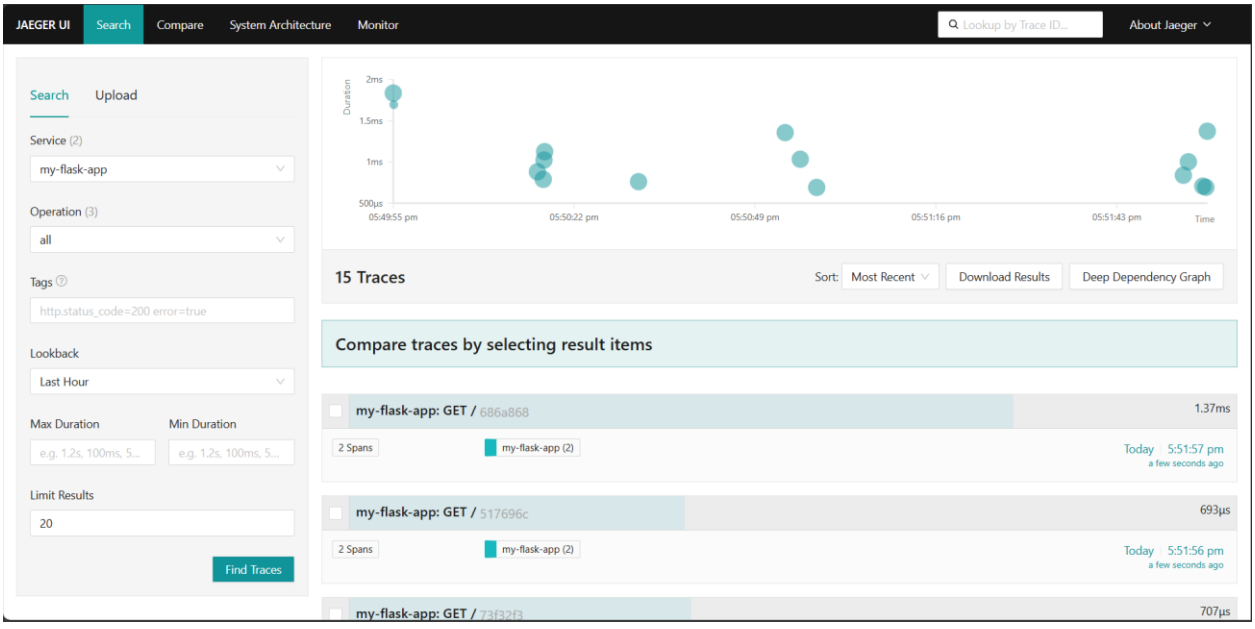
Лабораторная работа № 3

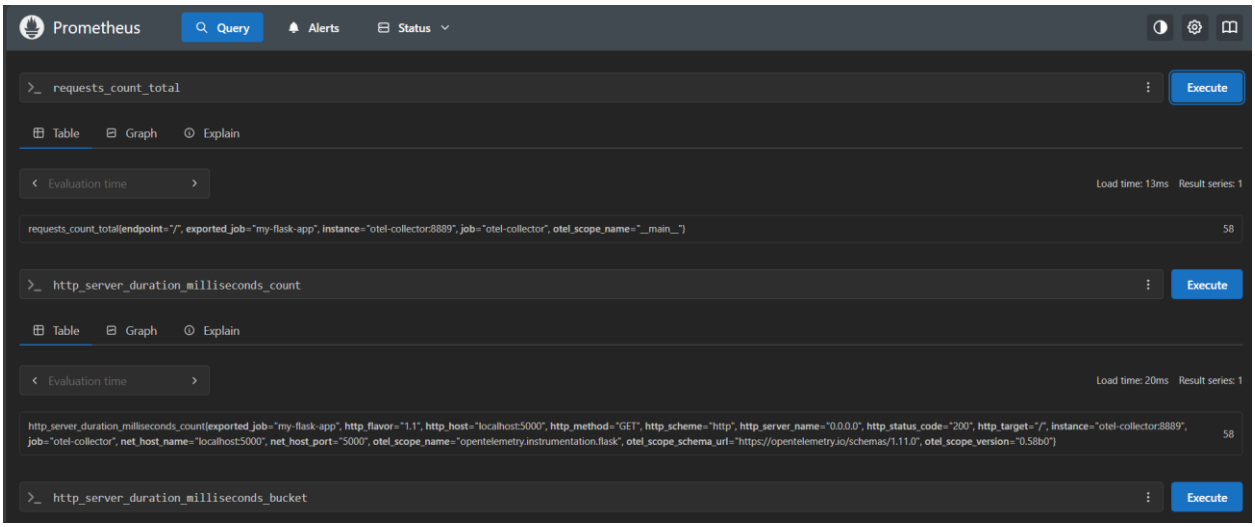
Задание 1

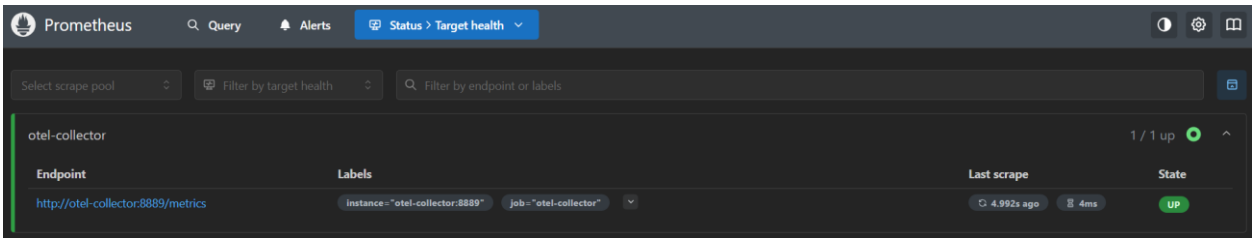
```
D:\PAPKA\НИиСПО\labs\lab3>docker-compose up -d
time="2025-09-26T17:48:22+03:00" level=warning msg="D:\\PAPKA\\НИиСПО\\labs\\lab3\\docker-compose.yml: the attribute `ve
rsion` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 6/6
 ✔ jaeger Pulled                                                                                                  7.7s
   ✔ 9824c27679d3 Already exists                                                                                  0.0s
   ✔ f7e653370860 Pull complete                                                                                   0.6s
   ✔ 8d7378b5be59 Pull complete                                                                                   0.7s
   ✔ cd2d8fa92658 Pull complete                                                                                   5.0s
   ✔ 5c2a38d12742 Pull complete                                                                                   5.1s
Compose can now delegate builds to bake for better performance.
 To do so, set COMPOSE_BAKE=true.
[+] Building 21.2s (12/12) FINISHED                                                                docker:desktop-linux
 => [my-flask-app internal] load build definition from Dockerfile                                                 0.1s
 => => transferring dockerfile: 205B                                                                              0.0s
 => [my-flask-app internal] load metadata for docker.io/library/python:3.9-slim                                   2.0s
 => [my-flask-app auth] library/python:pull token for registry-1.docker.io                                        0.0s
 => [my-flask-app internal] load .dockerignore                                                                    0.1s
 => => transferring context: 64B                                                                                  0.0s
 => [my-flask-app 1/5] FROM docker.io/library/python:3.9-slim@sha256:cf0704507972b63c9b20382dd6f05248572d6b259614  4.1s
 => => resolve docker.io/library/python:3.9-slim@sha256:cf0704507972b63c9b20382dd6f05248572d6b25961410305f96479bf  0.1s
 => => sha256:cf0704507972b63c9b20382dd6f05248572d6b25961410305f96479bf2e8a23c 10.36kB / 10.36kB                  0.0s
 => => sha256:161727d2d61fdfe4836d11f82fb437a3fcb2f4ce5b85951805f0717687ce110f 1.74kB / 1.74kB                    0.0s
 => => sha256:56cea0119ab69043114ce215d355f9f343a55b74b58001450df2e00478fb3529 5.30kB / 5.30kB                    0.0s
 => => sha256:1d454ace0e384876850a0aa5ef6b8c45705445114ab233959bdab71a577b9200 1.29MB / 1.29MB                    0.5s
 => => sha256:41dc2499d8fe1ea2351cc01f3716ce6a95ad0e9bf90c0819fd0c4a93cf4e9b24 13.37MB / 13.37MB                  2.7s
 => => sha256:7fcdf9369fa96e0413fe19da3d316fb6c3bfb0d7371fa4ce617617cac3e8de12 249B / 249B                        0.7s
 => => extracting sha256:1d454ace0e384876850a0aa5ef6b8c45705445114ab233959bdab71a577b9200                         0.3s
 => => extracting sha256:41dc2499d8fe1ea2351cc01f3716ce6a95ad0e9bf90c0819fd0c4a93cf4e9b24                         1.0s
 => => extracting sha256:7fcdf9369fa96e0413fe19da3d316fb6c3bfb0d7371fa4ce617617cac3e8de12                         0.0s
 => [my-flask-app internal] load build context                                                                    0.1s
 => => transferring context: 1.34kB                                                                               0.0s
 => [my-flask-app 2/5] WORKDIR /app                                                                               0.9s
 => [my-flask-app 3/5] COPY requirements.txt .                                                                    0.1s
 => [my-flask-app 4/5] RUN pip install --no-cache-dir -r requirements.txt                                        12.7s
 => [my-flask-app 5/5] COPY app.py .                                                                              0.2s
