# Л09. Docker compose

Харисов Ринат ПМИ-5

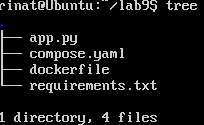
## 1. Собираем веб-приложение (flask+redis)

Установка Docker compose:

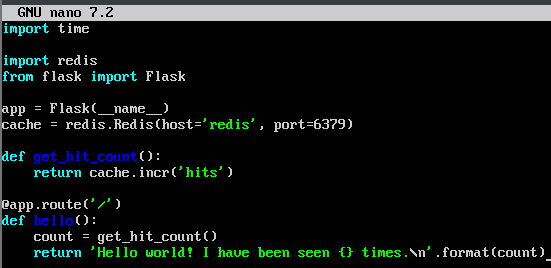
sudo apt update

sudo apt install docker-compose-v2

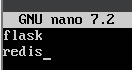
Подготавливаем структуру:



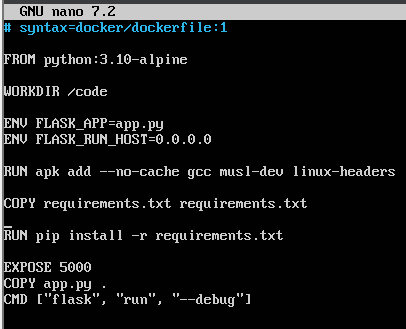
Создаем веб приложение на Python:



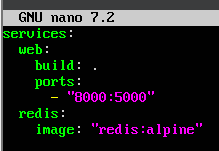
Объявим зависимости в requirements.txt:



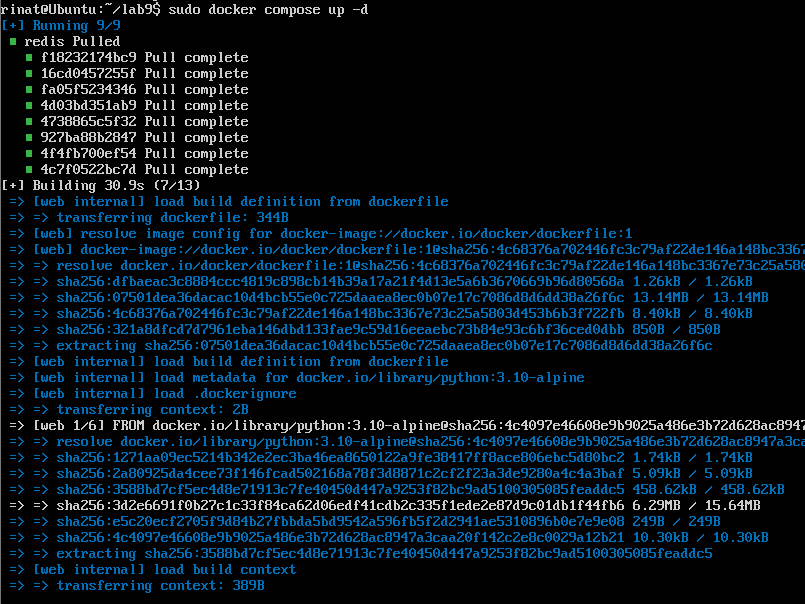
Настраиваем dockerfile для контейнера с flask:

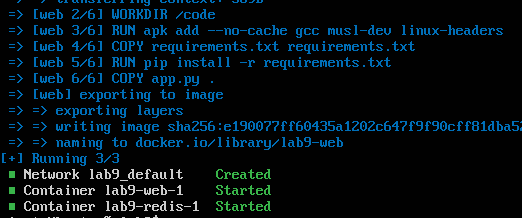


Прописываем docker-compose файл (с запуском flask и redis):

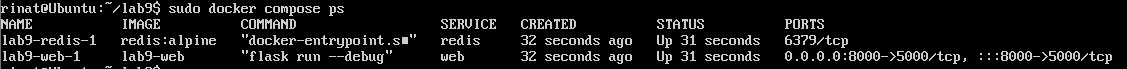


Запускаем docker compose up -d:

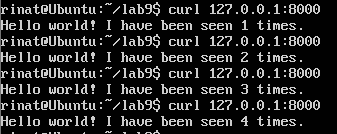




Проверяем контейнеры:

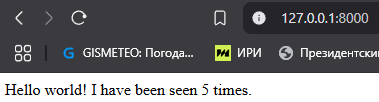


Проверяем работу:



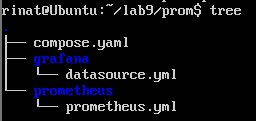
Пробрасываем порты и проверяем в браузере:



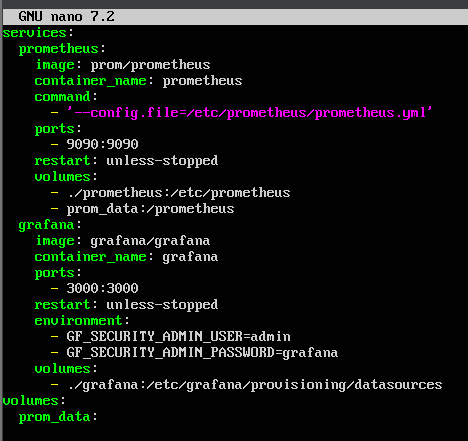


## 2. Собираем мониторинг (prometheus+grafana)

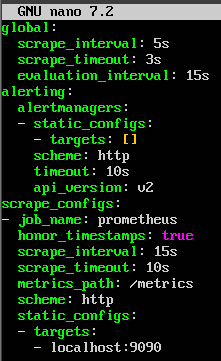
Подготовим структуру:



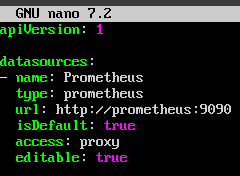
Изменим compose.yaml:



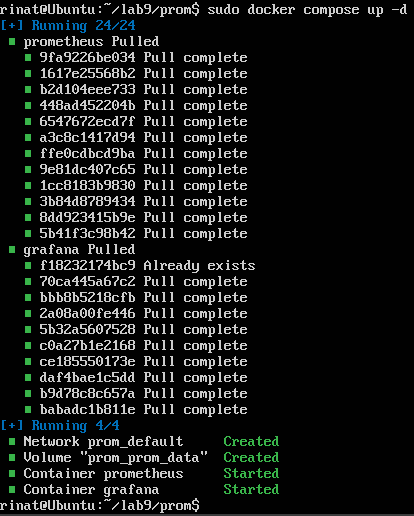
Изменяем конфиг для prometheus - prometheus.yml:



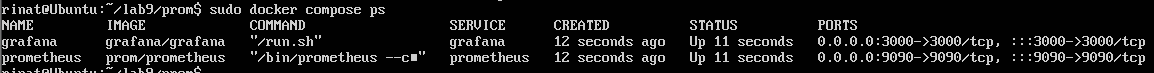
Изменяем конфиг для grafana - datasource.yml:



Запускаем docker-compose:



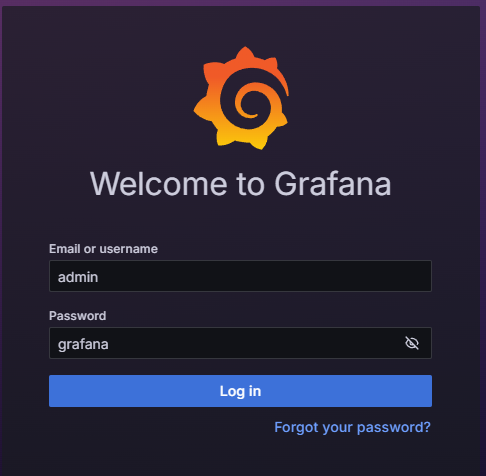
Проверяем работу:



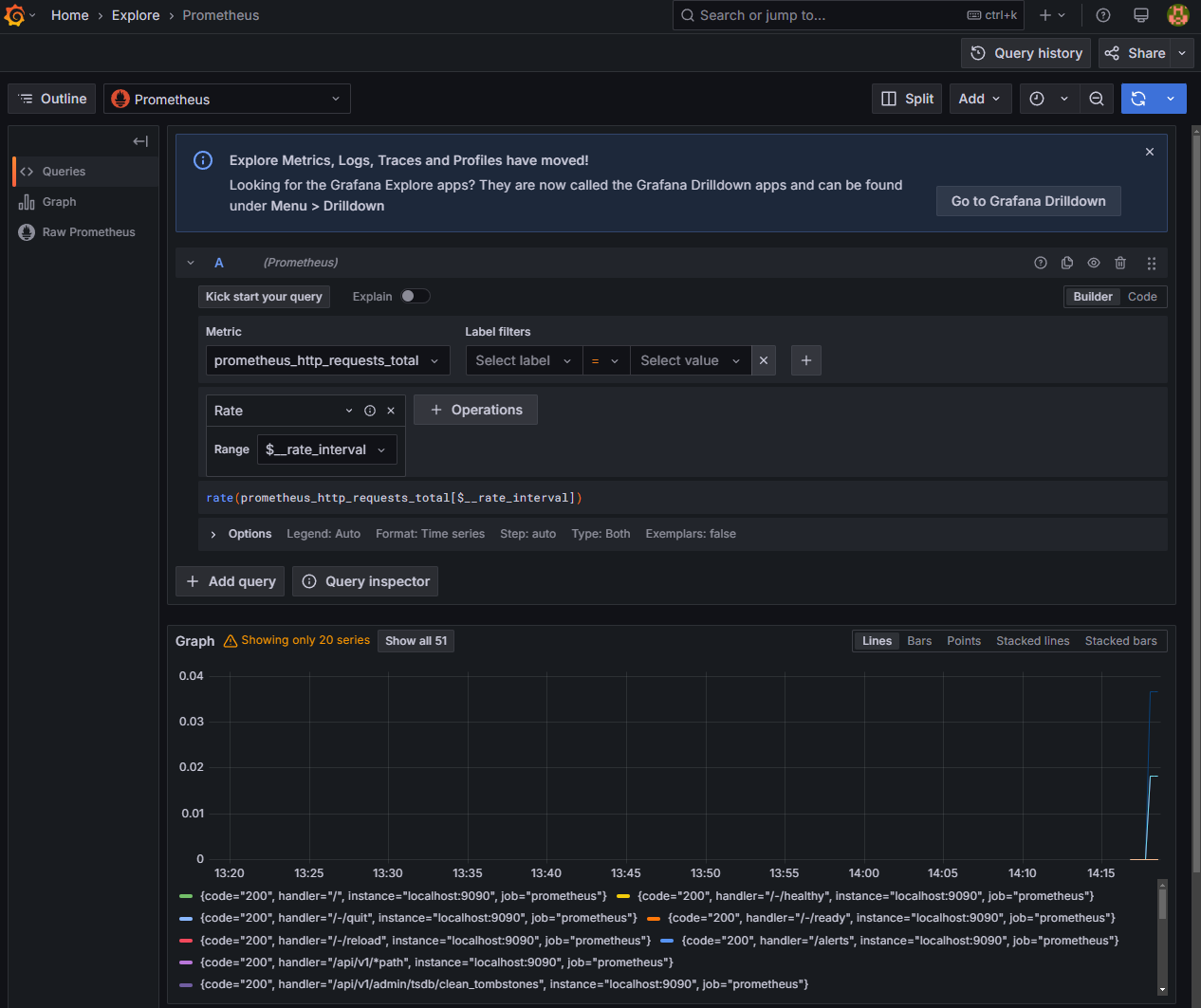
Пробрасываем порт:



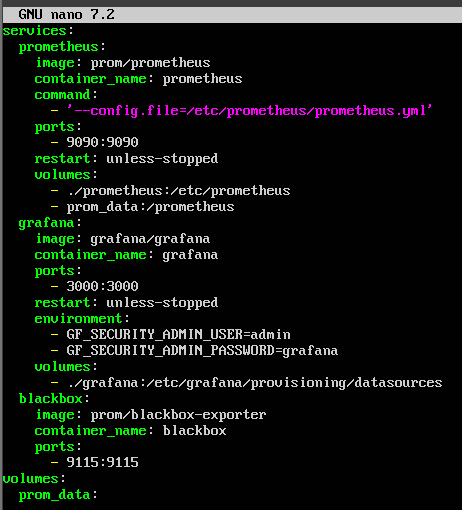
Заходим в Grafana:



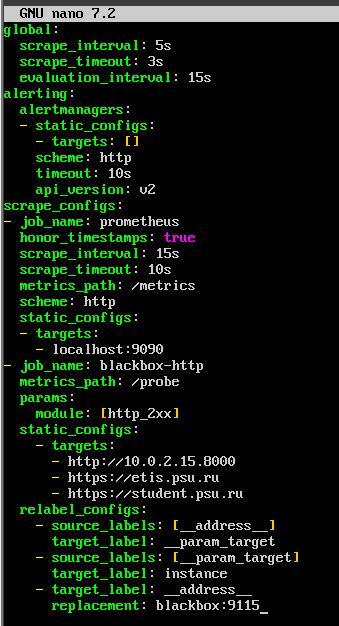
Смотрим метрики Prometheus:



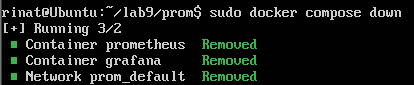
Добавляем в docker-compose blackbox:

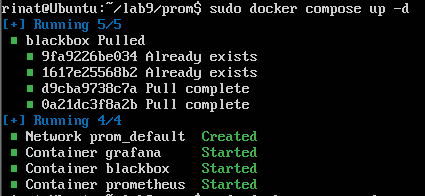


Изменяем конфиг prometheus и добавляем мониторинг Flask приложения:

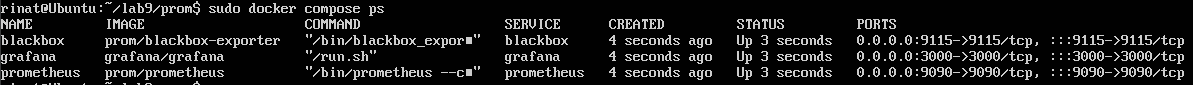


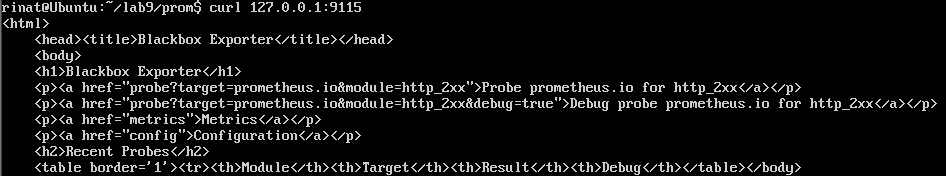
Перезапускаем стек:



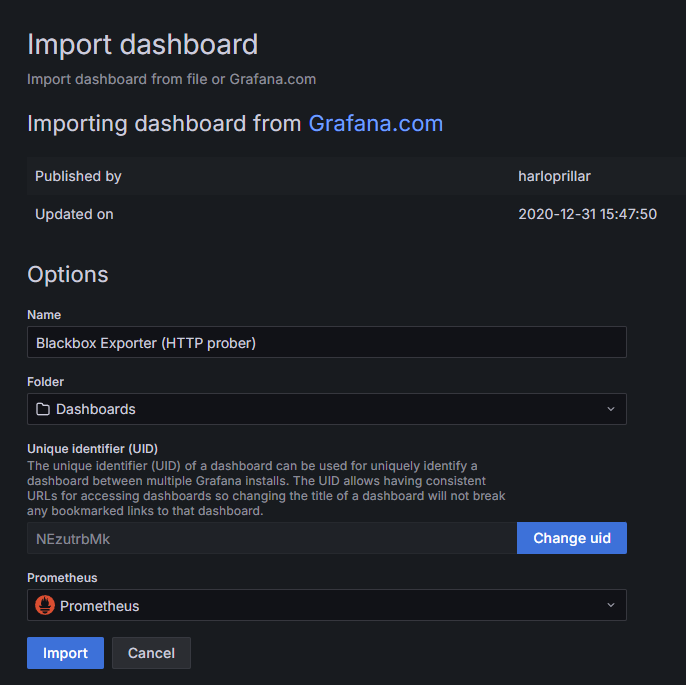


Проверяем работу:

****

****

Создаем dashboard:

****

Метрики успешно отображаются:

