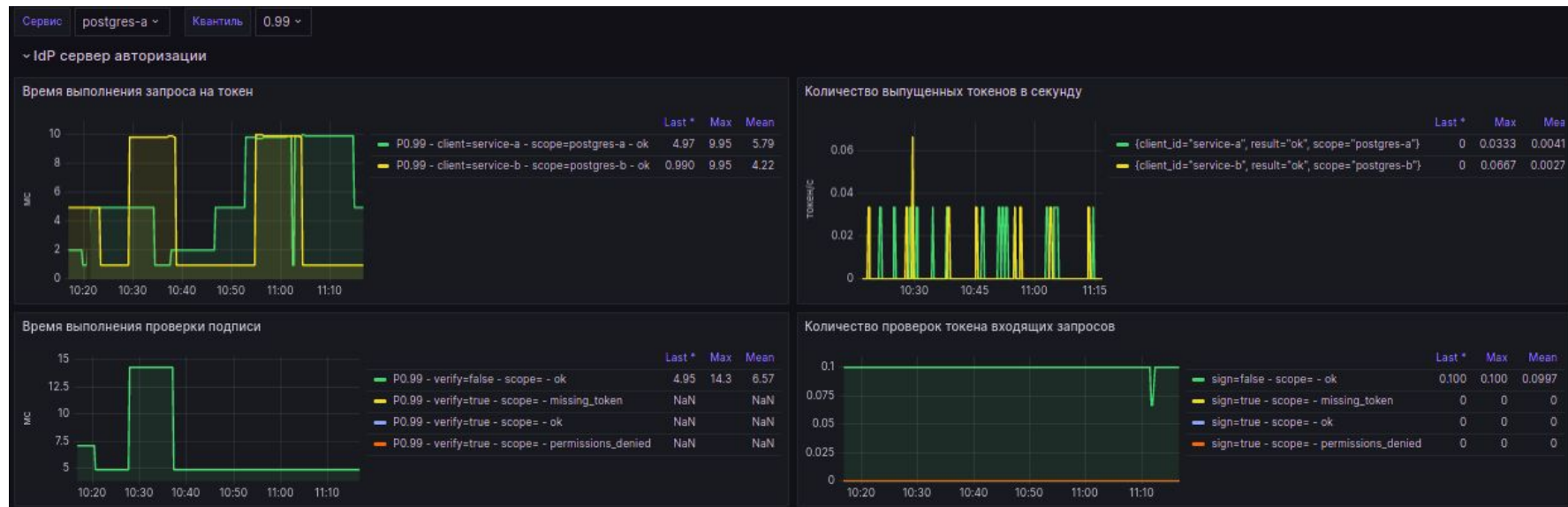
postgres-a

Roles (через запятую):

RW

Добавить новые права

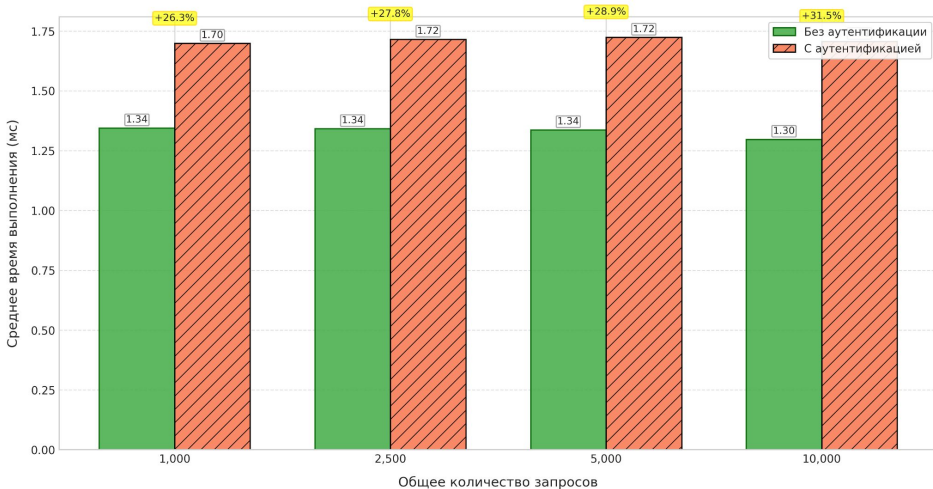
Мониторинг показателей сервисов



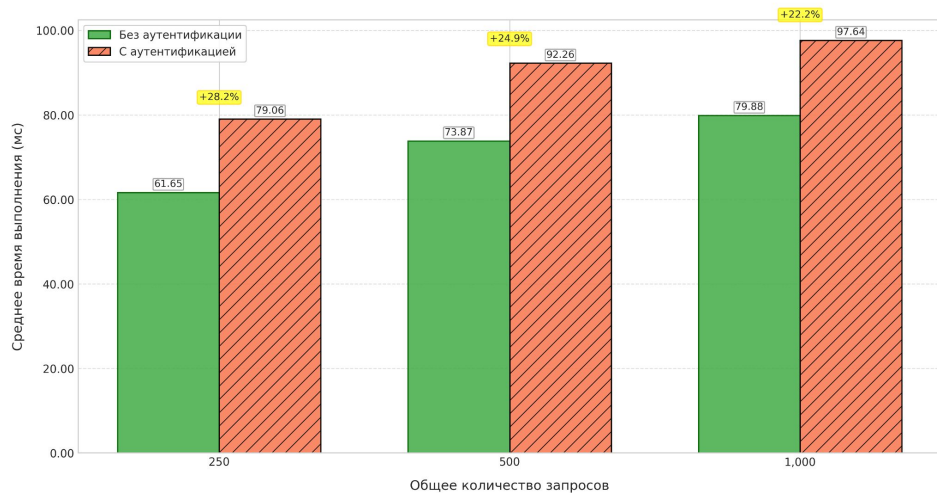
- **Prometheus** – инструмент для сбора данных (метрик) сервиса.
- **Grafana** – инструмент для визуализации собранных данных.
- Всего было задано **15 метрик**.

Исследование влияния работы авторизации

Сравнение времени выполнения запросов: Лёгкие запросы



Сравнение времени выполнения запросов: Тяжёлые запросы



```
`INSERT INTO log (message) VALUES (Benchmark at %s", time.Now())`
```

```
`WITH heavy_cte AS (SELECT generate_series(1,1000000) AS data)  
SELECT COUNT(*), AVG(data) FROM heavy_cte`
```

Заключение

Выполнена цель: реализован программно-алгоритмический комплекс для авторизации запросов в инфраструктурные сервисы.

Выполнены задачи:

1. Проведен обзор предметной области, существующих решений, подходов к аутентификации и авторизации в микросервисной архитектуре.
2. Разработаны и описаны ключевые алгоритмы работы программно-алгоритмического комплекса, реализующего аутентификацию и авторизацию запросов в инфраструктурные сервисы.
3. Проведено исследование влияния работы авторизации на время выполнения запросов в инфраструктурные сервисы.