### **Задание 1**

Напишите ответ в свободной форме, не больше одного абзаца текста.

Установите Docker Compose и опишите, для чего он нужен и как может улучшить вашу жизнь.

### **Решение 1**

Docker Compose - инструмент для управления несколькими контейнерами. Он входит в состав пакета docker и позволяет управлять комплексным приложением, которое состоит из более 1 контейнера при помощи файла конфигурации “yaml”, в котором описываются контейнеры и их настройки.

### **Задание 2**

Выполните действия и приложите текст конфига на этом этапе.

Создайте файл docker-compose.yml и внесите туда первичные настройки:

* version;
* services;
* volumes;
* networks.

При выполнении задания используйте подсеть 10.5.0.0/16. Ваша подсеть должна называться: <ваши фамилия и инициалы>-my-netology-hw. Все приложения из последующих заданий должны находиться в этой конфигурации.

### **Решение 2**

Листинг файла конфига:

version: '3'

services:

netology-app1:

volumes:

networks:

d.dunaev-my-netology-hw:

driver: bridge

ipam:

config:

- subnet: 10.5.0.0/16

gateway: 10.5.0.1

### **Задание 3**

Выполните действия:

1. Создайте конфигурацию docker-compose для Prometheus с именем контейнера <ваши фамилия и инициалы>-netology-prometheus.
2. Добавьте необходимые тома с данными и конфигурацией (конфигурация лежит в репозитории в директории [6-04/prometheus](https://github.com/netology-code/sdvps-homeworks/tree/main/6-04/prometheus)).
3. Обеспечьте внешний доступ к порту 9090 c докер-сервера.

### **Решение 3**

Листинг файла конфига:

version: '3'

services:

prometheus:

image: prom/prometheus:v2.47.2

container\_name: d.dunaev-netology-prometheus

command: --web.enable-lifecycle --config.file=/etc/prometheus/prometheus.yml

ports:

**-** 9090:9090

volumes:

**-** ./prometheus:/etc/prometheus

**-** prometheus-data:/prometheus

networks:

**-** d.dunaev-my-netology-hw

restart: always

volumes:

prometheus-data:

networks:

d.dunaev-my-netology-hw:

driver: bridge

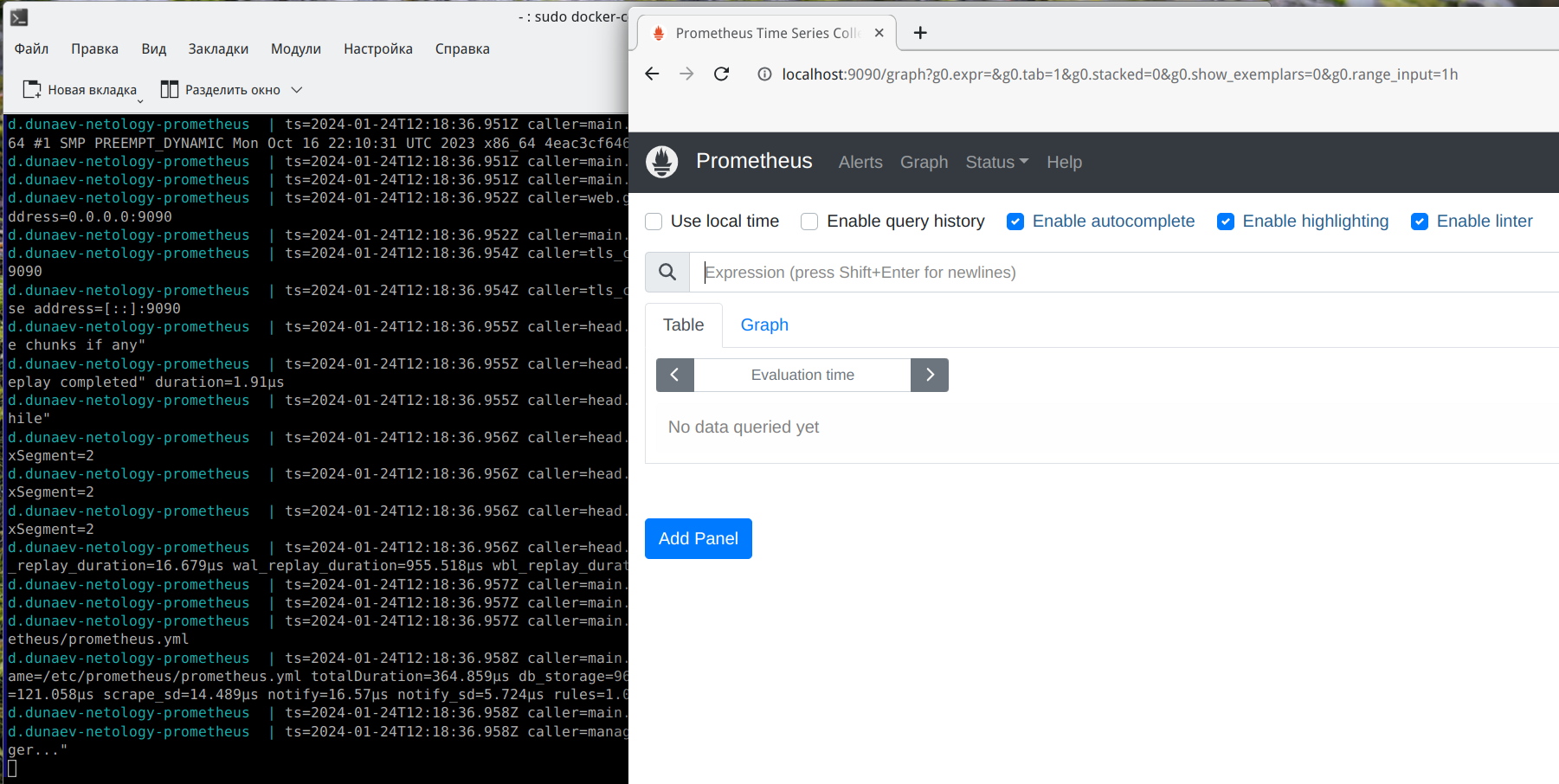
ipam:

config:

**-** subnet: 10.5.0.0/16

gateway: 10.5.0.1

Скриншот веб-интерфейса Prometheus:



### **Задание 4**

Выполните действия:

1. Создайте конфигурацию docker-compose для Pushgateway с именем контейнера <ваши фамилия и инициалы>-netology-pushgateway.
2. Обеспечьте внешний доступ к порту 9091 c докер-сервера.

### **Решение 4**

Листинг конфига:

pushgateway:

image: prom/pushgateway:v1.6.2

container\_name: d.dunaev-netology-pushgateway

ports:

**-** 9091:9091

networks:

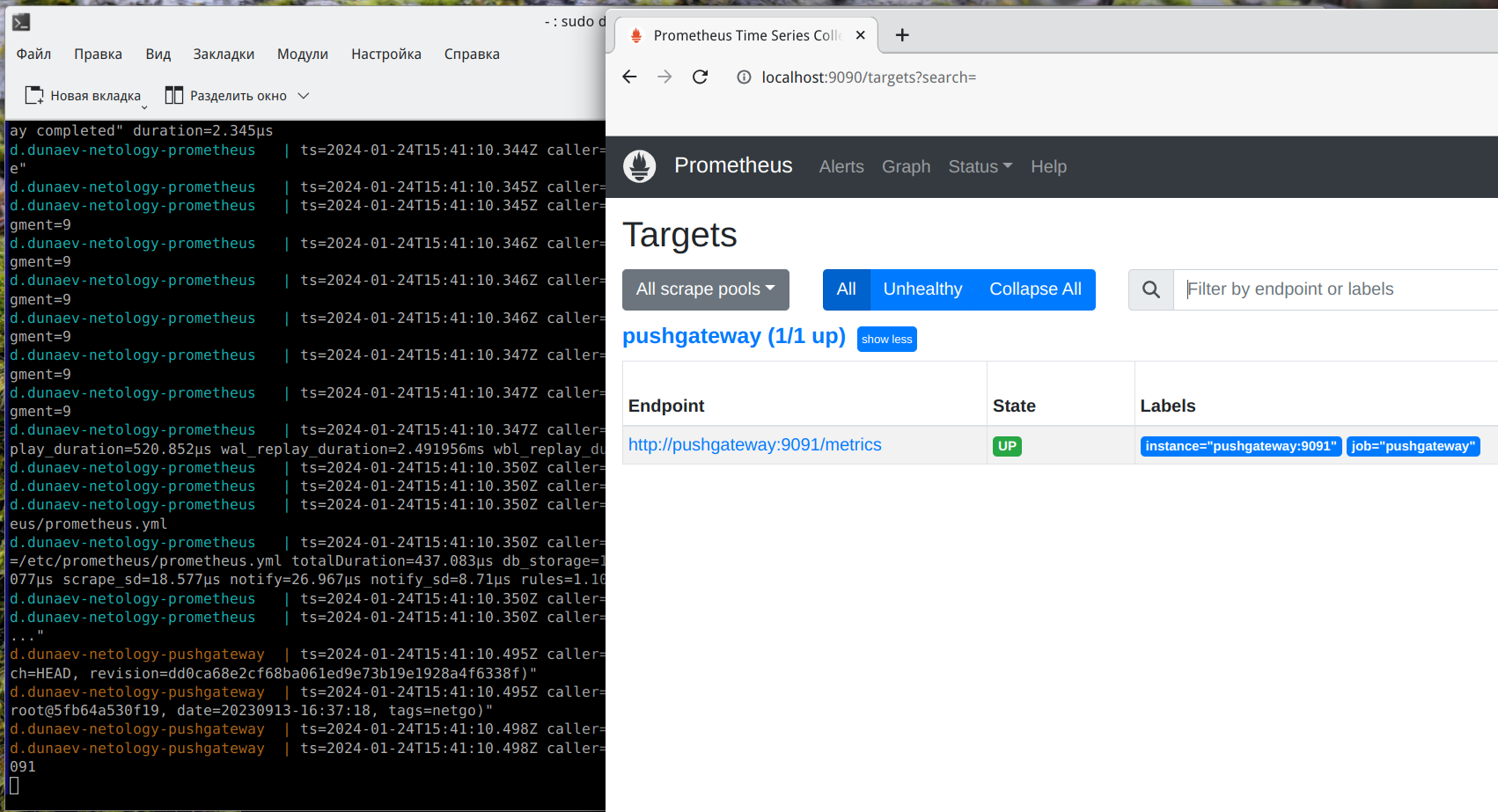
**-** d.dunaev-my-netology-hw

depends\_on:

**-** prometheus

restart: unless-stopped

Скриншот веб-интерфейса:



### **Задание 5**

Выполните действия:

1. Создайте конфигурацию docker-compose для Grafana с именем контейнера <ваши фамилия и инициалы>-netology-grafana.
2. Добавьте необходимые тома с данными и конфигурацией (конфигурация лежит в репозитории в директории [6-04/grafana](https://github.com/netology-code/sdvps-homeworks/tree/main/6-04/grafana)).
3. Добавьте переменную окружения с путем до файла с кастомными настройками (должен быть в томе), в самом файле пропишите логин=<ваши фамилия и инициалы> пароль=netology.
4. Обеспечьте внешний доступ к порту 3000 c порта 80 докер-сервера.