 => [my-flask-app] exporting to image                                                                             0.6s
 => => exporting layers                                                                                           0.5s
 => => writing image sha256:82431424c5e961e68c87e2a989591491357d0757bb901fccc0f32bb3105636c7                      0.0s
 => => naming to docker.io/library/lab3-my-flask-app                                                              0.0s
 => [my-flask-app] resolving provenance for metadata file                                                         0.0s
[+] Running 4/4
 ✔ my-flask-app                 Built                                                                            0.0s
 ✔ Network lab3_default         Created                                                                          0.1s
 ✔ Container lab3-jaeger-1      Started                                                                          0.9s
 ✔ Container lab3-my-flask-app-1  Started                                                                        0.9s
```



Задание 2

```
D:\РАРКА\НИиСПО\labs\lab3\task2>docker-compose up -d
Compose can now delegate builds to bake for better performance.
 To do so, set COMPOSE_BAKE=true.
[+] Building 2.8s (12/12) FINISHED                                                                  docker:desktop-linux
 => [my-flask-app internal] load build definition from Dockerfile                                                  0.0s
 => => transferring dockerfile: 204B                                                                               0.0s
 => [my-flask-app internal] load metadata for docker.io/library/python:3.9-slim                                    2.5s
 => [my-flask-app auth] library/python:pull token for registry-1.docker.io                                         0.0s
 => [my-flask-app internal] load .dockerignore                                                                     0.0s
 => => transferring context: 64B                                                                                   0.0s
 => [my-flask-app 1/5] FROM docker.io/library/python:3.9-slim@sha256:cf0704507972b63c9b20382dd6f05248572d6b259614  0.0s
 => [my-flask-app internal] load build context                                                                     0.0s
 => => transferring context: 63B                                                                                   0.0s
 => CACHED [my-flask-app 2/5] WORKDIR /app                                                                         0.0s
 => CACHED [my-flask-app 3/5] COPY requirements.txt .                                                              0.0s
 => CACHED [my-flask-app 4/5] RUN pip install --no-cache-dir -r requirements.txt                                   0.0s
 => CACHED [my-flask-app 5/5] COPY app.py .                                                                        0.0s
 => [my-flask-app] exporting to image                                                                              0.0s
 => => exporting layers                                                                                            0.0s
 => => writing image sha256:335fa938566c5b875927c8242e5b8fd3c9fdd7ee65a6414288846daff5cd5820                       0.0s
 => => naming to docker.io/library/task2-my-flask-app                                                              0.0s
 => [my-flask-app] resolving provenance for metadata file                                                          0.0s
[+] Running 5/5
 ✔ my-flask-app                    Built                                                                           0.0s
 ✔ Network task2_default           Created                                                                         0.1s
 ✔ Container task2-prometheus-1    Started                                                                         0.9s
 ✔ Container task2-otel-collector-1  Started                                                                       1.0s
 ✔ Container task2-my-flask-app-1  Started                                                                         1.3s
```

