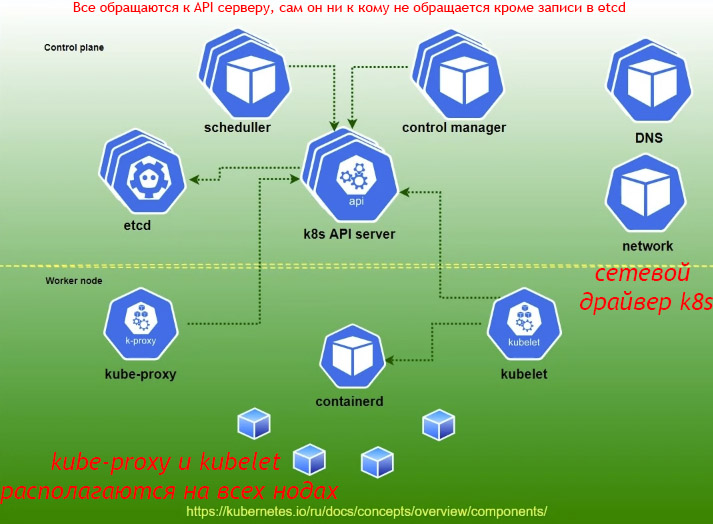
****

Всё, кроме kubelet и containerd это докер образы.

Kubelet, containerd - это приложение, работает как обычный сервис

**API server**

Общение с кластером kubernetes происходит через API server (интерфейс взаимодействия между программами)

Например манифест файл (создание подов/деплоймента и тд) передаем в кластер через API сервер. Kubectl через rest api общается с кластером. Он же ограничивает доступ к rest api, таким образом ограничиваем доступ к каким-то компонентам k8s

**Etcd**

Распределённое и высоконадёжное хранилище данных в формате "ключ-значение", которое используется как основное хранилище всех данных кластера в Kubernetes

**Controller-manager**

Отвечает за то, что вы описали в etcd появилось в кластере

Компонент control plane запускает процессы *контроллера* (управляющий цикл который отслеживает общее состяние кластера через API-сервер и вносит изменения пытаясь привести текущее состояние к желаемому состоянию)

Эти контроллеры включают:

- Контроллер узла (Node Controller): уведомляет и реагирует на сбои узла.

- Контроллер репликации (Replication Controller): поддерживает правильное количество подов для каждого объекта контроллера репликации в системе.

- Контроллер конечных точек (Endpoints Controller): заполняет объект конечных точек (Endpoints), то есть связывает сервисы (Services) и поды (Pods).

- Контроллеры учетных записей и токенов (Account & Token Controllers): создают стандартные учетные записи и токены доступа API для новых пространств имен

**Scheduller**

Компонент, который отвечает за размещение подов на конкретных нодах кластера.

Он не сам размещает под на ноде а делает эту информацию доступной для **kubelet** - а он уже взаимодействует с системой контейнеризации (**docker/containerD/crio**) и запускает имедж в контейнере.

Делает liveness rediness startup пробы, стоит на всех нодах

**Kube-proxy**

Организация работы сервисов (iptables) стоит на всех нодах

**Последовательность запросов**

На примере replica set, манифест который описывает replica set который нужно получить в системе

все компоненты с определенной периодичностью запрашивают данные в REST API

самые загруженные оказываются API и БД etcd

