14 ДЗ - Метрики и мониторинг. Логирование.

Собираем Dockerfile и docker-compose.

Dockerfile:

```
FROM clickhouse/clickhouse-server:25.2.1
MAINTAINER Maksim Kulikov <max.uoles@rambler.ru>

RUN apt-get update -y --fix-missing
RUN DEBIAN_FRONTEND=noninteractive apt-get -yq upgrade
RUN apt-get install nano mc python3 pip kafkacat -y
RUN pip install clickhouse_driver

COPY clickhouse/prometheus.xml /etc/clickhouse-server/config.d/prometheus.xml

EXPOSE 8123 9000 9363

ENTRYPOINT ["/entrypoint.sh"]
```

Собираем docker-compose.yml для запуска приложений:

docker-compose-prometheus.yml

```
version: "3.6"

services:
  clickhouse-server:
    container_name: clickhouse
    image: uoles/clickhouse:25.2.1
    hostname: clickhouse-server
    build:
      context: .
      dockerfile: clickhouse-25.2.1.Dockerfile
    environment:
      CLICKHOUSE_DB: my_database
      CLICKHOUSE_USER: username
      CLICKHOUSE_DEFAULT_ACCESS_MANAGEMENT: 1
      CLICKHOUSE_PASSWORD: password
    ports:
      - "18123:8123"
      - "19000:9000"
      - "19363:9363"
    ulimits:
      nofile:
        soft: 262144
        hard: 262144
    networks:
      - default

  prometheus:
    image: prom/prometheus:latest
    volumes:
      - ./prometheus:/etc/prometheus/
    container_name: prometheus
    hostname: prometheus
    command:
      - --config.file=/etc/prometheus/prometheus.yml
    ports:
      - 9090:9090
    restart: unless-stopped
    environment:
      TZ: "Europe/Moscow"
    networks:
      - default
    depends_on:
      - clickhouse-server
    links:
      - clickhouse-server

  grafana:
    image: grafana/grafana
    user: root
    depends_on:
      - prometheus
    ports:
      - 3000:3000
    volumes:
      - ./grafana:/var/lib/grafana
      - ./grafana/provisioning/:/etc/grafana/provisioning/
    container_name: grafana
    hostname: grafana
    restart: unless-stopped
    environment:
      TZ: "Europe/Moscow"
    networks:
      - default

networks:
  default:
    ipam:
      driver: default
      config:
        - subnet: 172.28.0.0/16
```

Настройки.

Настройки clickhouse для доступа promethus к метрикам:
docker/clickhouse/prometheus.xml

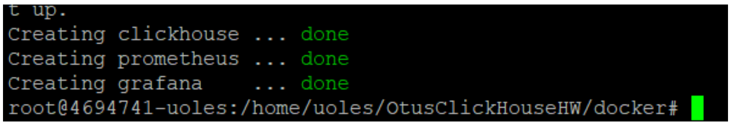
```
<clickhouse>
        <prometheus>
                <endpoint>/metrics</endpoint>
                <port>9363</port>
                <metrincs>true</metrincs>
                <events>true</events>
                <asynchronous_metrics>true</asynchronous_metrics>
                <errors>true</errors>
        </prometheus>
</clickhouse>
```

Настройки promethus для забора метрик из clickhouse:
docker/prometheus/prometheus.yml

```
scrape_configs:
  - job_name: clickhouse_metrics
    static_configs:
      - targets: [ 'clickhouse-server:9363' ]
    scrape_interval: 1m
    scrape_timeout: 30s
    metrics_path: "/metrics"
```

