Kubernetes 101:
ОСНОВЫ

Андрей Ветров

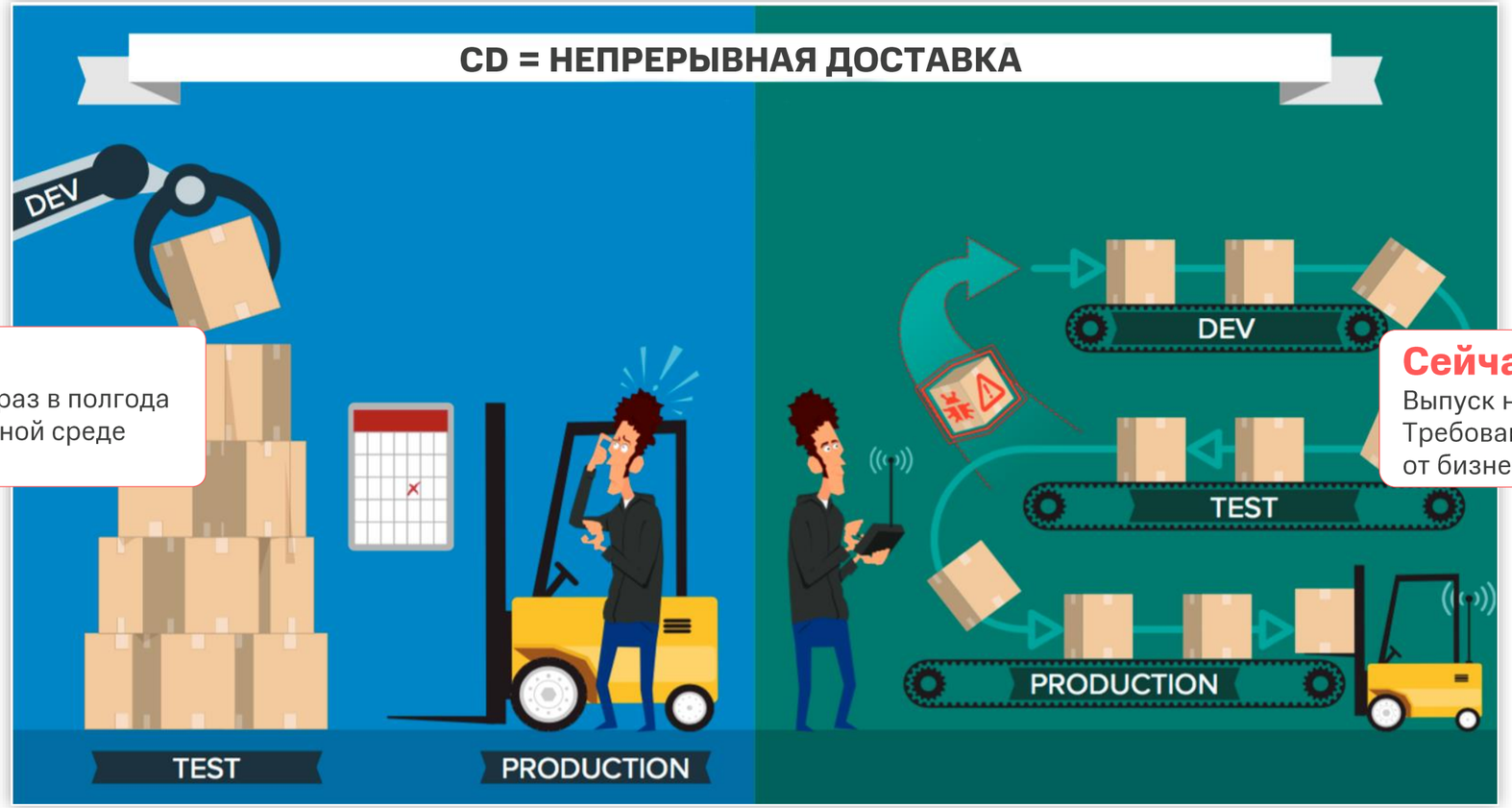
Senior Systems Engineer,
VMware

Cloud Thinking

Непрерывная доставка

Source: Pivotal – The Journey to Cloud Native

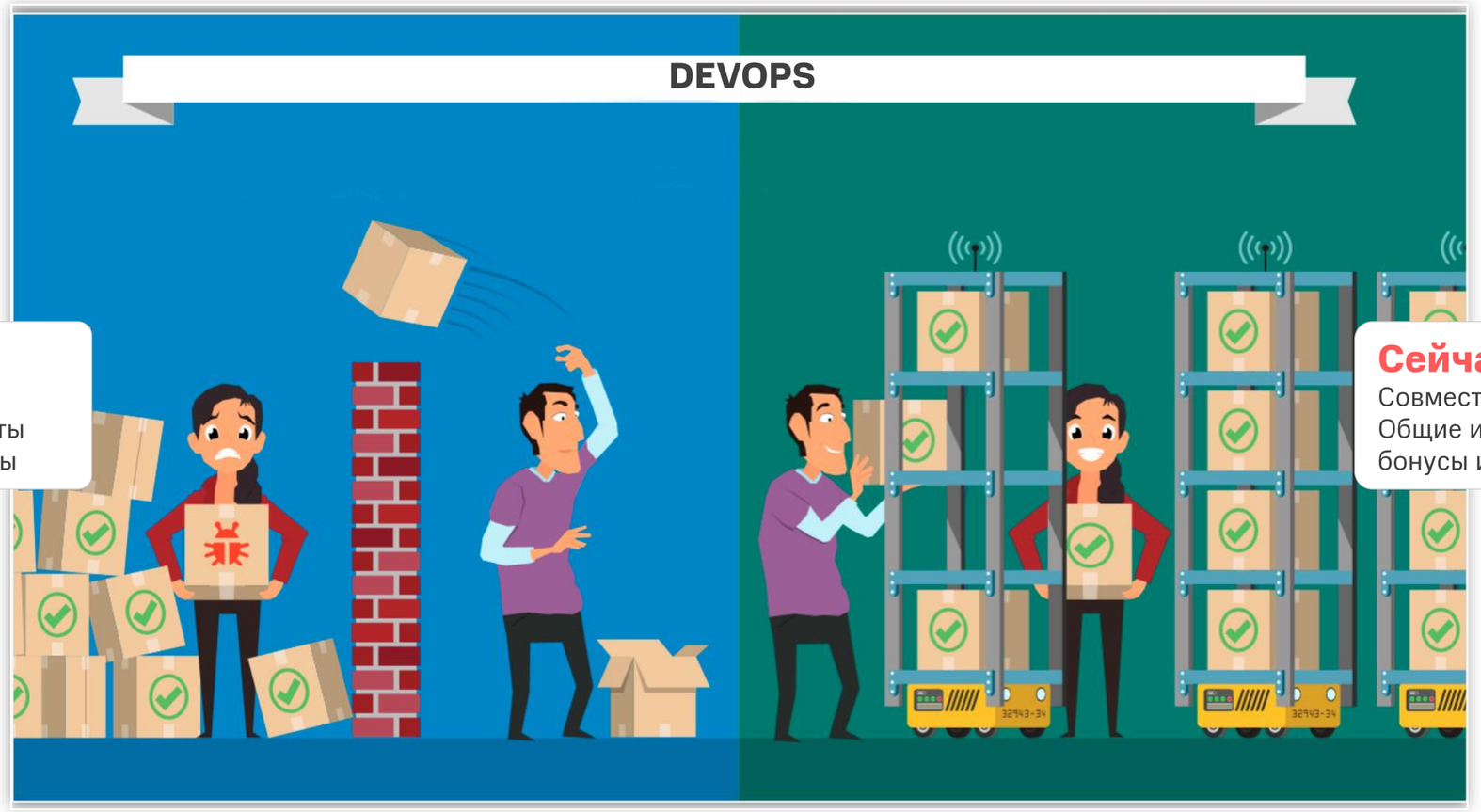
Раньше:
Выпуск новой версии 1 раз в полгода
Много проблем в реальной среде



Сейчас:
Выпуск новой версии 1 раз в неделю
Требования высоких стандартов кода от бизнеса

DevOps

Source: Pivotal – The Journey to Cloud Native



Раньше:
Не моя проблема
Раздельные инструменты
Непрозрачные процессы

Сейчас:
Совместная ответственность
Общие инструменты, процессы, бонусы и культура

Микросервисы

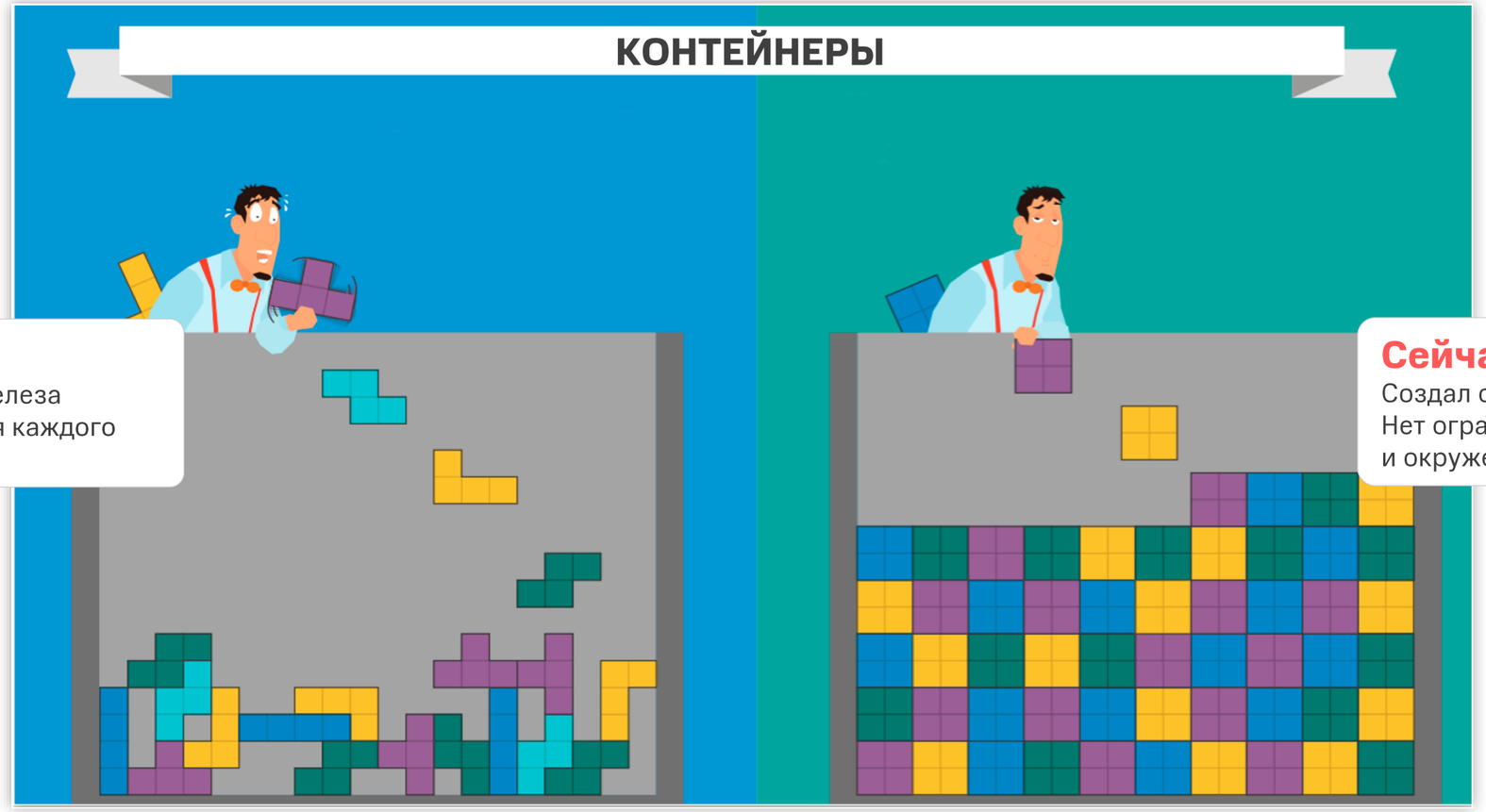
Source: Pivotal – The Journey to Cloud Native