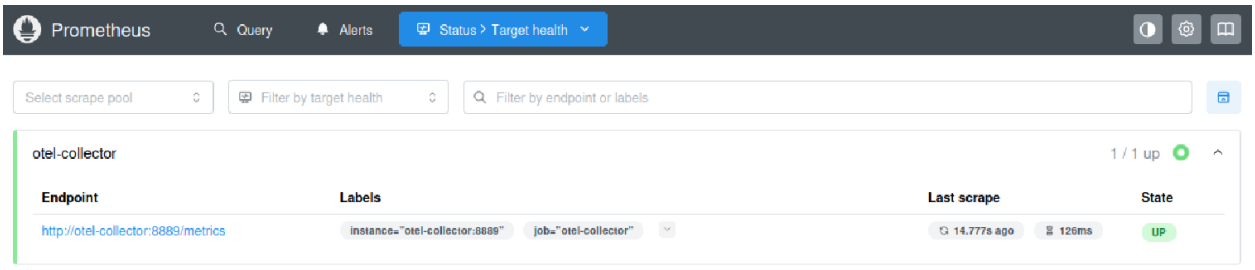


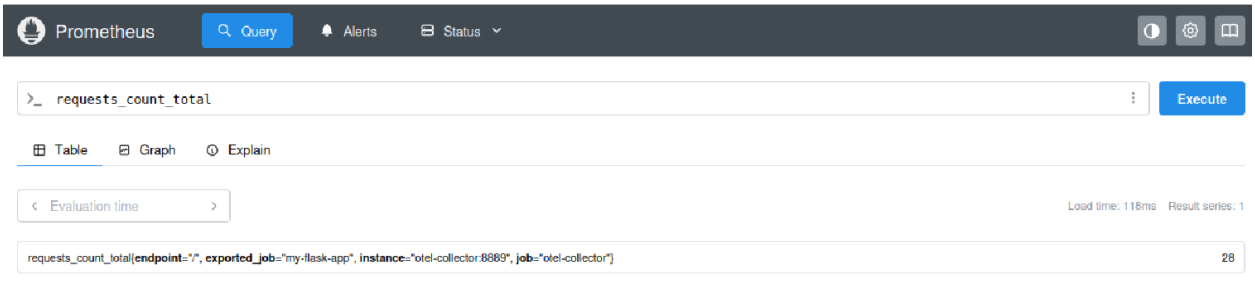


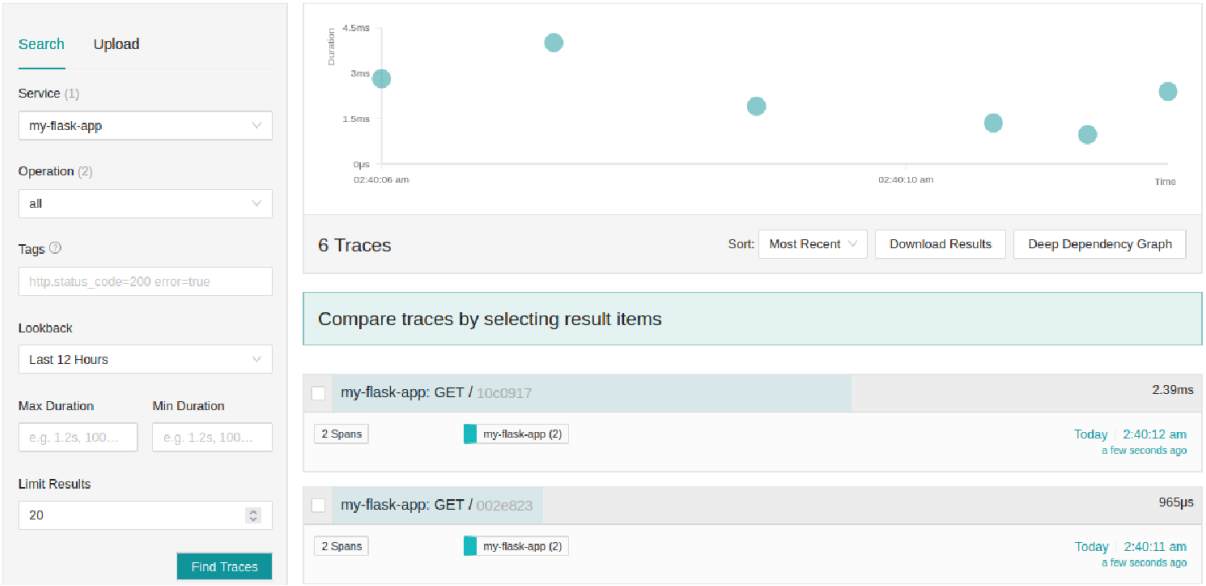
Задание 3

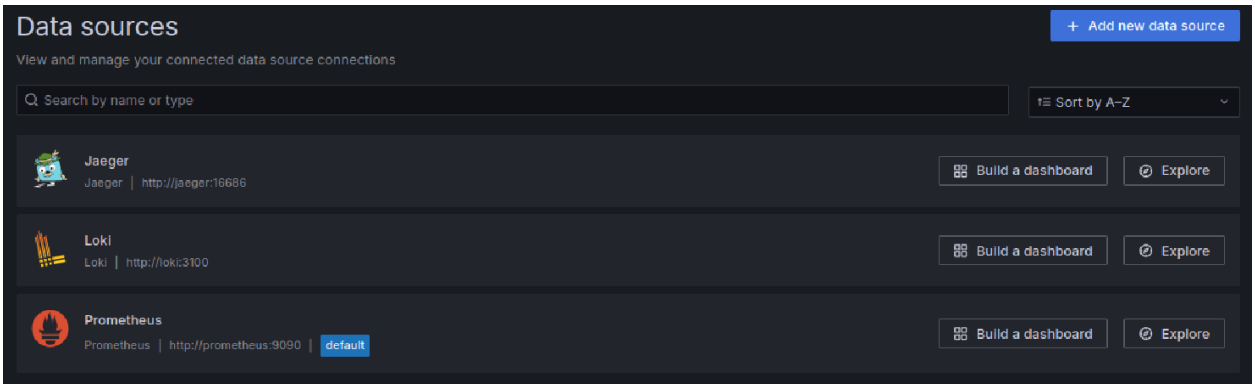
```
admin12345@admin12345-VirtualBox:~/Desktop/lab3/temp_repo/task3$ UID=$(id -u) GID=$(id -g) docker compose up -d
bash: UID: readonly variable
WARN[0000] The "UID" variable is not set. Defaulting to a blank string.
[+] Running 8/8
 ✔ Network task3_monitoring          Created
 ✔ Container task3-prometheus-1      Started
 ✔ Container task3-promtail-1        Started
 ✔ Container task3-loki-1            Started
 ✔ Container task3-my-flask-app-1    Started
 ✔ Container task3-grafana-1         Started
 ✔ Container task3-jaeger-1          Started
 ✔ Container task3-otel-collector-1  Started
```

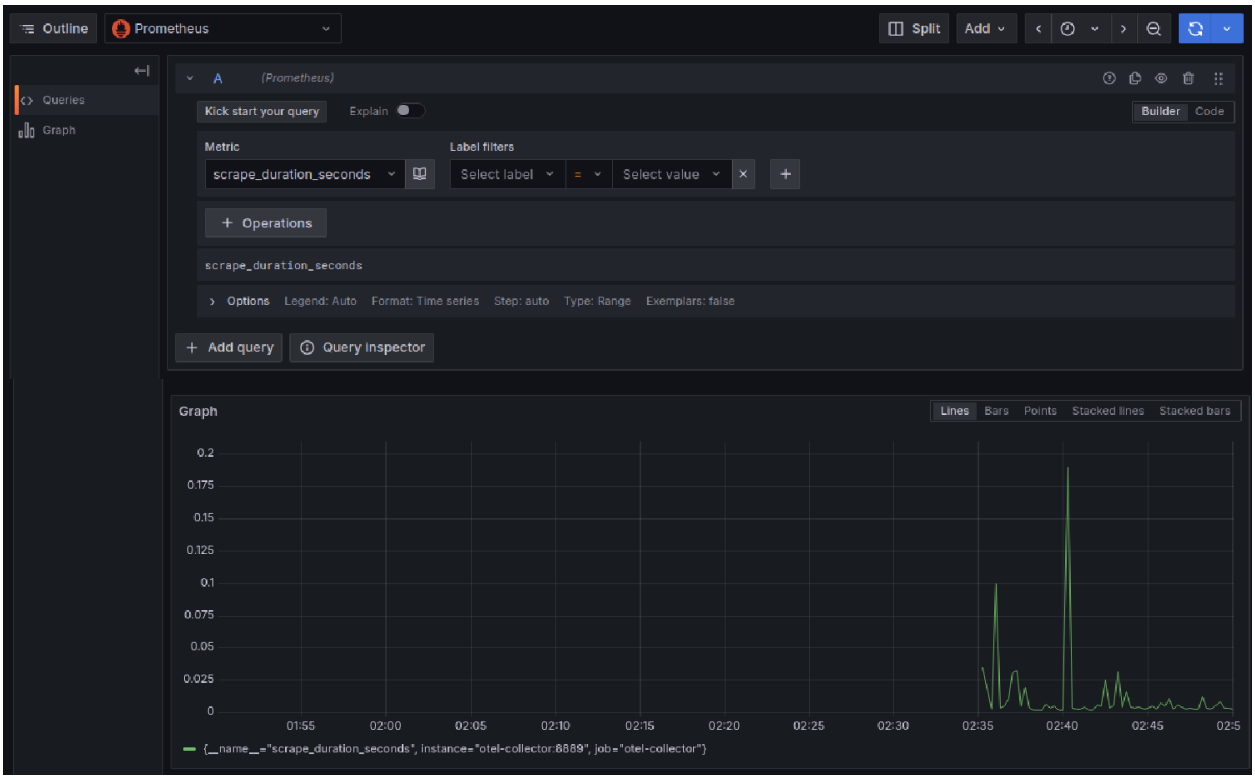
```
admin12345@admin12345-VirtualBox:~/Desktop/lab3/temp_repo/task3$ docker ps
CONTAINER ID   IMAGE                                    COMMAND                  CREATED          STATUS          P
ORTS                                                                                                            NAM
ES
177bedd4b740   otel/opentelemetry-collector-contrib:0.85.0   "/otelcol-contrib --…"   39 minutes ago   Up 38 minutes   0
.0.0.0:4318->4318/tcp, [::]:4318->4318/tcp, 4317/tcp, 55678-55679/tcp, 0.0.0.0:8889->8889/tcp, [::]:8889->8889/tcp   tas
k3-otel-collector-1
29ea28f26e1c   task3-my-flask-app                       "python app.py"          39 minutes ago   Up 39 minutes   0
.0.0.0:5000->5000/tcp, [::]:5000->5000/tcp                                                                      tas
k3-my-flask-app-1
9240420e8721   grafana/grafana:latest                   "/run.sh"                39 minutes ago   Up 39 minutes   0
.0.0.0:3000->3000/tcp, [::]:3000->3000/tcp                                                                      tas
k3-grafana-1
f823f35e4c84   grafana/promtail:2.9.0                   "/usr/bin/promtail -…"   39 minutes ago   Up 39 minutes
                                                                                                                tas
k3-promtail-1
87c87b62c7e8   grafana/loki:2.9.0                       "/usr/bin/loki -conf…"   39 minutes ago   Up 39 minutes   0
.0.0.0:3100->3100/tcp, [::]:3100->3100/tcp                                                                      tas
k3-loki-1
597e722977fb   prom/prometheus:latest                   "/bin/prometheus --c…"   39 minutes ago   Up 39 minutes   0
.0.0.0:9090->9090/tcp, [::]:9090->9090/tcp                                                                      tas
k3-prometheus-1
ad7d4fb8c5c6   jaegertracing/all-in-one:latest          "/go/bin/all-in-one-…"   39 minutes ago   Up 39 minutes   4
317-4318/tcp, 9411/tcp, 14250/tcp, 14268/tcp, 0.0.0.0:16686->16686/tcp, [::]:16686->16686/tcp                   tas
k3-jaeger-1
0533380db0a2   kindest/node:v1.25.3                     "/usr/local/bin/entr…"   32 hours ago     Up 6 hours      0
.0.0.0:80->80/tcp, 0.0.0.0:443->443/tcp, 127.0.0.1:43039->6443/tcp                                              kin
d-control-plane
```

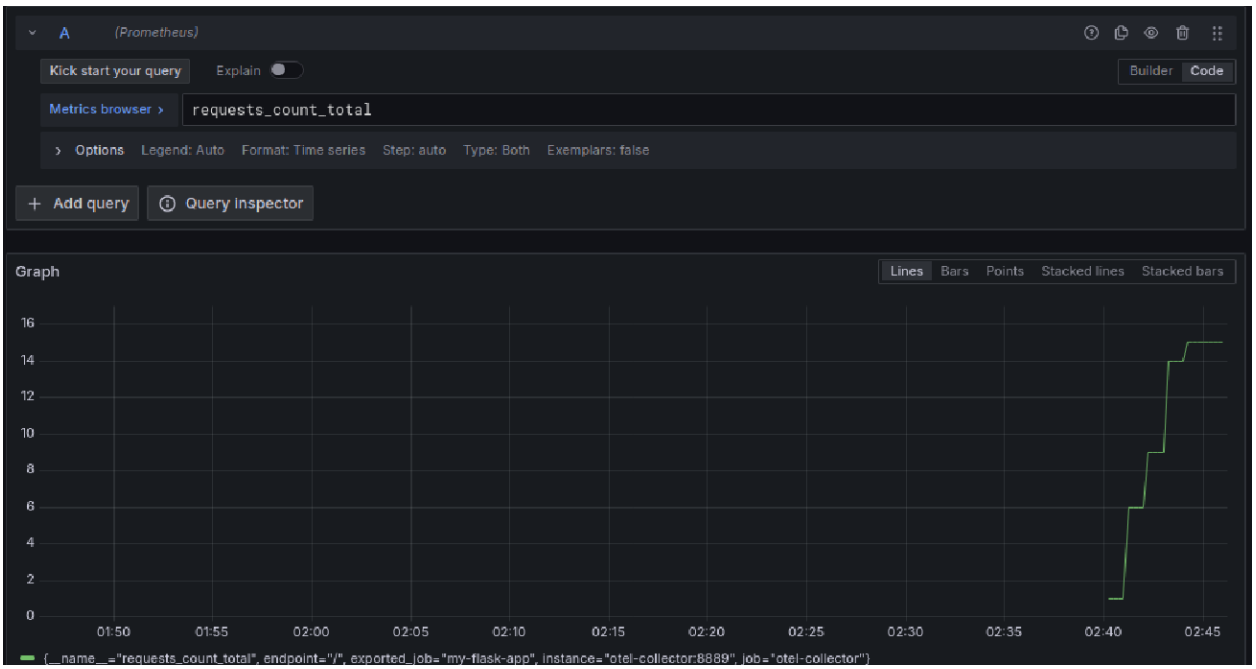


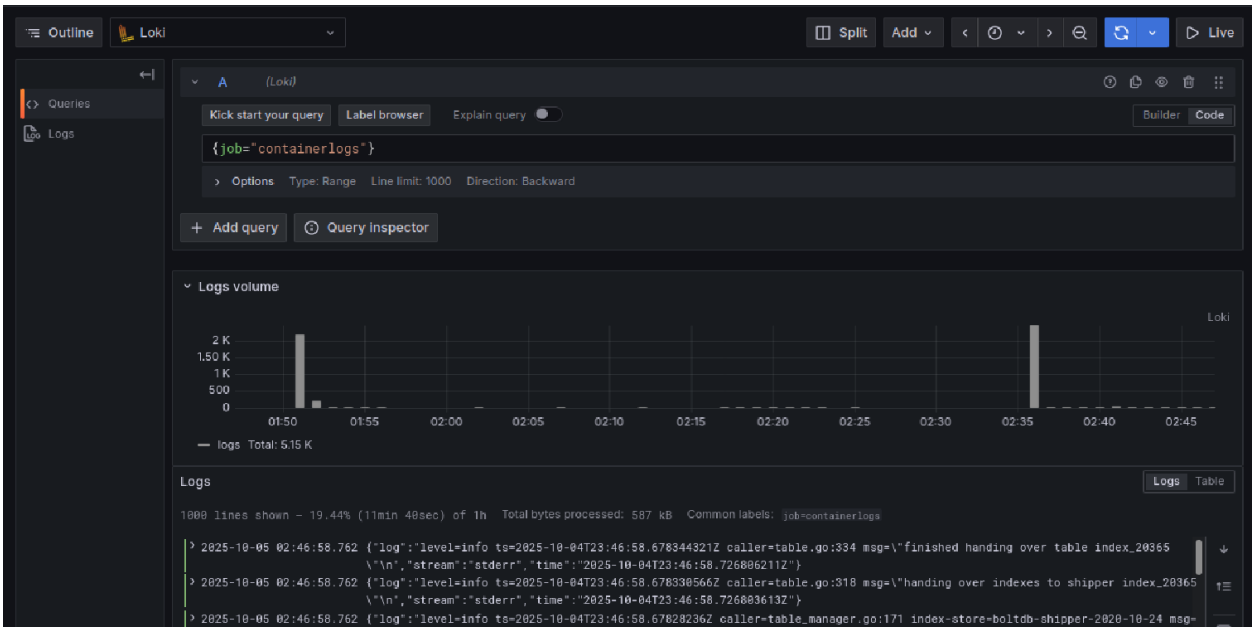


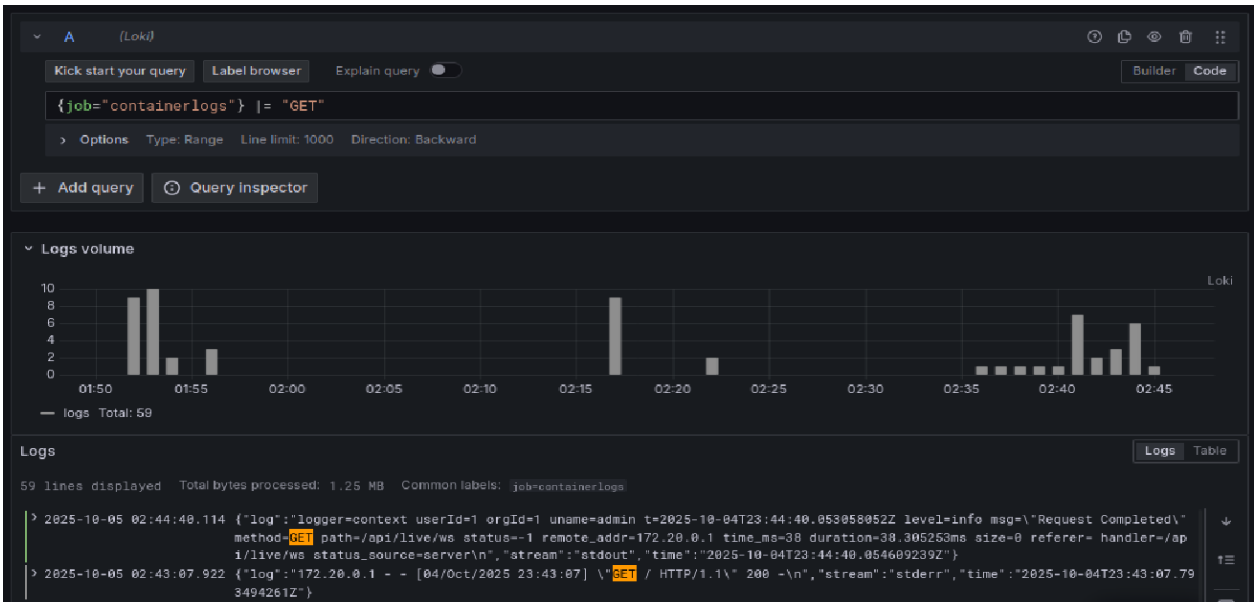

Search
Upload
Service (1)
my-flask-app
Operation (2)
all
Tags
http.status_code=200 error=true
Lookback
Last 12 Hours
Max Duration
e.g. 1.2s, 100...
Min Duration
e.g. 1.2s, 100...
Limit Results
20
Find Traces
4.5ms
3ms
1.5ms
0µs
Duration
02:40:06 am
02:40:10 am
Time
6 Traces
Sort:
Most Recent
Download Results
Deep Dependency Graph
Compare traces by selecting result items
my-flask-app: GET / 10c0917
2.39ms
2 Spans
my-flask-app (2)
Today | 2:40:12 am
a few seconds ago
my-flask-app: GET / 002e823
965µs
2 Spans
my-flask-app (2)
Today | 2:40:11 am
a few seconds ago

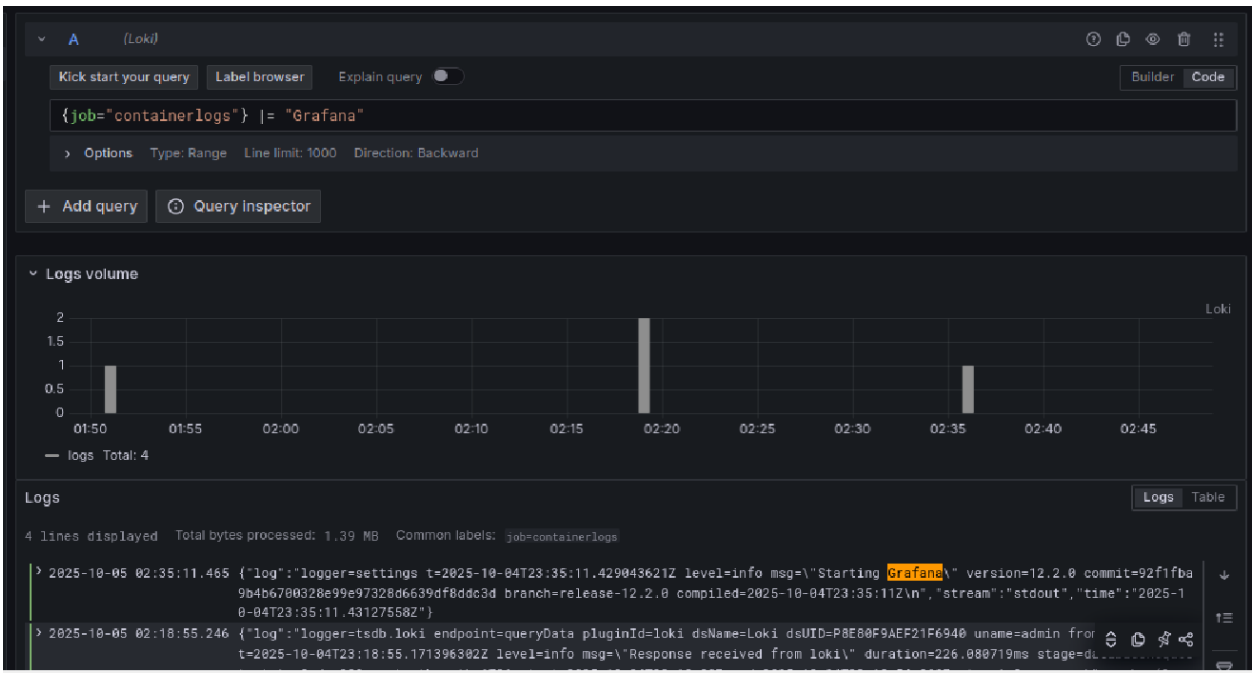

Data sources
View and manage your connected data source connections
+ Add new data source
Search by name or type
Sort by A–Z
Jaeger
Jaeger | http://jaeger:16686
Build a dashboard
Explore
Loki
Loki | http://loki:3100
Build a dashboard
Explore
Prometheus
Prometheus | http://prometheus:9090 | default
Build a dashboard
Explore

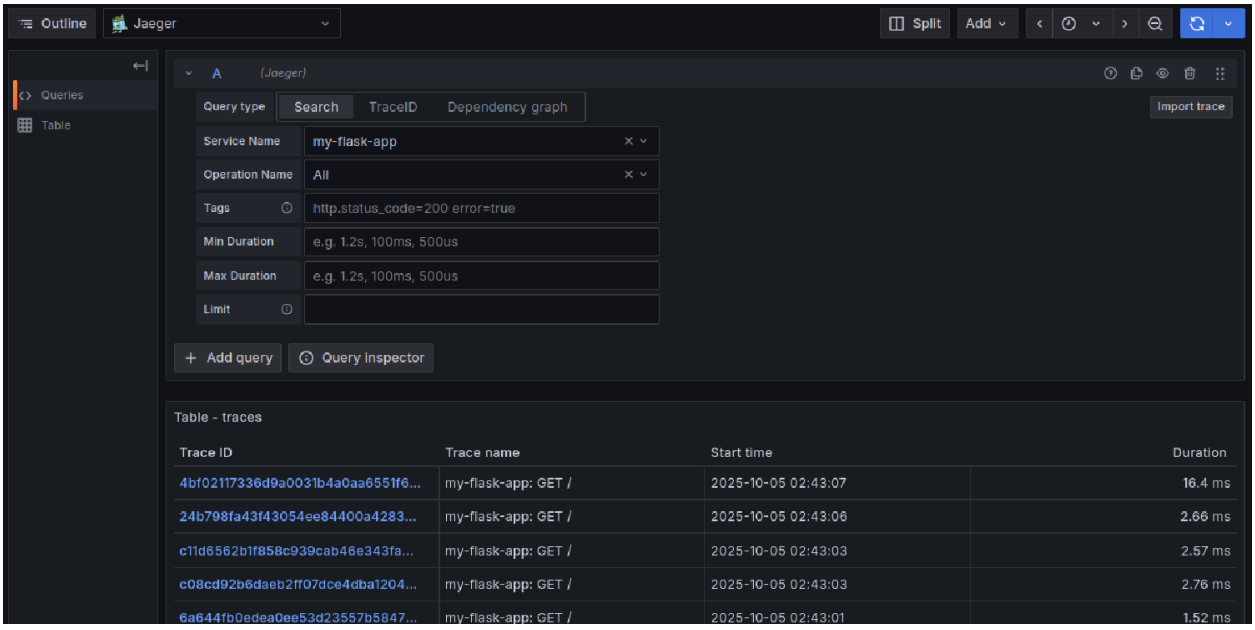

Outline
Prometheus
Split
Add
Queries
Graph
A (Prometheus)
Kick start your query
Explain
Builder
Code
Metric
Label filters
scrape_duration_seconds
Select label
Select value
+ Operations
scrape_duration_seconds
Options Legend: Auto Format: Time series Step: auto Type: Range Exemplars: false
+ Add query
Query inspector
Graph
Lines
Bars
Points
Stacked lines
Stacked bars
0.2
0.175
0.15
0.125
0.1
0.075
0.05
0.025
0
01:55
02:00
02:05
02:10
02:15
02:20
02:25
02:30
02:35
02:40
02:45
{__name__="scrape_duration_seconds", instance="otel-collector:8889", job="otel-collector"}


A (Prometheus)
Kick start your query
Explain
Builder Code
Metrics browser ›
requests_count_total
Options Legend: Auto Format: Time series Step: auto Type: Both Exemplars: false
Add query
Query inspector
Graph
Lines Bars Points Stacked lines Stacked bars
16
14
12
10
8
6
4
2
0
01:50 01:55 02:00 02:05 02:10 02:15 02:20 02:25 02:30 02:35 02:40 02:45
{__name__="requests_count_total", endpoint="/", exported_job="my-flask-app", instance="otel-collector:8889", job="otel-collector"}


Outline
Loki
Split
Add
Live
Queries
Logs
A (Loki)
Kick start your query
Label browser
Explain query
Builder Code
{job="containerlogs"}
Options Type: Range Line limit: 1000 Direction: Backward
Add query
Query inspector
Logs volume
Loki
2 K
1.50 K
1 K
500
0
01:50 01:55 02:00 02:05 02:10 02:15 02:20 02:25 02:30 02:35 02:40 02:45
logs Total: 5.15 K
Logs
Logs Table
1000 lines shown – 19.44% (11min 40sec) of 1h Total bytes processed: 587 kB Common labels: job=containerlogs
2025-10-05 02:46:58.762 {"log":"level=info ts=2025-10-04T23:46:58.678344321Z caller=table.go:334 msg=\"finished handing over table index_20365 \"\n","stream":"stderr","time":"2025-10-04T23:46:58.726806211Z"}
2025-10-05 02:46:58.762 {"log":"level=info ts=2025-10-04T23:46:58.678330566Z caller=table.go:318 msg=\"handing over indexes to shipper index_20365 \"\n","stream":"stderr","time":"2025-10-04T23:46:58.726803613Z"}
2025-10-05 02:46:58.762 {"log":"level=info ts=2025-10-04T23:46:58.67828236Z caller=table_manager.go:171 index-store=boltdb-shipper-2020-10-24 msg=


A (Loki)
Kick start your query
Label browser
Explain query
Builder Code
{job="containerlogs"} |= "GET"
Options Type: Range Line limit: 1000 Direction: Backward
Add query
Query inspector
Logs volume
Loki
10
8
6
4
2
0
01:50 01:55 02:00 02:05 02:10 02:15 02:20 02:25 02:30 02:35 02:40 02:45
logs Total: 59
Logs
Logs Table
59 lines displayed Total bytes processed: 1.25 MB Common labels: job=containerlogs
2025-10-05 02:44:40.114 {"log":"logger=context userId=1 orgId=1 uname=admin t=2025-10-04T23:44:40.053058052Z level=info msg=\"Request Completed\" method=GET path=/api/live/ws status=-1 remote_addr=172.20.0.1 time_ms=38 duration=38.305253ms size=0 referer= handler=/api/live/ws status_source=server\n","stream":"stdout","time":"2025-10-04T23:44:40.054609239Z"}
2025-10-05 02:43:07.922 {"log":"172.20.0.1 - - [04/Oct/2025 23:43:07] \"GET / HTTP/1.1\" 200 -\n","stream":"stderr","time":"2025-10-04T23:43:07.793494261Z"}


A (Loki)
Kick start your query
Label browser
Explain query
Builder
Code
{job="containerlogs"} |= "Grafana"
Options Type: Range Line limit: 1000 Direction: Backward
Add query
Query inspector
Logs volume
Loki
2
1.5
1
0.5
0
01:50
01:55
02:00
02:05
02:10
02:15
02:20
02:25
02:30
02:35
02:40
02:45
logs Total: 4
Logs
Logs
Table
4 lines displayed
Total bytes processed: 1.39 MB
Common labels: job=containerlogs
2025-10-05 02:35:11.465 {"log":"logger=settings t=2025-10-04T23:35:11.429043621Z level=info msg=\"Starting Grafana\" version=12.2.0 commit=92f1fba
9b4b6700328e99e97328d6639df8ddc3d branch=release-12.2.0 compiled=2025-10-04T23:35:11Z\n","stream":"stdout","time":"2025-1
0-04T23:35:11.43127558Z"}
2025-10-05 02:18:55.246 {"log":"logger=tsdb.loki endpoint=queryData pluginId=loki dsName=Loki dsUID=P8E80F9AEF21F6940 uname=admin fro
t=2025-10-04T23:18:55.171396302Z level=info msg=\"Response received from loki\" duration=226.080719ms stage=d


Outline
Jaeger
Split
Add
Queries
Table
A (Jaeger)
Query type
Search
TraceID
Dependency graph
Import trace
Service Name
my-flask-app
Operation Name
All
Tags
http.status_code=200 error=true
Min Duration
e.g. 1.2s, 100ms, 500us
Max Duration
e.g. 1.2s, 100ms, 500us
Limit
Add query
Query inspector
Table - traces
Trace ID
Trace name
Start time
Duration
4bf02117336d9a0031b4a0aa6551f6...
my-flask-app: GET /
2025-10-05 02:43:07
16.4 ms
24b798fa43f43054ee84400a4283...
my-flask-app: GET /
2025-10-05 02:43:06
2.66 ms
c11d6562b1f858c939cab46e343fa...
my-flask-app: GET /
2025-10-05 02:43:03
2.57 ms
c08cd92b6daeb2ff07dce4dba1204...
my-flask-app: GET /
2025-10-05 02:43:03
2.76 ms
6a644fb0edea0ee53d23557b5847...
my-flask-app: GET /
2025-10-05 02:43:01
1.52 ms

Контрольные вопросы:

1. OpenTelemetry — это открытый стандарт и набор инструментов для работы с телеметрическими данными (метрики, логи и трассировки). Атрибуты — это пары ключ-значение, которые описывают элемент, например, событие. События — это аннотации на трассировках, обозначающие значимые моменты времени. Контекст — это информация, которая передается между компонентами системы для связи трассировок. Логи — записи о событиях в системе. Трассировки показывают путь запроса через различные сервисы. Показатели (метрики) — это числовые измерения, собранные с течением времени, например, загрузка ЦП.

2. Наблюдаемость — это способность понимать внутреннее состояние системы на основе её внешних данных, таких как метрики, логи и трассировки. Надежность — это способность системы выполнять требуемые функции в заданных условиях. Показатели являются ключевым элементом наблюдаемости, предоставляя данные о состоянии системы для анализа.

3. Трассировка отслеживает путь запроса. Распределенная трассировка делает то же самое, но через несколько микросервисов, что усложняет сбор данных и их связывание.

4. Сигналы в OpenTelemetry: трассировки предоставляют представление о пути запроса через распределенную систему; метрики — это числовые данные, измеряемые с течением времени, такие как количество запросов в секунду или использование памяти; логи — это текстовые записи о событиях.

5. Автоматические инструменты не требуют изменения исходного кода и внедряются на уровне среды выполнения. Ручные инструменты требуют от разработчика явного вызова API OpenTelemetry в коде для создания спанов, метрик или логов. Библиотеки — это специфичные для языков и фреймворков пакеты, которые упрощают как ручное, так и автоматическое инструментирование.

6. Спецификация определяет стандарты для всех телеметрических данных и API. Библиотеки инструментальных средств — это код для конкретных языков программирования, который используется для сбора телеметрии. Сборщики — это прокси-сервисы, которые получают, обрабатывают и экспортируют телеметрические данные, снимая эту нагрузку с приложений. Экспортеры отвечают за отправку данных в различные бэкенды для хранения и анализа, такие как Prometheus или Jaeger. Измерительные инструменты — это API для захвата метрик.

7. Ресурс в OpenTelemetry представляет собой объект, генерирующий телеметрические данные. Например, приложение, сервис, хост или контейнер.

8. Экспортеры OpenTelemetry — это компоненты, которые преобразуют и отправляют собранные телеметрические данные в системы хранения и анализа, такие как Prometheus или Jaeger.

9. Prometheus — это система мониторинга, которая работает по модели "pull", периодически опрашивая сервисы для сбора метрик в виде временных рядов.

10. Ключевые концепции Prometheus: Метрики — числовые данные, собираемые с течением времени. Временной ряд — последовательность точек данных, индексированных по времени, с уникальной комбинацией имени метрики и меток. Метки — это пары ключ-значение, которые позволяют фильтровать и агрегировать метрики. Scraping — процесс опроса целевых систем для сбора метрик. PromQL — мощный язык запросов для извлечения и анализа данных. Оповещения — правила, которые определяют условия для отправки уведомлений о проблемах.

11. Counter — это монотонно возрастающее значение. Gauge — это числовое значение, которое может как увеличиваться, так и уменьшаться. Histogram — отслеживает распределение наблюдений по настраиваемым корзинам. Summary также измеряет распределение, но предоставляет квантили на стороне клиента.

12. Grafana запрашивает данные у источников, используя их родной язык запросов, например, PromQL для Prometheus или SQL для БД. После получения данных Grafana может их преобразовать перед визуализацией.

13. Источники данных Grafana — это плагины для подключения к различным системам, таким как Prometheus, Loki, Elasticsearch или SQL-базам, для получения данных.

14. Панели мониторинга Grafana — это настраиваемые экраны, состоящие из панелей, которые визуализируют данные из подключенных источников.

15. Процесс обработки журналов Grafana Loki — агент отправляет логи с метками в Loki, который индексирует только метки, а не весь текст, что делает его быстрым и экономичным.

16. Варианты сборки журналов для Grafana Loki — для сбора логов используются специальные агенты, самые популярные из которых — Promtail, Fluentd, Fluent Bit и OpenTelemetry Collector.