### **Решение 5**

Листинг конфига:

grafana:

image: grafana/grafana

container\_name: d.dunaev-netology-grafana

environment:

GF\_PATHS\_CONFIG: /etc/grafana/custom.ini

ports:

- 80:3000

volumes:

- ./grafana:/etc/grafana

- grafana-data:/var/lib/grafana

networks:

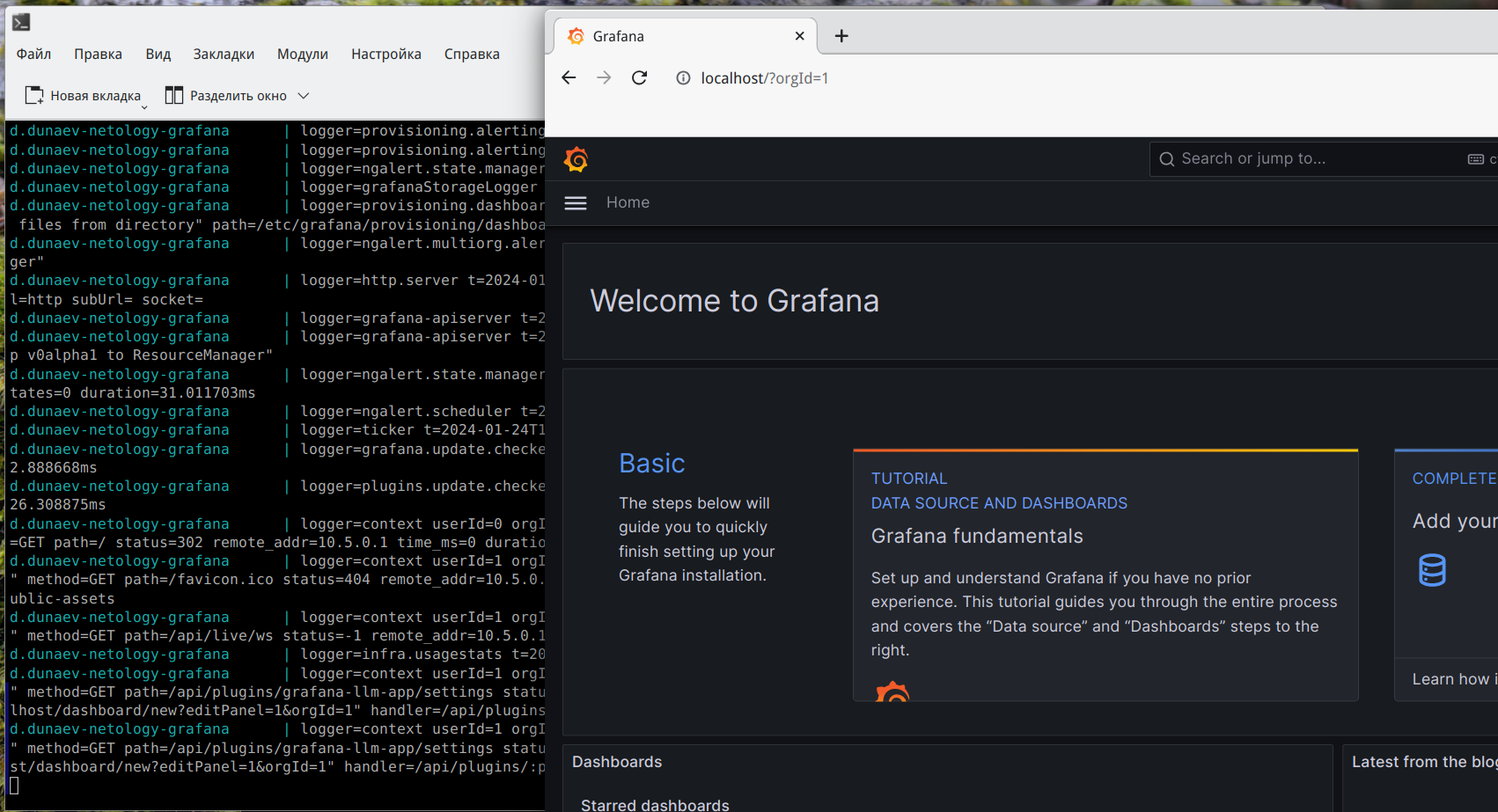
- d.dunaev-my-netology-hw

depends\_on:

- prometheus

restart: unless-stopped

Скриншот веб-интерфейса:



### **Задание 6**

Выполните действия.

1. Настройте поочередность запуска контейнеров.
2. Настройте режимы перезапуска для контейнеров.
3. Настройте использование контейнерами одной сети.
4. Запустите сценарий в detached режиме.

### **Решение 6**

Используются следующие настройки для сервисов:

services:

prometheus:

networks:

- d.dunaev-my-netology-hw

restart: always

pushgateway:

networks:

- d.dunaev-my-netology-hw

depends\_on:

- prometheus

restart: unless-stopped

grafana:

networks:

- d.dunaev-my-netology-hw

depends\_on:

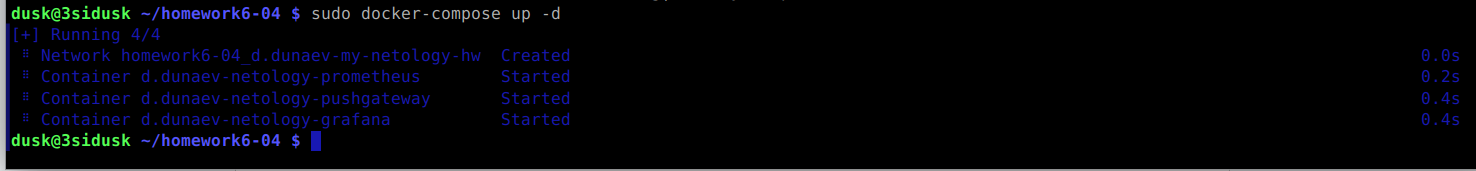
- prometheus

restart: unless-stopped

Команда для запуска в детач-режиме:

sudo docker-compose up -d

Скриншот:



### **Задание 7**

Выполните действия.

1. Выполните запрос в Pushgateway для помещения метрики <ваши фамилия и инициалы> со значением 5 в Prometheus: echo "<ваши фамилия и инициалы> 5" | curl --data-binary @- http://localhost:<внешний порт выбранный вами в пункте 2>/metrics/job/netology.
2. Залогиньтесь в Grafana с помощью логина и пароля из предыдущего задания.
3. Cоздайте Data Source Prometheus (Home -> Connections -> Data sources -> Add data source -> Prometheus -> указать "Prometheus server URL = http://<ваши фамилия и инициалы>-netology-prometheus:9090" -> Save & Test).
4. Создайте график на основе добавленной в пункте 5 метрики (Build a dashboard -> Add visualization -> Prometheus -> Select metric -> Metric explorer -> <ваши фамилия и инициалы -> Apply.

В качестве решения приложите:

* docker-compose.yml целиком;
* скриншот команды docker ps после запуске docker-compose.yml;
* скриншот графика, постоенного на основе вашей метрики.

### **Решение 7**

Листинг файла:

version: '3'

services:

prometheus:

image: prom/prometheus:v2.47.2

container\_name: d.dunaev-netology-prometheus

command: --web.enable-lifecycle --config.file=/etc/prometheus/prometheus.yml

ports:

- 9090:9090

volumes:

- ./prometheus:/etc/prometheus

- prometheus-data:/prometheus

networks:

- d.dunaev-my-netology-hw

restart: always

pushgateway:

image: prom/pushgateway:v1.6.2

container\_name: d.dunaev-netology-pushgateway

ports:

- 9091:9091

networks:

- d.dunaev-my-netology-hw

depends\_on:

- prometheus

restart: unless-stopped

grafana:

image: grafana/grafana

container\_name: d.dunaev-netology-grafana

environment:

GF\_PATHS\_CONFIG: /etc/grafana/custom.ini

ports:

- 80:3000

volumes:

- ./grafana:/etc/grafana

- grafana-data:/var/lib/grafana

networks:

- d.dunaev-my-netology-hw

depends\_on:

- prometheus

restart: unless-stopped

volumes:

prometheus-data:

grafana-data:

networks:

d.dunaev-my-netology-hw:

driver: bridge

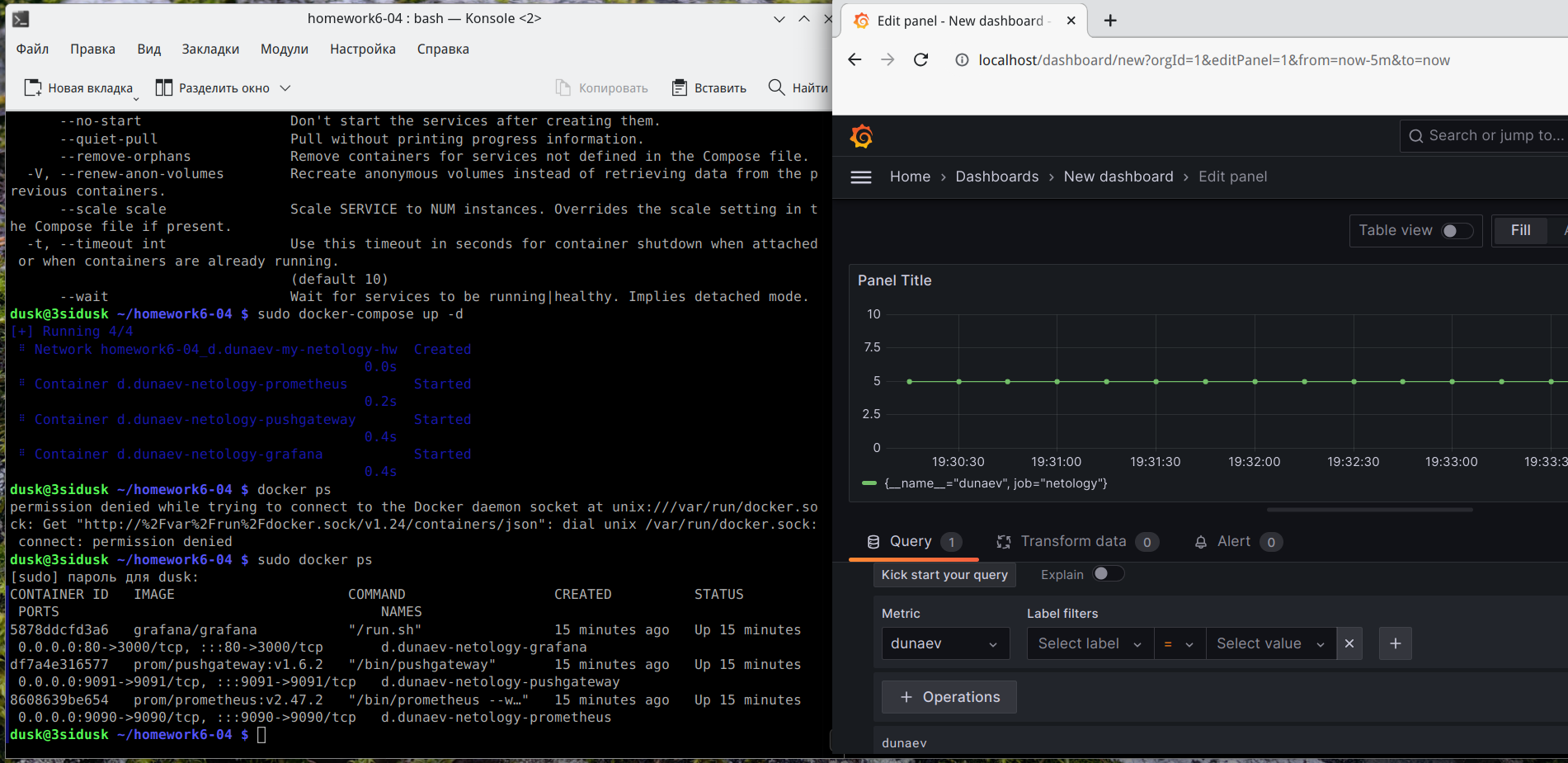
ipam:

config:

- subnet: 10.5.0.0/16

gateway: 10.5.0.1

Скриншот:



### **Задание 8**

Выполните действия:

1. Остановите и удалите все контейнеры одной командой.

В качестве решения приложите скриншот консоли с проделанными действиями.

### **Решение 8**

Скриншот:

