**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' ]

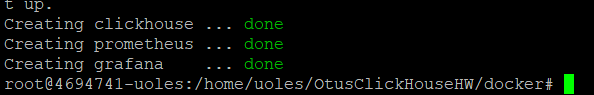
scrape\_interval: 1m

scrape\_timeout: 30s

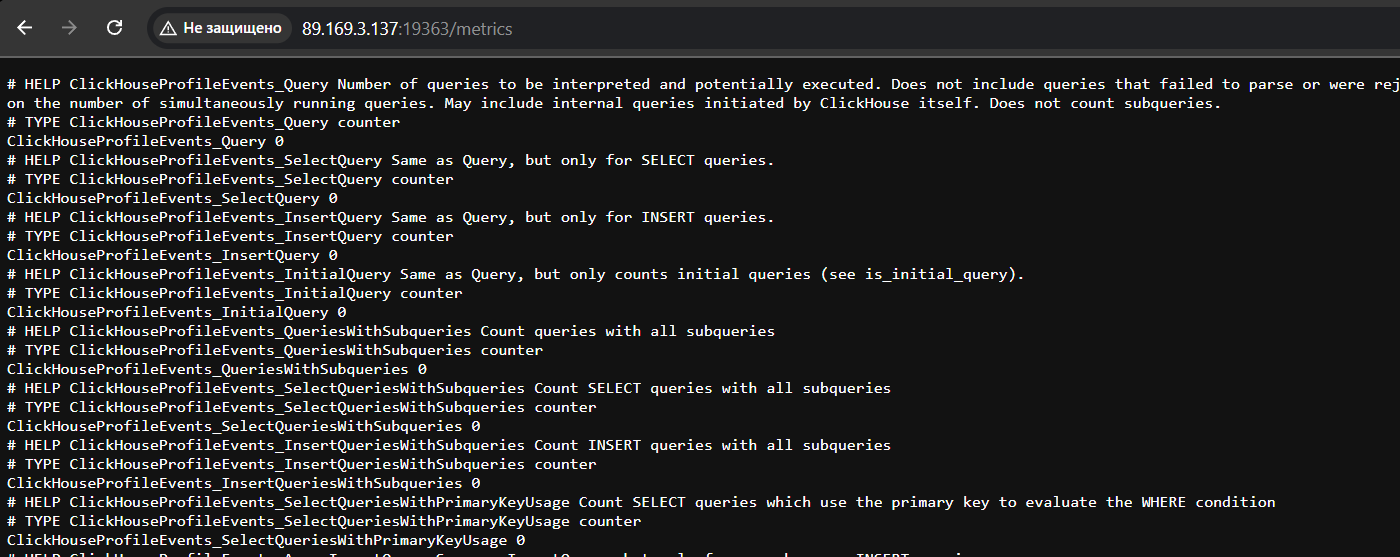
metrics\_path: "/metrics"

**Поднимаем docker-compose.**

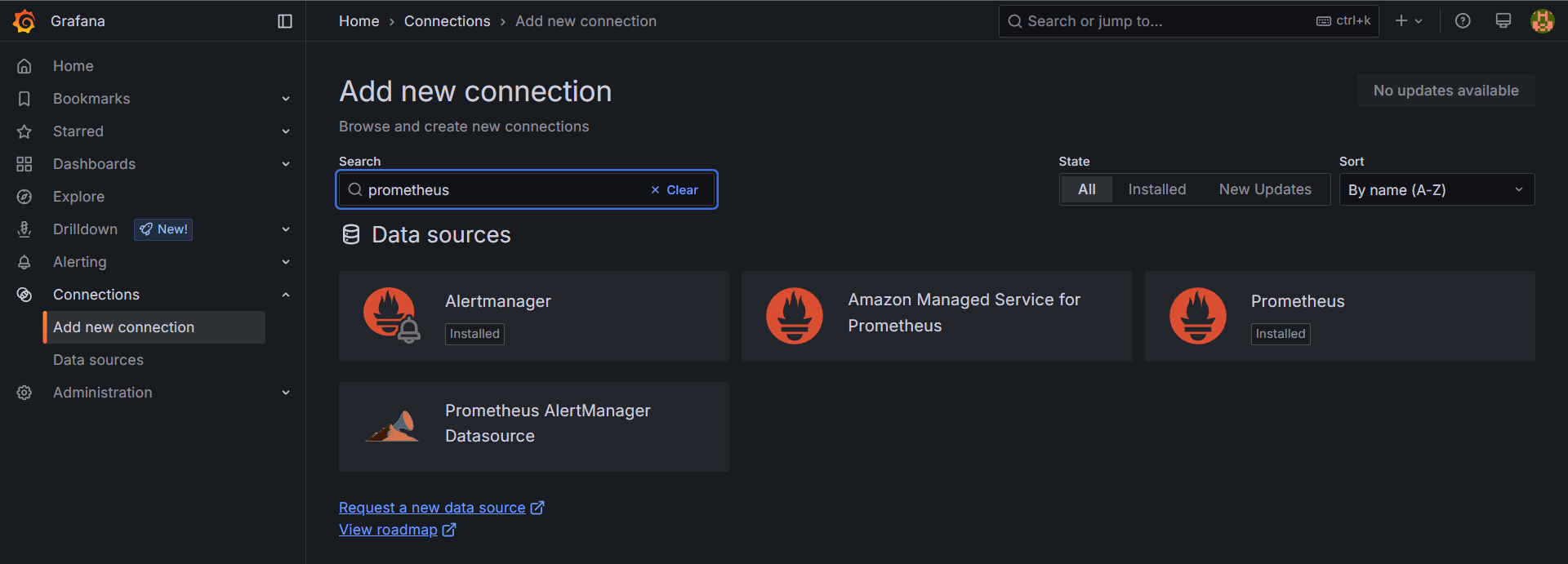
Используем команду: docker-compose -f docker-compose-prometheus.yml up –d

