## **Сценарии**

**Дата теста** 09.10.2024

1. Выбираем 2 запроса на чтение (/user/get/{id} и /user/search из спецификации).

План нагрузочного тестирования, который шлет запросы на эти api: делаем вызов эндпойнтов /user/get/{id} и /user/search jmeter с помощью 10 потоков в течении 5 минут.

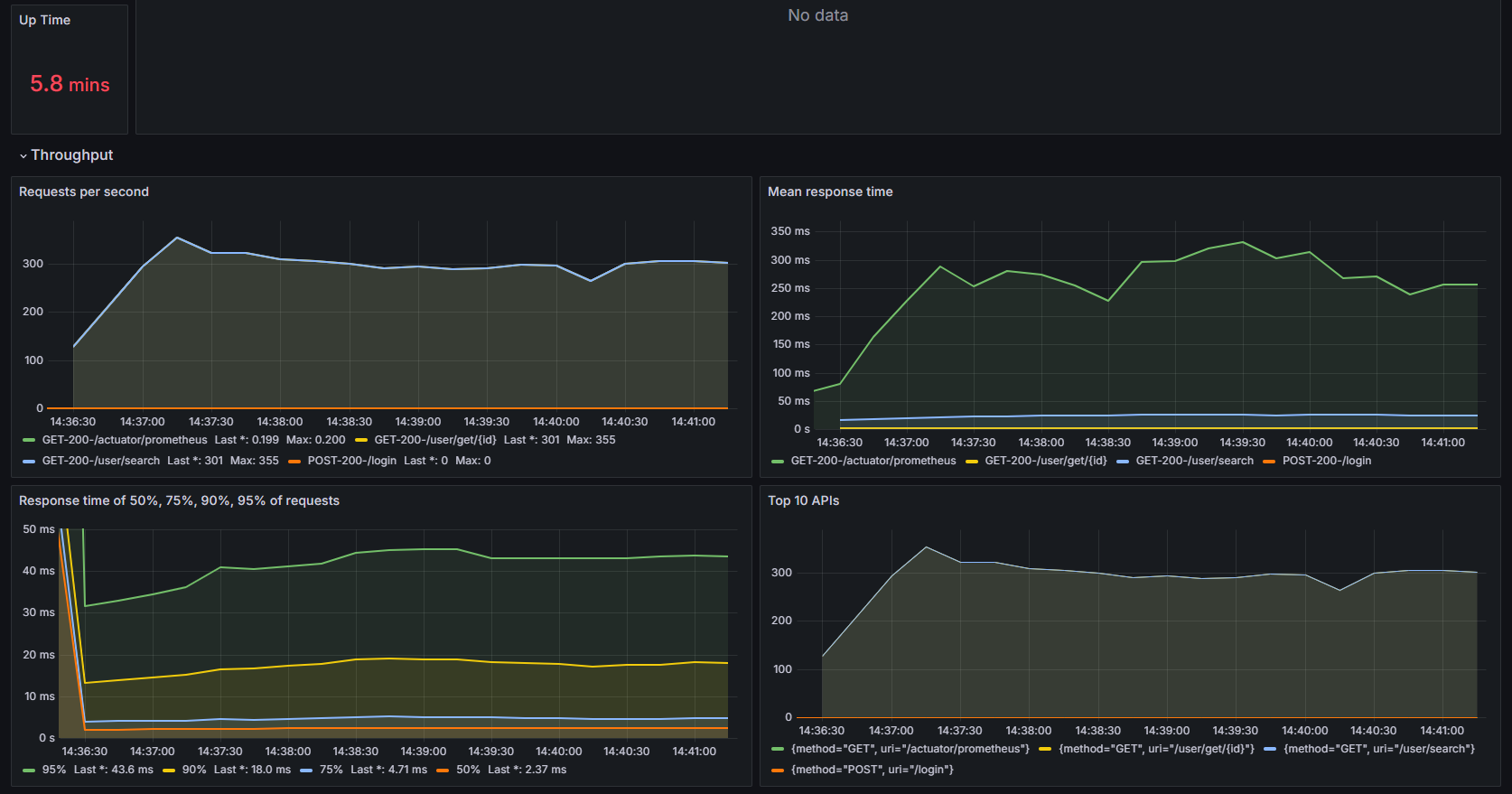
1. Создаем нагрузку на чтение с помощью составленного на предыдущем шаге плана, делаем замеры.

Получаем следующие показатели:

- RPS ~ 300

- Response time of 90% of request: 18 ms

График:



1. Настраиваем 2 слейва и 1 мастер. Включаем потоковую репликацию (синхронную и асинхронную).
2. Добавляем в проект replicated datasource - ReplicationRoutingDataSource, который выбирает использовать master или slave в зависимости от типа транзакции.

Переносим запросы на чтение (transaction == read only) на чтение со слейва.

1. Создаем нагрузку на чтение с помощью составленного на предыдущем шаге плана, делаем замеры. Добавить сравнение результатов в отчет.

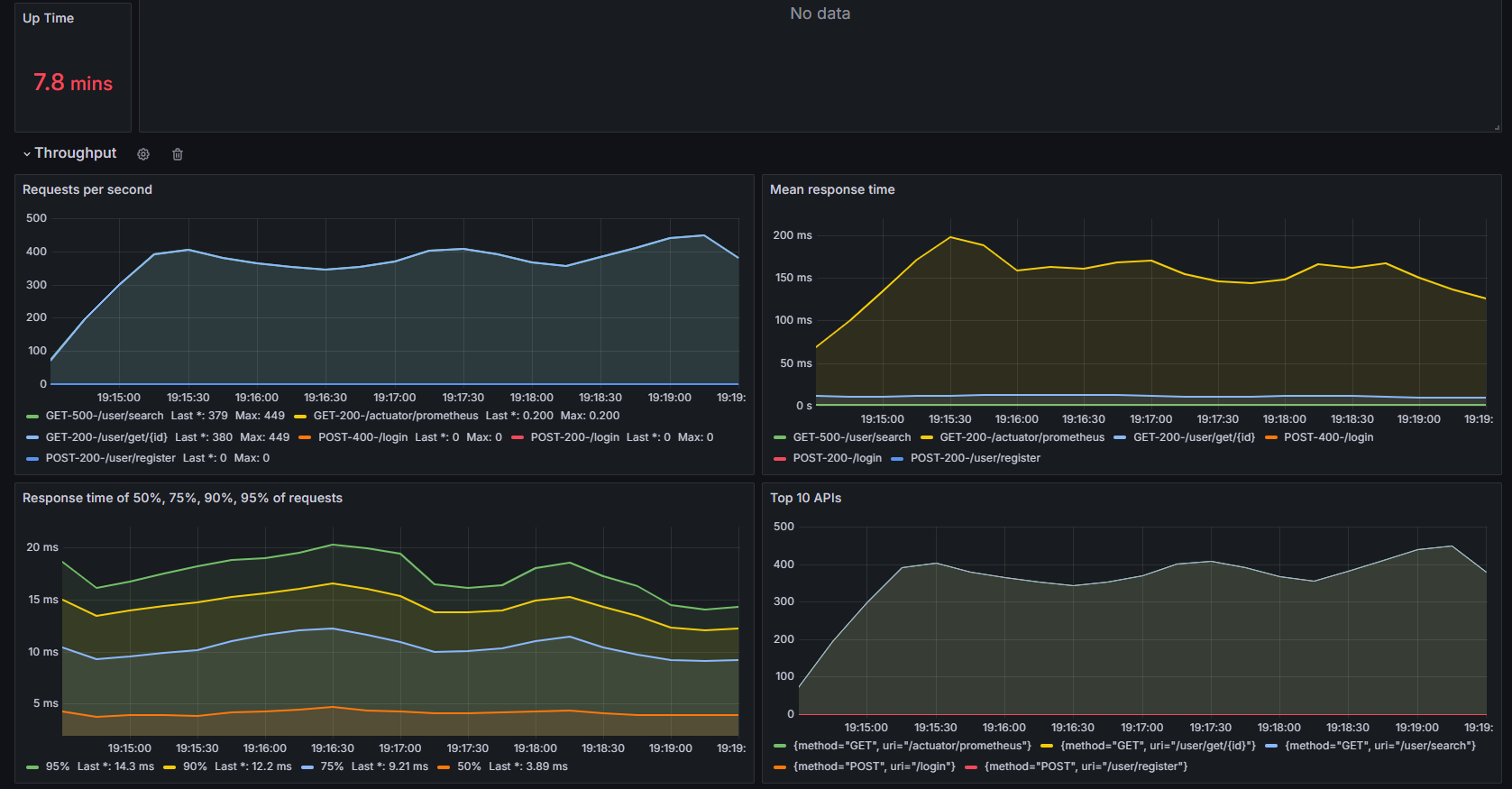
Синхронная репликация

Получаем следующие показатели:

- RPS ~ 355

- Response time of 90% of request: 15 ms

График:



Выводы: после масштабирования БД запросы на чтение стали выполняться быстрее.

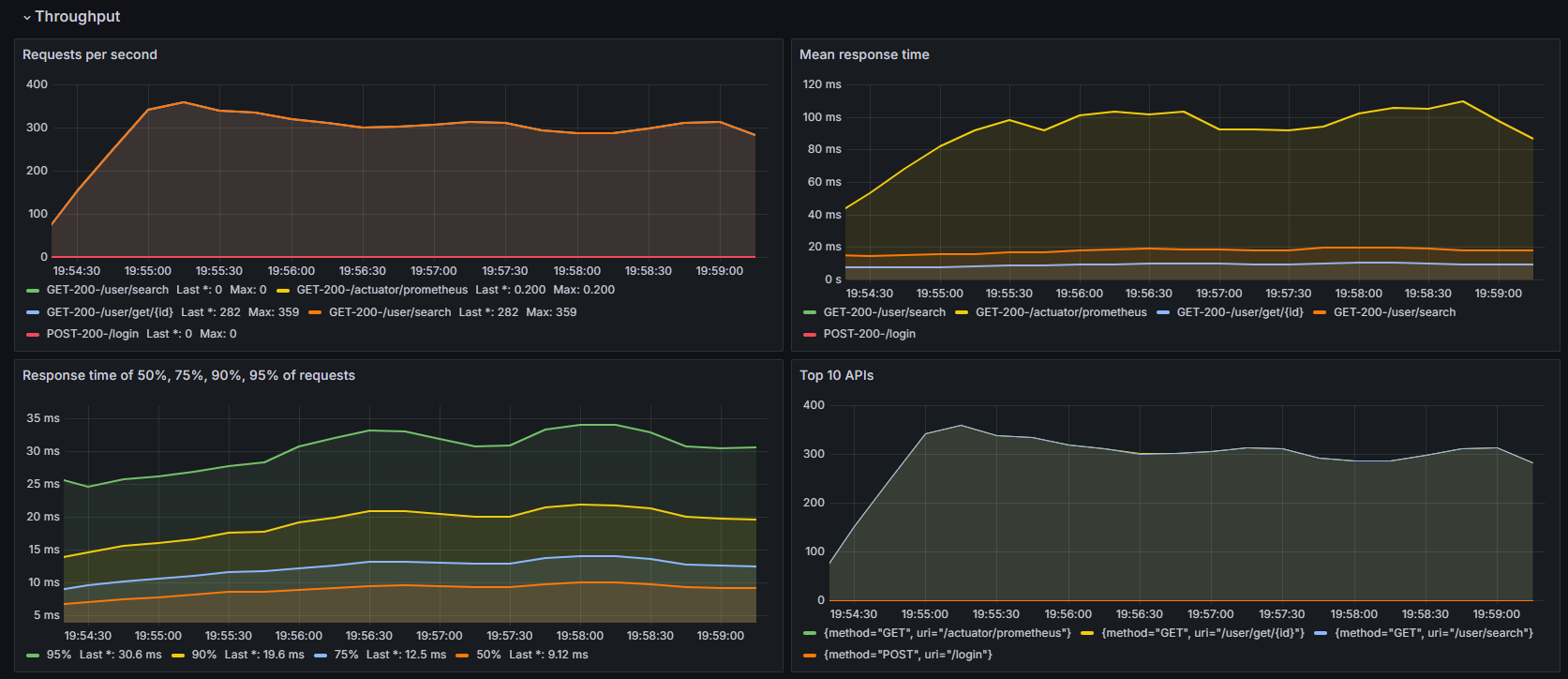
Асинхронная репликация

Получаем следующие показатели:

- RPS ~ 313

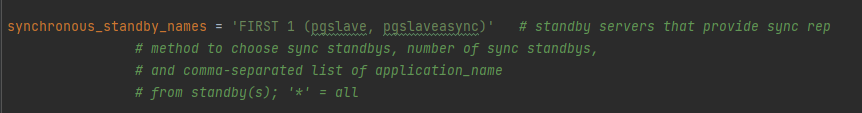
- Response time of 90% of request: 20 ms

График:



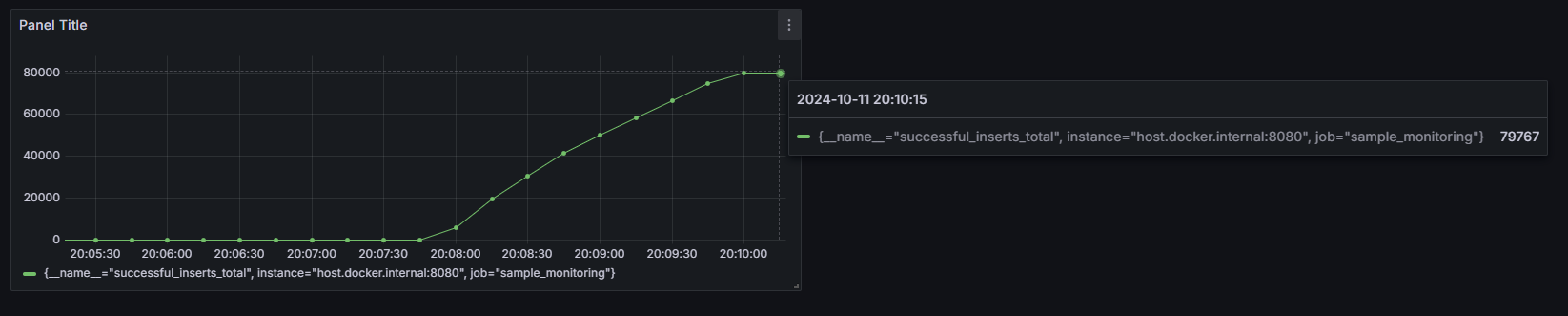
Выводы: после асинхронной репликации запросы на чтение стали выполняться незначительно медленнее.

1. Настраиваем кворумную синхронную репликацию.



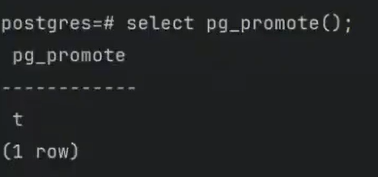
1. Создаем нагрузку на запись в таблицу posts. На стороне, которой нагружаем считать, сколько строк мы успешно записали.

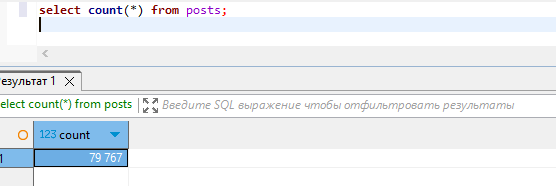
Для фиксации количества успешных записей используем дашборд Grafana.



Успешно записалось 79767 строк в таблицу posts.

1. Убиваем одну из реплик (docker stop pgslaveasync)
2. Заканчиваем нагрузку на запись.
3. Выбираем самый свежий слейв - psslave. Промоутим его до мастера. Переключаем на него второй слейв.



1. Проверяем, есть ли потери транзакций. 

Потери транзакций не произошло, так как pgslave был подключен синхронно, а pgslaveasync – асинхронно.