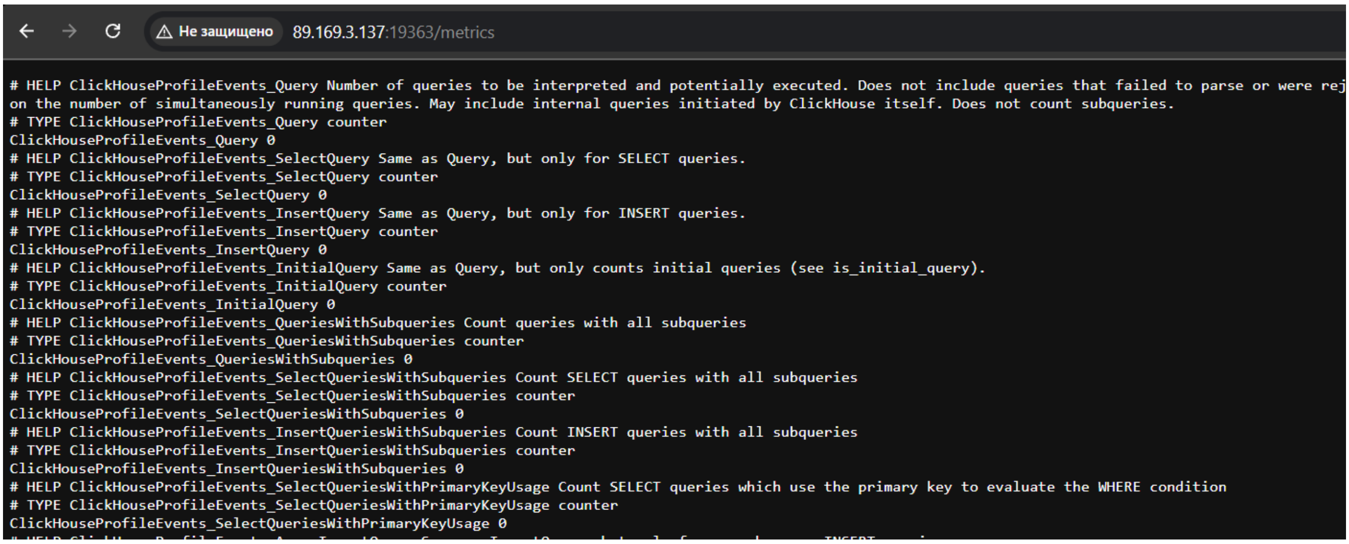
Поднимаем docker-compose.

Используем команду: docker-compose -f docker-compose-prometheus.yml up –d



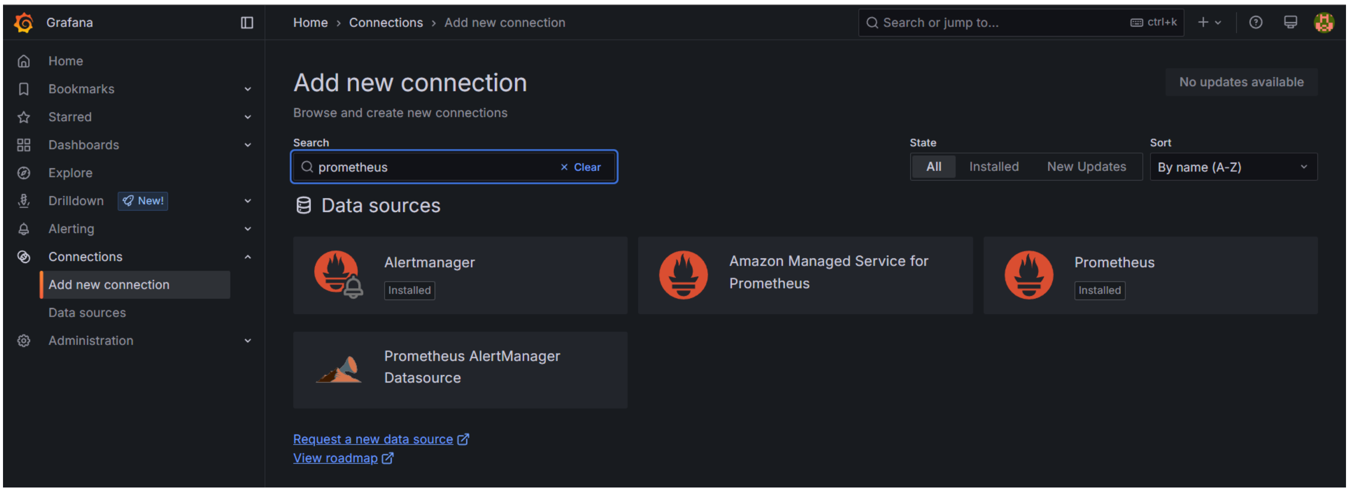
```
t up.
Creating clickhouse ... done
Creating prometheus ... done
Creating grafana    ... done
root@4694741-uoles:/home/uoles/OtusClickHouseHW/docker#
```

