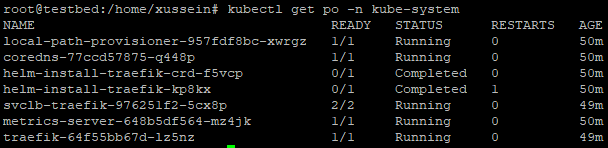
# **Задание 1**

Выполните действия:

1. Запустите Kubernetes локально, используя k3s или minikube на свой выбор.
2. Добейтесь стабильной работы всех системных контейнеров.
3. В качестве ответа пришлите скриншот результата выполнения команды kubectl get po -n kube-system.

# **Решение 1**



# **Задание 2**

Есть файл с деплоем:

---

apiVersion: apps/v1

kind: Deployment

metadata:

name: redis

spec:

selector:

matchLabels:

app: redis

replicas: 1

template:

metadata:

labels:

app: redis

spec:

containers:

- name: master

image: bitnami/redis

env:

- name: REDIS\_PASSWORD

value: password123

ports:

- containerPort: 6379

Выполните действия:

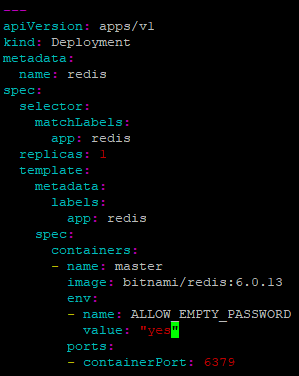
1. Измените файл с учётом условий:

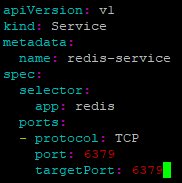
* redis должен запускаться без пароля;
* создайте Service, который будет направлять трафик на этот Deployment;
* версия образа redis должна быть зафиксирована на 6.0.13.

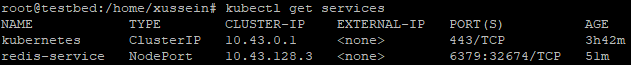
1. Запустите Deployment в своём кластере и добейтесь его стабильной работы.
2. В качестве решения пришлите получившийся файл.

# **Решение 2**









# **Задание 3**

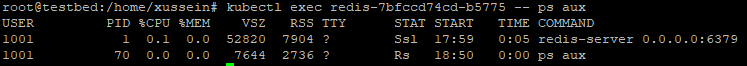
Выполните действия:

1. Напишите команды kubectl для контейнера из предыдущего задания:

* выполнения команды ps aux внутри контейнера;
* просмотра логов контейнера за последние 5 минут;
* удаления контейнера;
* проброса порта локальной машины в контейнер для отладки.

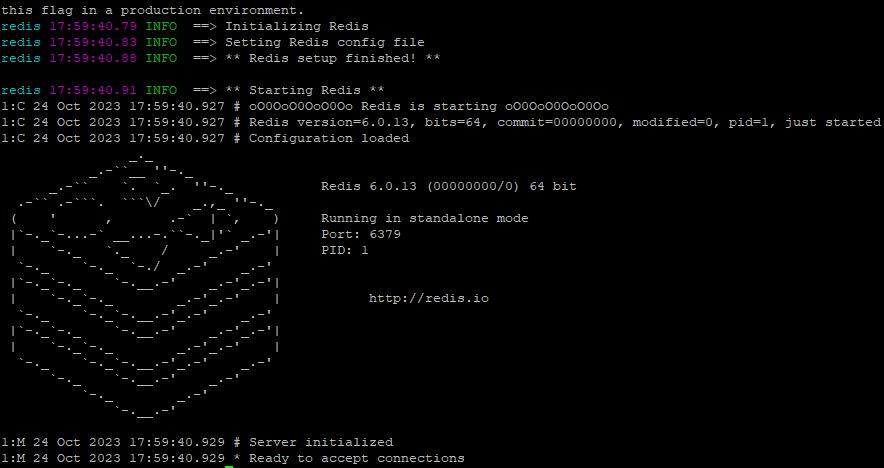
1. В качестве решения пришлите получившиеся команды.

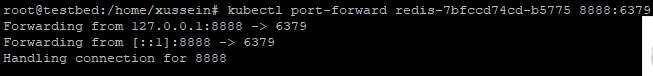
# **Решение 3**

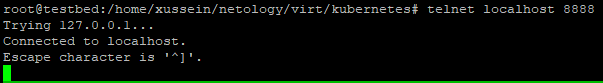


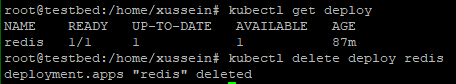


Поскольку с сервером с момента установки давно ничего не происходило то и лог за 5 минут пустой. Если глянуть целиком (конец лога):









# **Задание 4**

Есть конфигурация nginx:

location / {

add\_header Content-Type text/plain;

return 200 'Hello from k8s';

}

Выполните действия:

1. Напишите yaml-файлы для развёртки nginx, в которых будут присутствовать:

* ConfigMap с конфигом nginx;
* Deployment, который бы подключал этот configmap;
* Ingress, который будет направлять запросы по префиксу /test на наш сервис.

1. В качестве решения пришлите получившийся файл.

# **Решение 4**



Файл конфигурации:

---

apiVersion: v1

kind: ConfigMap

metadata:

name: nginx-config

data:

nginx.conf: |

server {

listen 80;

location / {

add\_header Content-Type text/plain;

return 200 'Hello from k8s';

}

}

---

apiVersion: apps/v1

kind: Deployment

metadata:

name: nginx-deployment

spec:

replicas: 1

selector:

matchLabels:

app: nginx

template:

metadata:

labels:

app: nginx

spec:

containers:

- name: nginx

image: nginx

ports:

- containerPort: 80

volumeMounts:

- name: nginx-config-volume

mountPath: /etc/nginx/conf.d

volumes:

- name: nginx-config-volume

configMap:

name: nginx-config

---

apiVersion: networking.k8s.io/v1

kind: Ingress

metadata:

name: nginx-ingress

spec:

rules:

- http:

paths:

- path: /test

pathType: Prefix

backend:

service:

name: nginx-service

port:

number: 80

---

apiVersion: v1

kind: Service

metadata:

name: nginx-service

spec:

selector:

app: nginx

ports:

- protocol: TCP

port: 80

targetPort: 80

Результат запроса curl:

