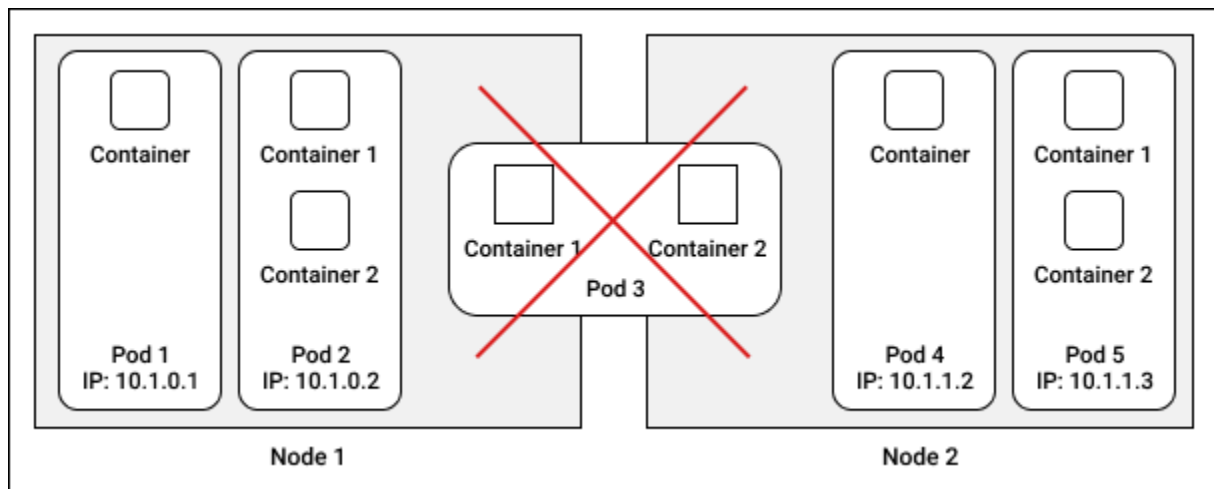


K8S 09: Kubernetes. Resources. Pod

Описание:

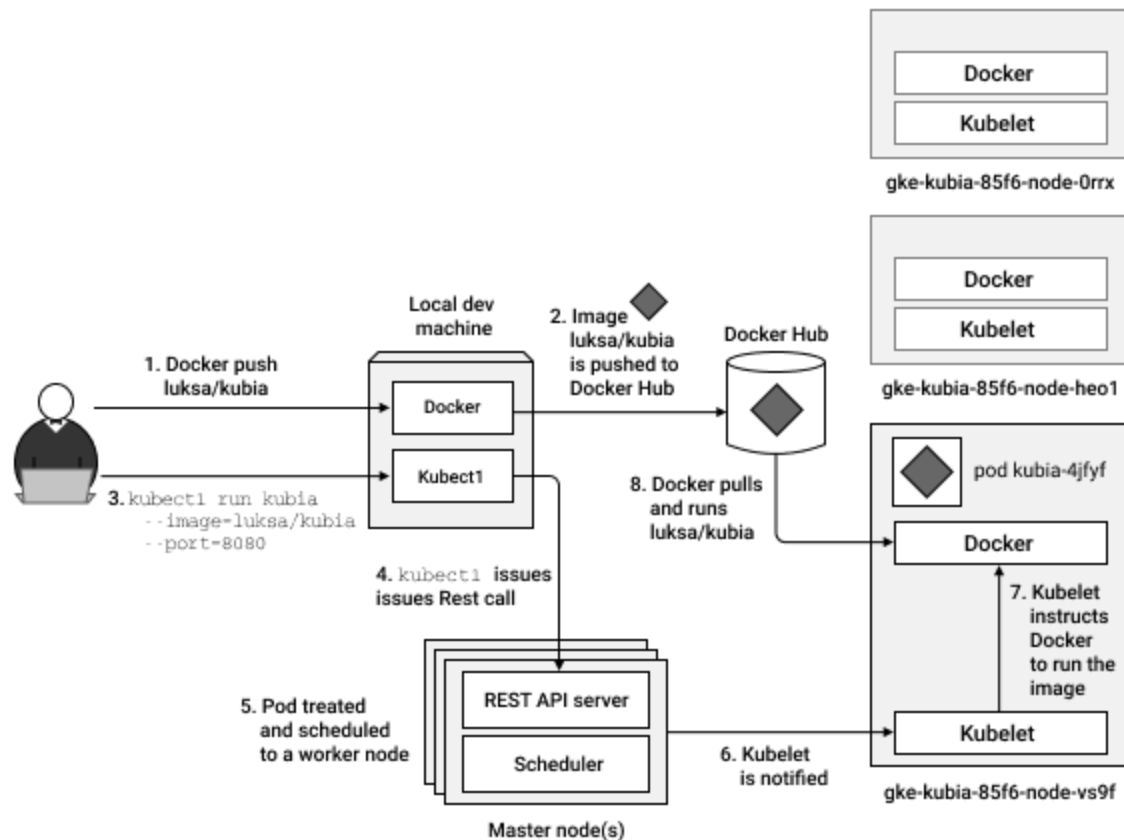
Pod представляет собой группу из одного или нескольких контейнеров. Это неделимая единица, с точки зрения Kubernetes. Они всегда выполняются на одном рабочем узле, в одном пространстве имени Linux. Каждый Pod подобен машине с IP-адресом, сетью, процессами, дисками. Приложение может состоять из нескольких процессов, например, front + back. Для корректной работы приложения вам, скорее всего, нужно будет запускать эти процессы в единой экосистеме. Каждый процесс будет запущен в отдельном контейнере. Набор контейнеров внутри Pod неделим. Если в составе pod находятся 4 контейнера, нельзя запустить два на одной физической машине и два на другой.



