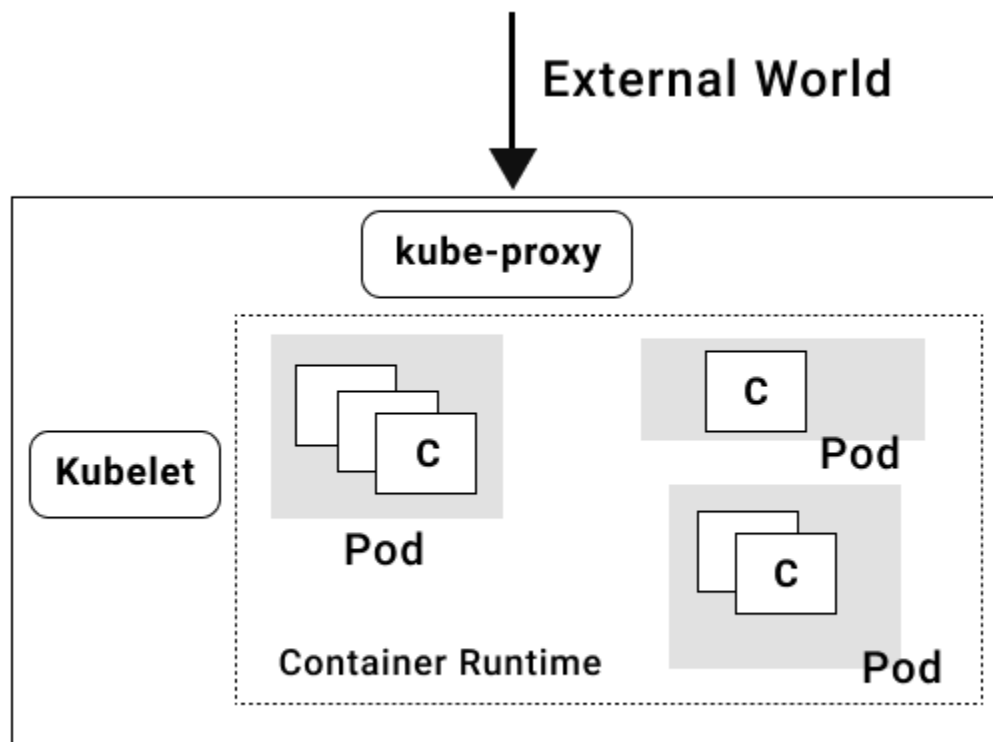


K8S 07: Kubernetes. Architecture.

Рабочие узлы

Описание:

На таких узлах запускается полезная нагрузка в виде приложений или задач.



Существуют три основных компонента:

- container runtime,
- kubelet,
- kube-proxy.

Container runtime

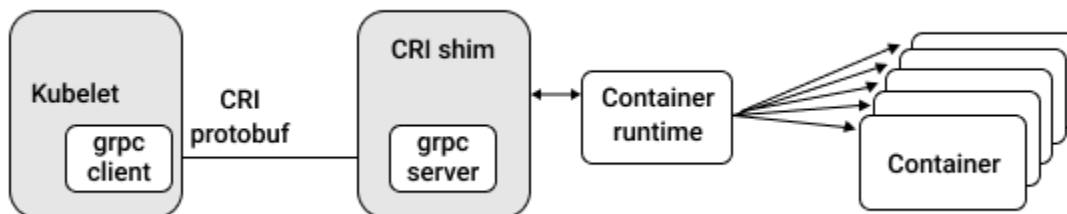
Container runtime — это совокупность инструментов для запуска контейнеров или среда исполнения контейнеров. В качестве таких инструментов может быть использован docker-ce, cri-o, containerd и т.д.

[Various container runtimes supported by Kubernetes](#)

Kubelet

Агент кластера запускается на каждом рабочем узле, соединяется с мастером и работает с ним по API. Агент кластера получает задачи на выполнение от мастера. Кроме того, он

сообщает мастеру статус узла и контейнеров. Помимо мастера агент кластера работает с container runtime для запуска контейнеров, используя CRI (container runtime interface).



Через специализированные плагины могут быть использованы различные реализации container runtime для запуска полезной нагрузки.

kube-проxy

Это сетевой прокси, запускаемый на каждом рабочем узле. Связывается с API-сервером для получения событий на создание/удаление Service, добавляет маршруты в таблицу маршрутизации и делает многое другое, позволяя контейнерам на разных узлах общаться друг с другом. Также этот элемент организывает доступ к приложениям со стороны внешних пользователей.

Полезные ссылки:

- [Nodes \(official docs\)](#)
- [Introduction to Kubernetes Architecture](#)
- [Kubernetes 101: Nodes](#)

Задание:

1. С помощью приведенной команды `kubectl run` создайте pod с nginx с такими параметрами:
 - имя — `nginx-demo`,
 - `image: nginx:latest`,
 - pod должен принимать запросы на 80 порту,
 - добавить `labels name=nginx-demo`.
2. Сохраните приведенный ниже конфиг в файл `nginx-demo-nodeport-svc.yml` и примените командой `kubectl apply -f nginx-demo-nodeport-svc.yml`. Мы разберем текущую конфигурацию позже.

```
apiVersion: v1
kind: Service
metadata:
  name: nginx-demo
labels:
  name: nginx-demo-nodeport-svc
```

```
spec:
  type: NodePort
  ports:
    - port: 80
      nodePort: 30180
      name: http
    - port: 443
      nodePort: 31443
      name: https
  selector:
    name: nginx-demo
```

3. Убедитесь, что сервис корректно создавался командой `kubectl get svc`. Вывод должен выглядеть примерно так:

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
nginx-demo	NodePort	10.106.74.164	<none>	80:30180/TCP,443:31443/TCP	46s

4. Найдите и объясните правила сетевой политики на VM с `minikube` на `kube-proxy`. Приведите в ответе найденные правила вместе с их описанием.
5. Выведите последние 20 сообщений из лога `kubelet` с помощью `journalctl`. Приведите команду и вывод в ответе.