Проверяем доступность метрик:



```
# HELP ClickHouseProfileEvents_Query Number of queries to be interpreted and potentially executed. Does not include queries that failed to parse or were rej
on the number of simultaneously running queries. May include internal queries initiated by ClickHouse itself. Does not count subqueries.
# TYPE ClickHouseProfileEvents_Query counter
ClickHouseProfileEvents_Query 0
# HELP ClickHouseProfileEvents_SelectQuery Same as Query, but only for SELECT queries.
# TYPE ClickHouseProfileEvents_SelectQuery counter
ClickHouseProfileEvents_SelectQuery 0
# HELP ClickHouseProfileEvents_InsertQuery Same as Query, but only for INSERT queries.
# TYPE ClickHouseProfileEvents_InsertQuery counter
ClickHouseProfileEvents_InsertQuery 0
# HELP ClickHouseProfileEvents_InitialQuery Same as Query, but only counts initial queries (see is_initial_query).
# TYPE ClickHouseProfileEvents_InitialQuery counter
ClickHouseProfileEvents_InitialQuery 0
# HELP ClickHouseProfileEvents_QueriesWithSubqueries Count queries with all subqueries
# TYPE ClickHouseProfileEvents_QueriesWithSubqueries counter
ClickHouseProfileEvents_QueriesWithSubqueries 0
# HELP ClickHouseProfileEvents_SelectQueriesWithSubqueries Count SELECT queries with all subqueries
# TYPE ClickHouseProfileEvents_SelectQueriesWithSubqueries counter
ClickHouseProfileEvents_SelectQueriesWithSubqueries 0
# HELP ClickHouseProfileEvents_InsertQueriesWithSubqueries Count INSERT queries with all subqueries
# TYPE ClickHouseProfileEvents_InsertQueriesWithSubqueries counter
ClickHouseProfileEvents_InsertQueriesWithSubqueries 0
# HELP ClickHouseProfileEvents_SelectQueriesWithPrimaryKeyUsage Count SELECT queries which use the primary key to evaluate the WHERE condition
# TYPE ClickHouseProfileEvents_SelectQueriesWithPrimaryKeyUsage counter
ClickHouseProfileEvents_SelectQueriesWithPrimaryKeyUsage 0
```

Настраиваем соединение с prometheus и дашборд в grafana.



Add new connection

Browse and create new connections

Search: prometheus

State: All

Sort: By name (A-Z)

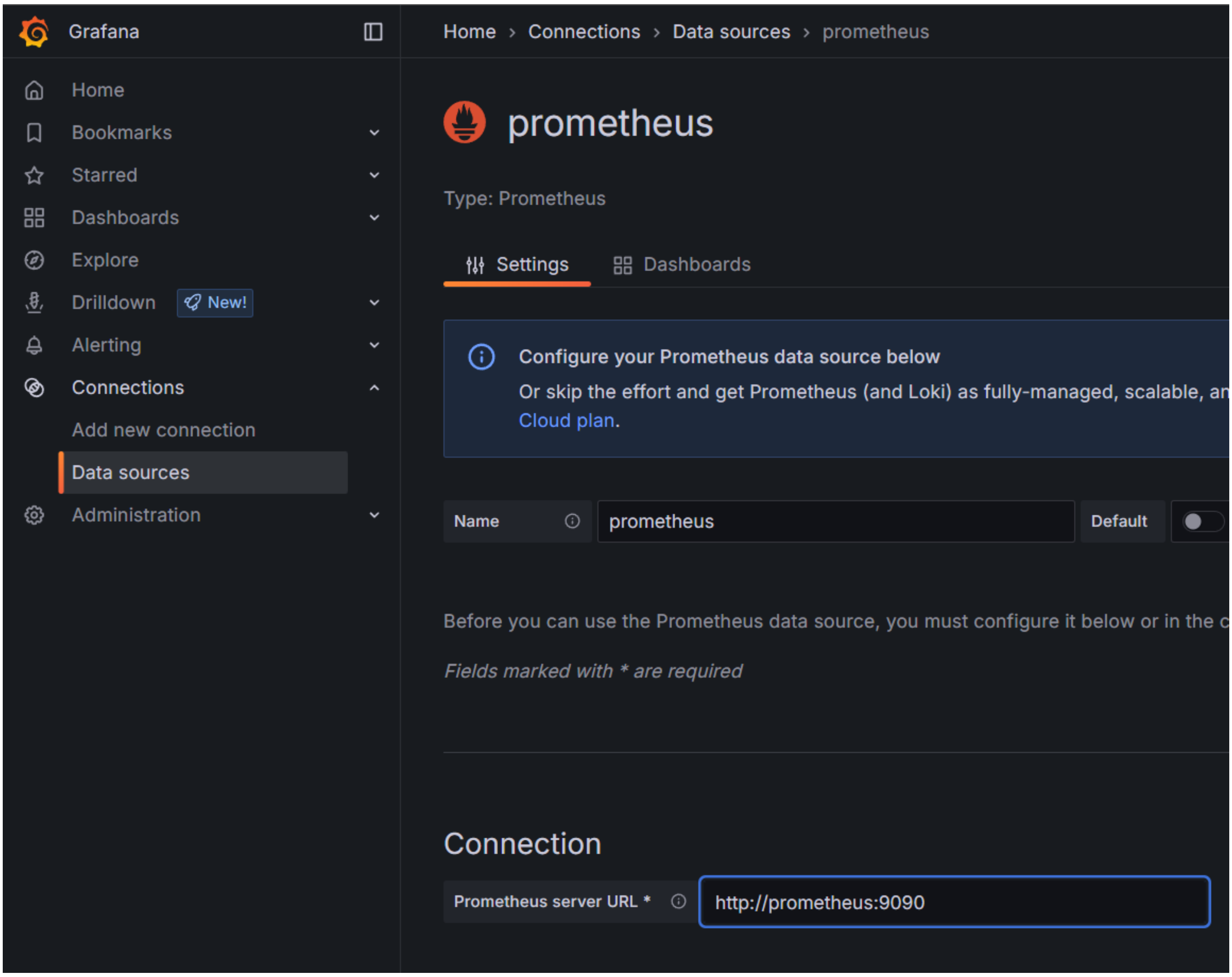
Data sources

Alertmanager

Amazon Managed Service for Prometheus

Prometheus

Prometheus AlertManager Datasource



prometheus

Type: Prometheus

Configure your Prometheus data source below
Or skip the effort and get Prometheus (and Loki) as fully-managed, scalable, an
Cloud plan.

Name: prometheus
Default

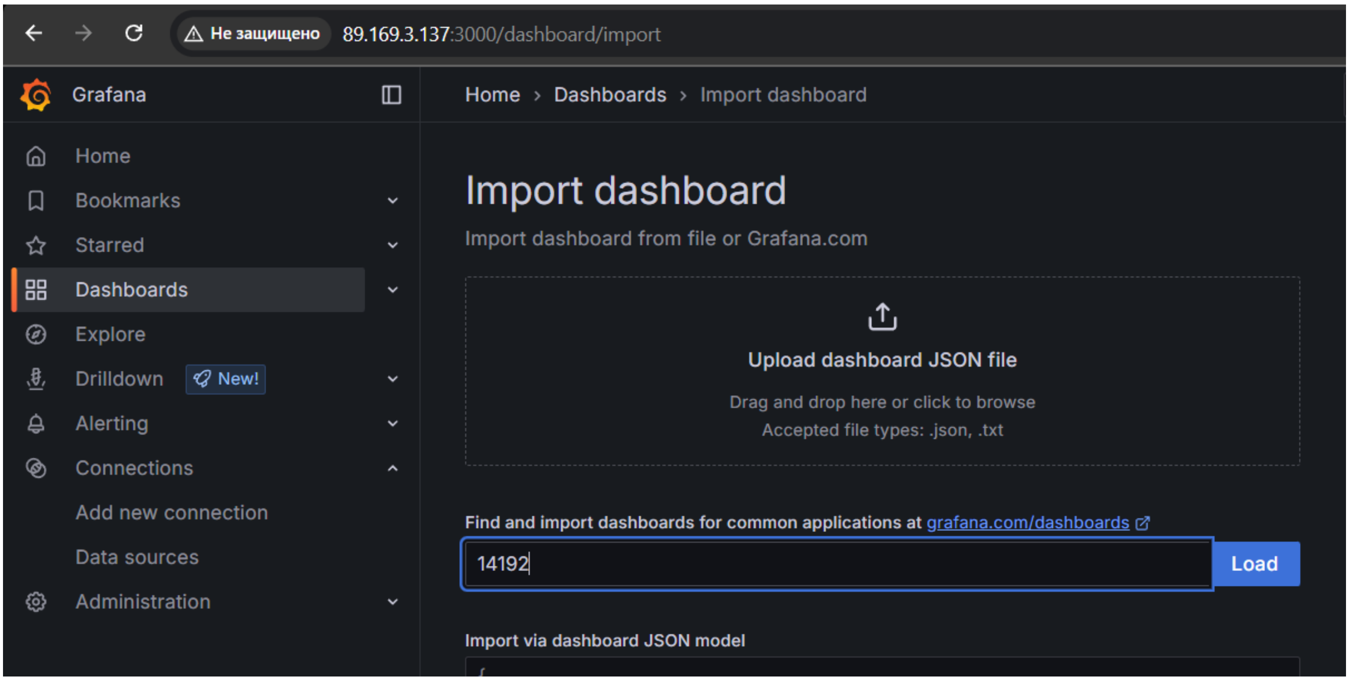
Before you can use the Prometheus data source, you must configure it below or in the c

*Fields marked with * are required*

Connection

Prometheus server URL *: http://prometheus:9090

Импортируем дашборд - https://grafana.com/grafana/dashboards/14192-clickhouse/



Import dashboard

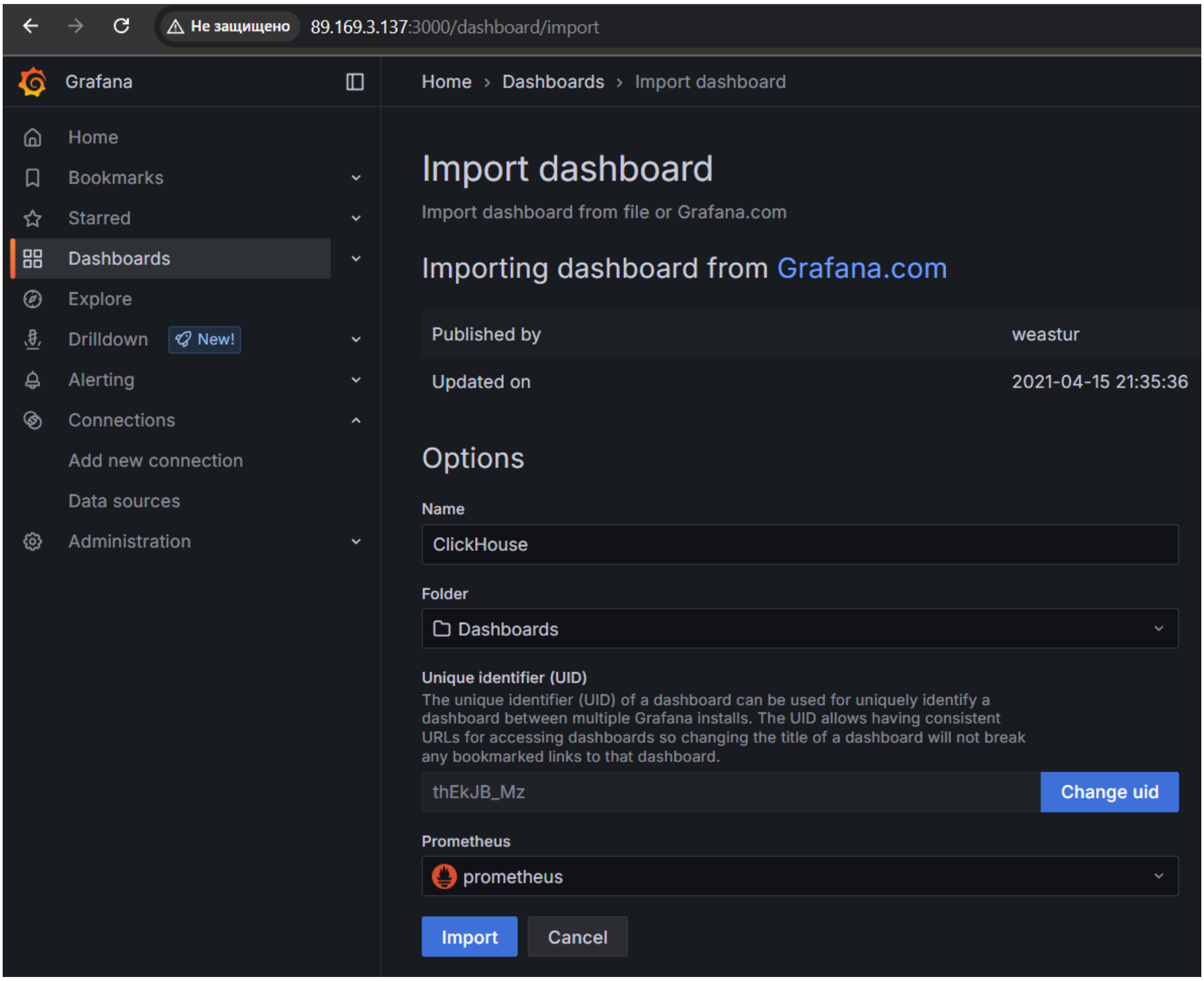
Import dashboard from file or Grafana.com

Upload dashboard JSON file
Drag and drop here or click to browse
Accepted file types: .json, .txt

Find and import dashboards for common applications at grafana.com/dashboards

14192

Import via dashboard JSON model



Import dashboard

Import dashboard from file or Grafana.com

Importing dashboard from Grafana.com

Published by	weastur
Updated on	2021-04-15 21:35:36

Options

Name: ClickHouse

Folder: Dashboards

Unique identifier (UID)

The unique identifier (UID) of a dashboard can be used for uniquely identify a dashboard between multiple Grafana installs. The UID allows having consistent URLs for accessing dashboards so changing the title of a dashboard will not break any bookmarked links to that dashboard.

thEkJB_Mz

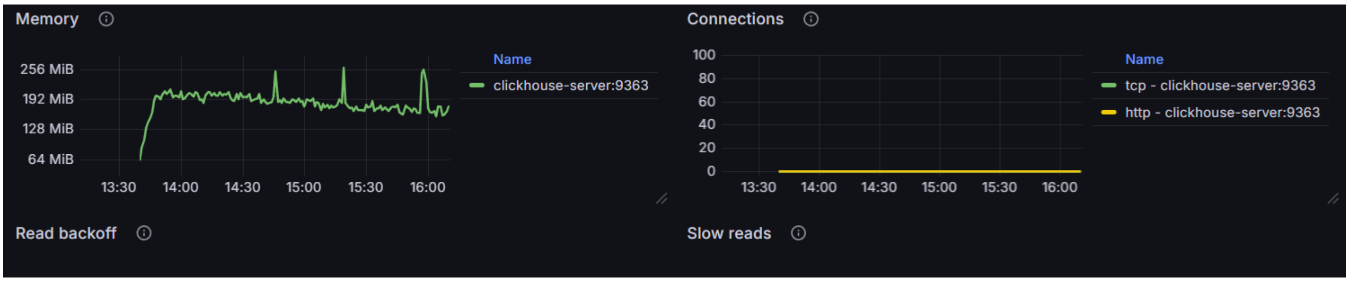
Change uid

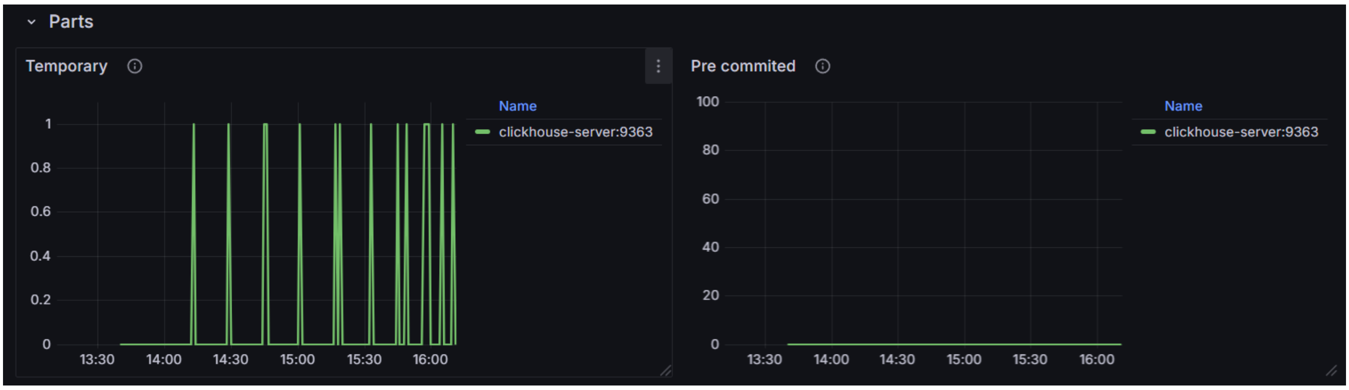
Prometheus: prometheus

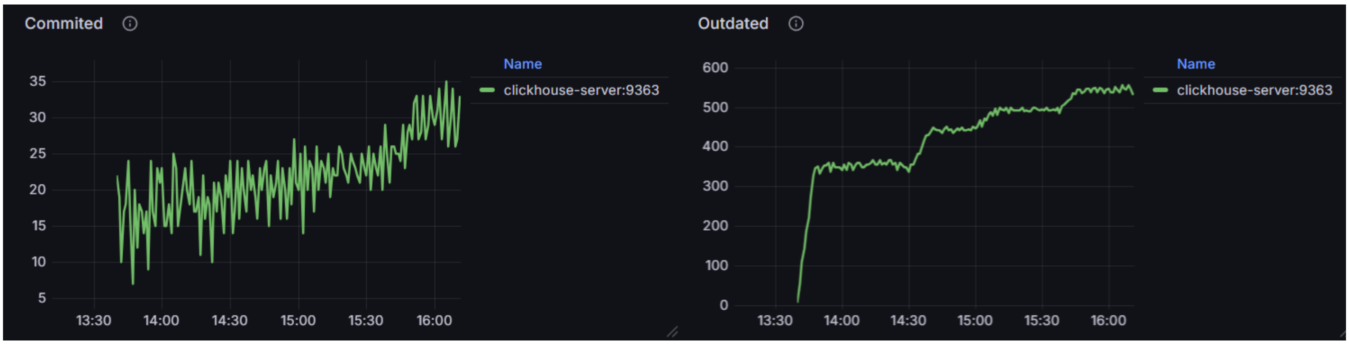
Import

Cancel

Некоторые графики из grafana.






Commited
35
30
25
20
15
10
5
13:30
14:00
14:30
15:00
15:30
16:00
Name
clickhouse-server:9363
Outdated
600
500
400
300
200
100
0
13:30
14:00
14:30
15:00
15:30
16:00
Name
clickhouse-server:9363