Pods — это временные объекты. Они могут запускаться и исчезать сотнями. Для обеспечения отказоустойчивости используются специальные контроллеры. Примерами таких контроллеров являются Deployments, ReplicaSets, ReplicationController. В прошлом задании мы уже запустили Pod с nginx, давайте проверим, что все работает.

```
kubectl get pods
```

Kubernetes не работает с одиночными контейнерами, в выводе будут только Pods. Больше информации о pod можно получить через `kubectl describe pod $podname`. При создании пода из предыдущего задания мы применили его описание из yaml-файла, которое было конвертировано в json и отправлено к API сервера. Давайте разберемся, что произошло в этот момент:



После отправки REST-запроса controller-manager определил различия между желаемым и фактическим состоянием кластера, передав на исполнение Kubernetes через API информацию о создании требуемого ресурса типа Pod. Далее scheduler назначил положение пода между узлами кластера, и информация об этом была передана kubelet на целевой ноде. В случае с minikube выбора, в общем-то, нет, но это упрощенная схема. Kubelet с помощью CRI передал информацию о необходимости создания контейнера в container runtime, реализованный с помощью runc и containerd (в нашем случае), после чего был создан требуемый ресурс. Именно это имеется в виду, когда произносится упрощенная фраза «Kubernetes создал под».

Полезные ссылки:

- [Pods in Kubernetes](#)
- [Pods \(official docs\)](#)
- [Overview of Pods](#)

Задание:

1. Используя представленный ниже шаблон, создать yaml-описание пода на базе cowsay docker image godlovedc/lolcow:latest. Также необходимо корректно установить .metadata.labels.app в соответствии с вашим email (@ заменить на _at_).

first-pod.json:

```
{
  "apiVersion": "v1",
  "kind": "Pod",
  "metadata": {
    "name": "first-pod",
    "labels": {
      "app": "label-it"
    }
  },
  "spec": {
    "containers": [
      {
        "name": "frontend",
        "image": "image-name"
      }
    ]
  }
}
```

2. Примените изменения из созданного файла `first-pod.yaml`. Команду и вывод сохранить.
3. Проверьте, что Pod успешно запущен. Команду и вывод сохранить.
4. Выведите подробности по запущенному поду. Команду и вывод сохранить.
5. Вывести логи пода. Команду и вывод сохранить.
6. Отправьте в ответ созданный `first-pod.yaml`, все сохраненные команды и вышеуказанные выводы.