



Автономная некоммерческая организация профессионального образования
«Международная Академия Информационных Технологий «ИТ ХАБ»

Отчет по практическому заданию

Развертывание Minikube. Работа с манифестами.

Выполнил студент группы 4ИБ1:

Атаман. Н

(инициалы, фамилия)

«24» октября 2024 г.

Москва, 2024

1. Minikube использует Docker для управления контейнерами, поэтому Docker был установлен в первую очередь. Так же для корректной работы Docker был создан пользователь:

```
sudo usermod -aG docker $USER
```

2. Далее был установлен kubectl для управления Kubernetes-кластером

3. Установка и запуск самого Minikube, он был скачен с зеркала с использованием curl и далее установлен. При первом запуске возникла проблема, отсылающая нас к шагу №1:

```
nikita@debian:~$ sudo install minikube-linux-amd64 /usr/local/bin/minikube
nikita@debian:~$ minikube version
minikube version: v1.34.0
commit: 210b148df93a80eb872ecbeb7e35281b3c582c61
nikita@debian:~$ minikube start --driver=docker
🐳 minikube v1.34.0 on Debian 12.7
🌟 Using the docker driver based on user configuration

💡 Exiting due to PROVIDER_DOCKER_NEWGRP: "docker version --format <no value>-<no value>:<no value>" exit status 1: permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/version": dial unix /var/run/docker.sock: connect: permission denied
💡 Suggestion: Add your user to the 'docker' group: 'sudo usermod -aG docker $USER && newgrp docker'
📖 Documentation: https://docs.docker.com/engine/install/linux-postinstall/
```

После создание пользователя запуск прошел успешно:

```
nikita@debian: ~/Documents/DevOps
nikita@debian:~/Documents/DevOps$ newgrp docker
nikita@debian:~/Documents/DevOps$ docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS          NAMES
nikita@debian:~/Documents/DevOps$ minikube start
🐳 minikube v1.34.0 on Debian 12.7
🌟 Automatically selected the docker driver
🔧 Using Docker driver with root privileges
👉 Starting "minikube" primary control-plane node in "minikube" cluster
📦 Pulling base image v0.0.45 ...
📦 Downloading Kubernetes v1.31.0 preload ...
> preloaded-images-k8s-v18-v1...: 326.69 MiB / 326.69 MiB 100.00% 23.44 M
> gcr.io/k8s-minikube/kicbase...: 487.89 MiB / 487.90 MiB 100.00% 24.14 M
🔥 Creating docker container (CPUs=2, Memory=2200MB) ...
🔧 Preparing Kubernetes v1.31.0 on Docker 27.2.0 ...
  ▪ Generating certificates and keys ...
  ▪ Booting up control plane ...
  ▪ Configuring RBAC rules ...
🔧 Configuring bridge CNI (Container Networking Interface) ...
🔧 Verifying Kubernetes components...
  ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌞 Enabled addons: storage-provisioner, default-storageclass
🏁 Done! kubectl is now configured to use "minikube" cluster and "default" name space by default
nikita@debian:~/Documents/DevOps$ kubectl create -f pod.yaml
```

4. Создание pod.yaml

Был создан файл pod.yaml с уже исправленным кодом, в котором не хватало знака “-” и пробела после него в строке “name: nginx-container” для корректной работы. Текст файла стал выглядеть так:

```
apiVersion: v1
kind: Pod
metadata:
  name: my-nginx
spec:
  containers:
  - name: nginx-container
    image: nginx:1.10
```

Далее с помощью команды был создан pod:

```
kubectl create -f pod.yaml
```

4. Запуск Redis Pod:

```
kubectl run redis --image=redis:0.5 -n default
```

```
redis:0.5 redis:0.5nikita@debian:~/Documents/DevOps$ kubectl get pod redis -n default
redis:0.5 redis:0.5nikita@debian:~/Documents/DevOps$ kubectl edit pod redis -n default
```

Далее была изменена версия с помощью команды:

```
kubectl edit pod redis -n default
```

Текст файла стал выглядеть так:

```
spec:
  containers:
  - image: redis:0.6
    imagePullPolicy: IfNotPresent
    name: redis
    resources: {}
```

Проверка логов показала, что редактирование было выполнено правильно:

```
nikita@debian:~/Documents/DevOps$ kubectl logs redis -n default
1:C 25 Oct 2024 12:30:58.167 # oOoOoOoOoOoOo Redis is starting oOoOoOoOoOoOo
1:C 25 Oct 2024 12:30:58.168 # Redis version=6.0.20, bits=64, commit=00000000, modified=0, pid=1, just started
1:C 25 Oct 2024 12:30:58.168 # Warning: no config file specified, using the default config. In order to specify a config file use redis-server /path/to/redis.conf
1:M 25 Oct 2024 12:30:58.170 * Running mode=standalone, port=6379.
1:M 25 Oct 2024 12:30:58.171 # Server initialized
1:M 25 Oct 2024 12:30:58.171 # WARNING you have Transparent Huge Pages (THP) support enabled in your kernel. This will create latency and memory usage issues with Redis. To fix this issue run the command 'echo madvise > /sys/kernel/mm/transparent_hugepage/enabled' as root, and add it to your /etc/rc.local in order to retain the setting after a reboot. Redis must be restarted after THP is disabled (set to 'madvise' or 'never').
1:M 25 Oct 2024 12:30:58.173 * Ready to accept connections
```