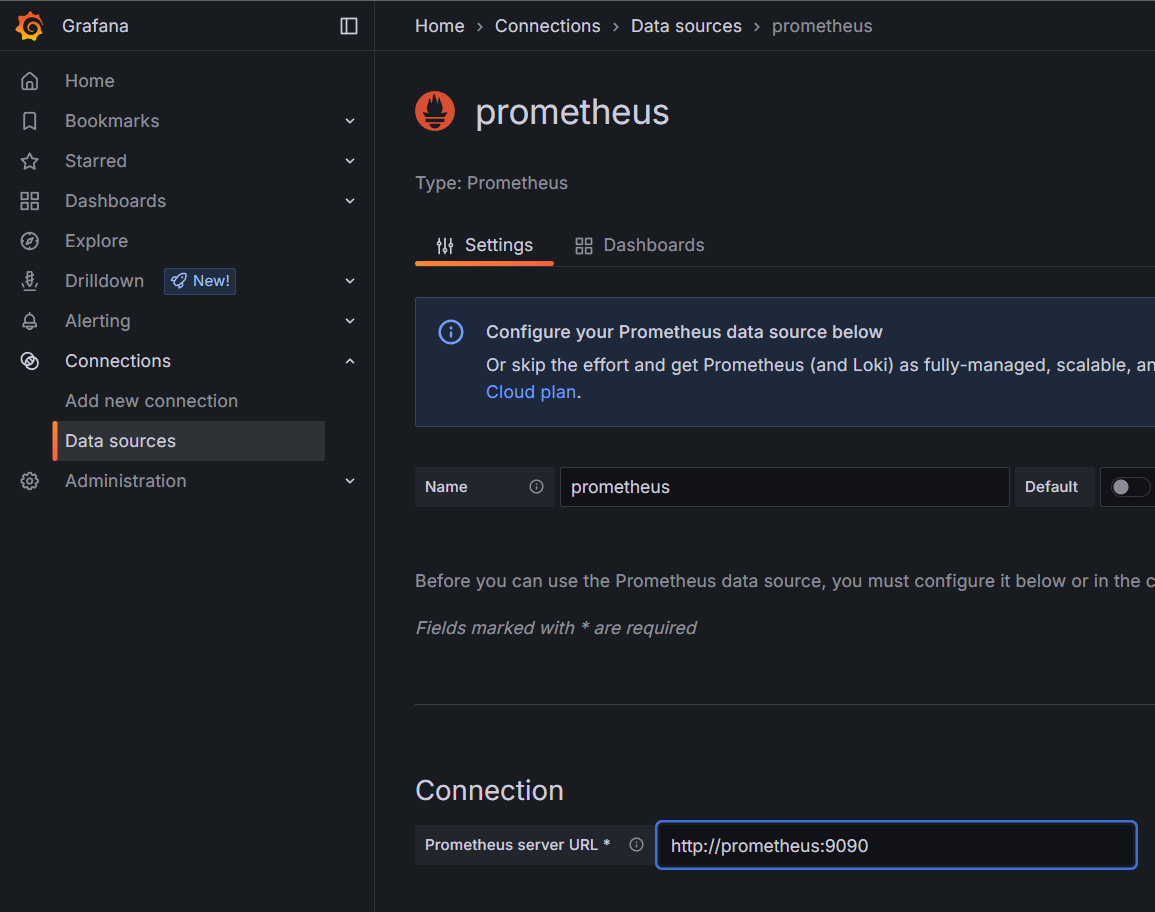


Проверяем доступность метрик:

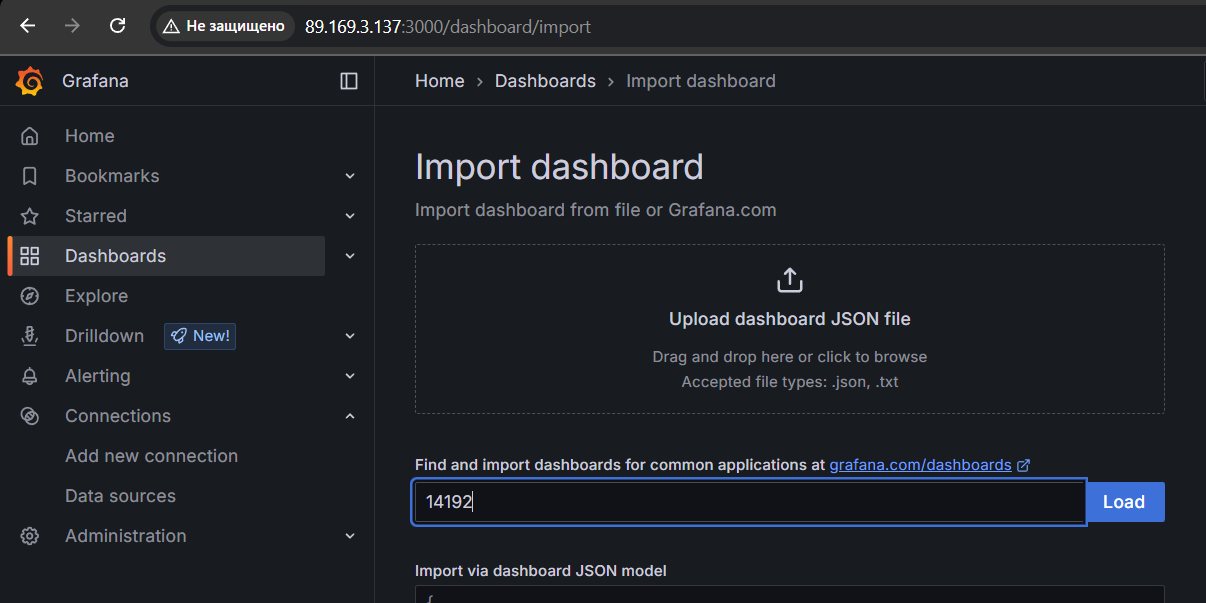


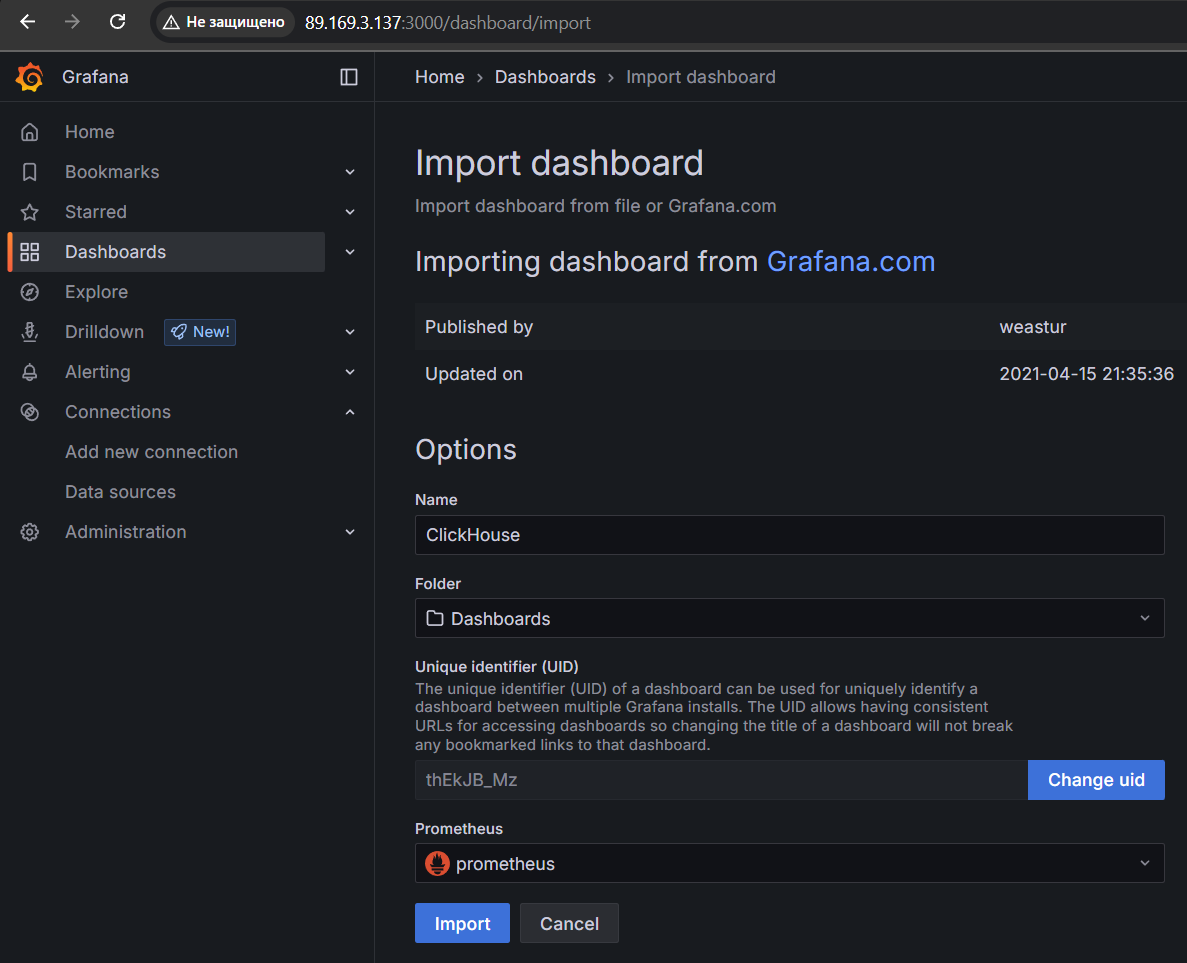
**Настраиваем соединение с prometheus и дашборд в grafana.**





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





**Некоторые графики из grafana.**