Раньше:
Монолитные приложения
Долгий цикл разработки: ожидание тестов и команды

Сейчас:
Отдельные небольшие сервисы
Независимое развертывание

Контейнеры

Source: Pivotal – The Journey to Cloud Native

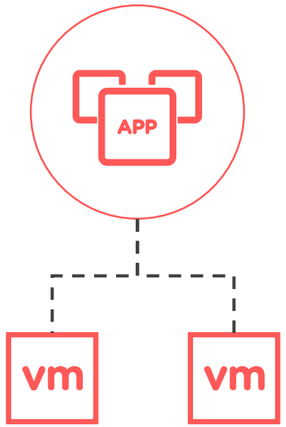


Раньше:
Совместимость ПО и железа
Сборка кода заново для каждого окружения

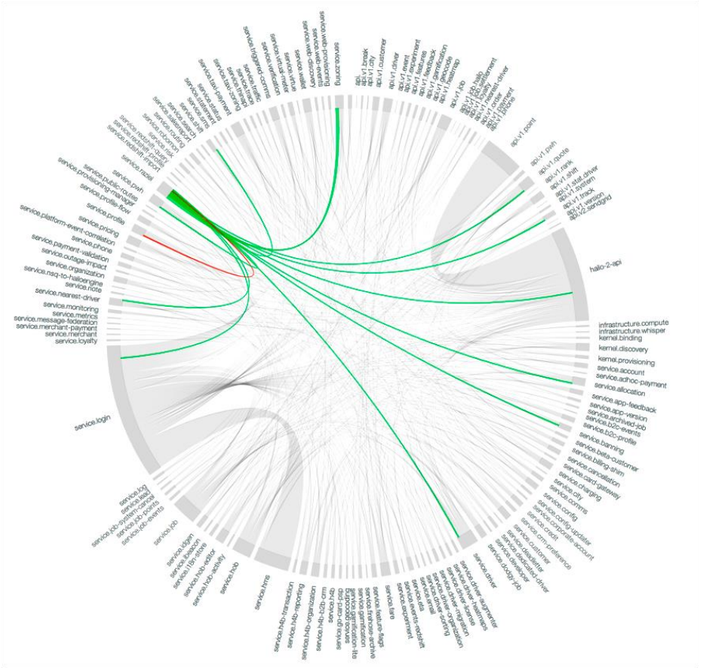
Сейчас:
Создал один раз, запустить
Нет ограничений к оборудованию и окружению

Приложения тогда и сейчас

Прошедшее настоящее

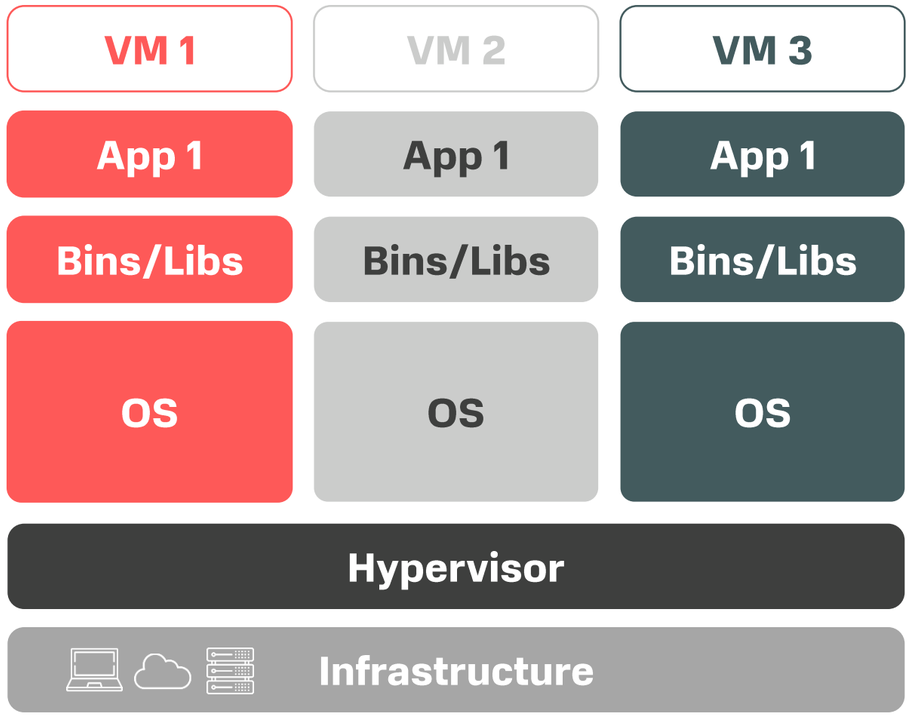


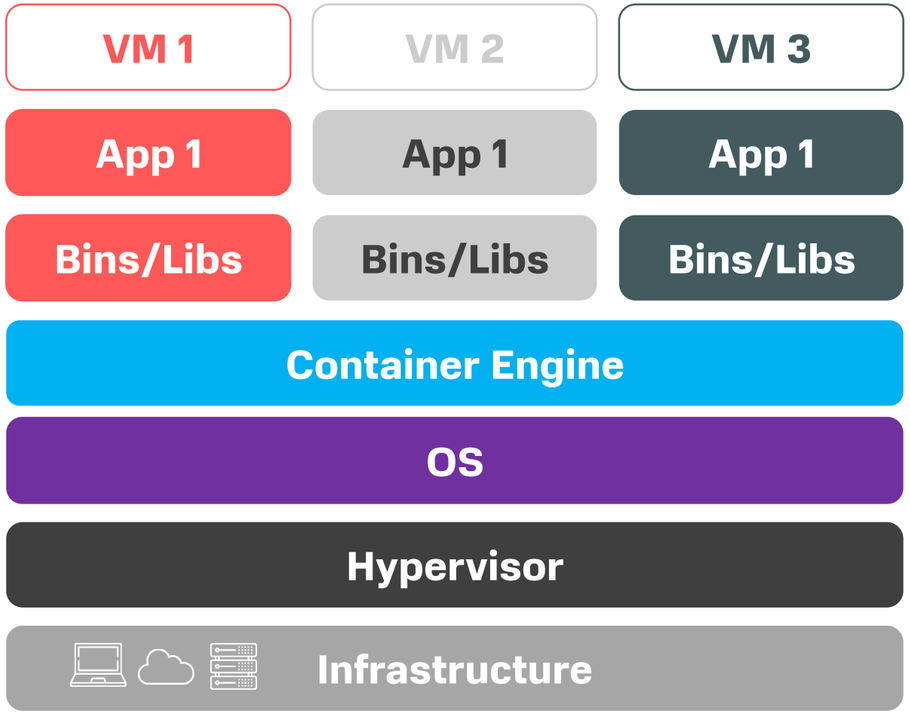
Настоящее будущее



VM vs Containers

Вспомним про Docker

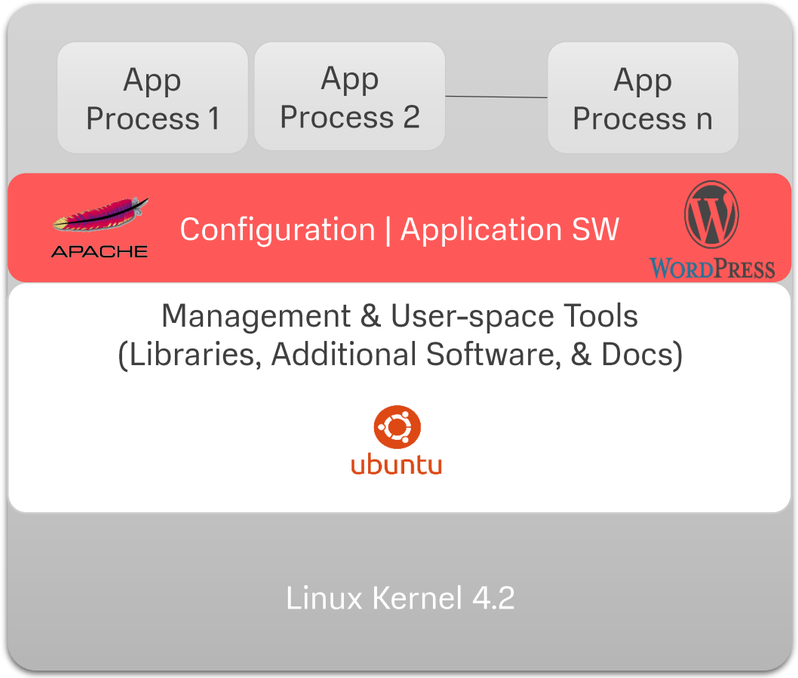




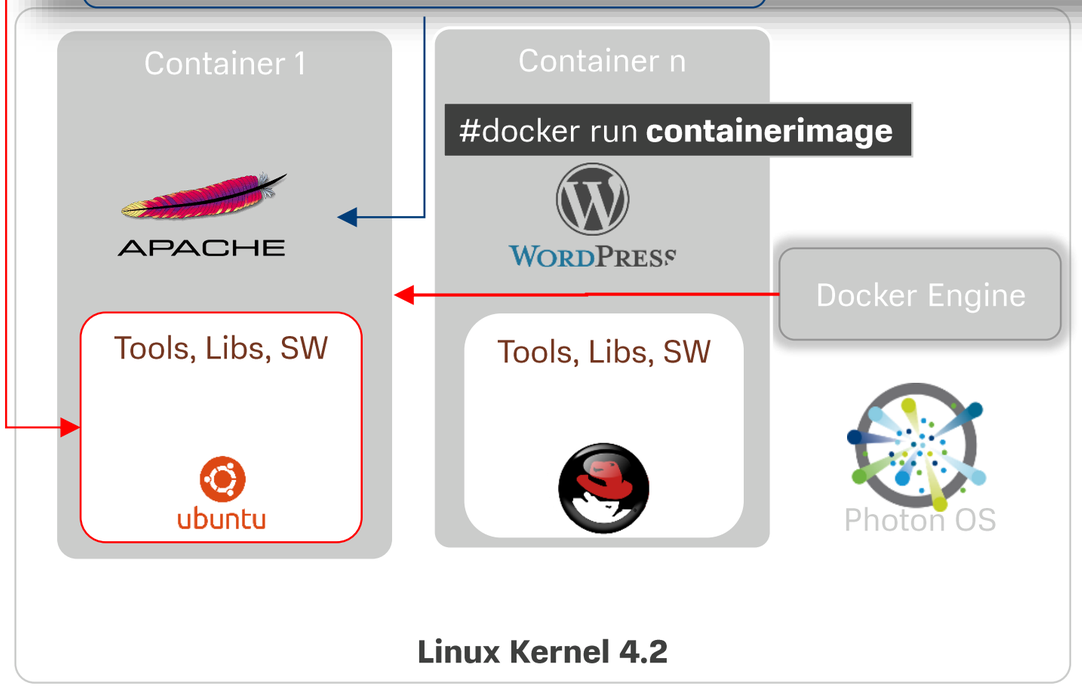
Зачем нужны контейнеры?

Container image создается через Dockerfile

```
15 lines (9 sloc)  389 Bytes
1  # A basic apache server. To use either add or bind mount content under /var/www
2  FROM ubuntu:12.04
3
4  MAINTAINER Kimbro Staken version: 0.1
5
6  RUN apt-get update && apt-get install -y apache2 && apt-get clean && rm -rf /var/lib/apt/l
7
8  ENV APACHE_RUN_USER www-data
9  ENV APACHE_RUN_GROUP www-data
10 ENV APACHE_LOG_DIR /var/log/apache2
11
12 EXPOSE 80
13
14 CMD ["/usr/sbin/apache2", "-D", "FOREGROUND"]
```



Standard Linux Host

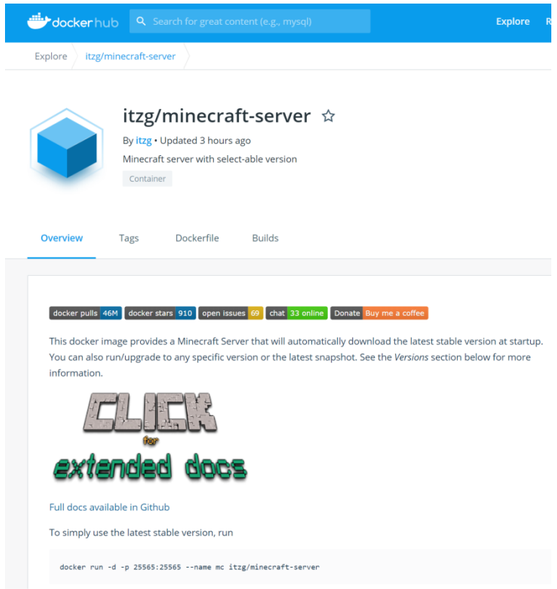


Linux Container Host

Вспомним про Docker

docker run -d -p 25565:25565 --name mc itzg/minecraft-server

docker run -d -e VERSION=1.7.9 ...



Dockerfile – это сценарий, который состоит из последовательности команд и аргументов, необходимых для создания образа.

minecraft.dockerfile

```
FROM adoptopenjdk/openjdk13:alpi

LABEL maintainer "itzg"

RUN apk add --no-cache -U \
  openssl \
  imagemagick \
  lsof \
  su-exec \
  shadow \
  bash \
  curl iputils wget \
  git \
  jq \
```

Docker-compose.yml – это описание мультиконтейнерного docker-приложения.

docker-compose.yml

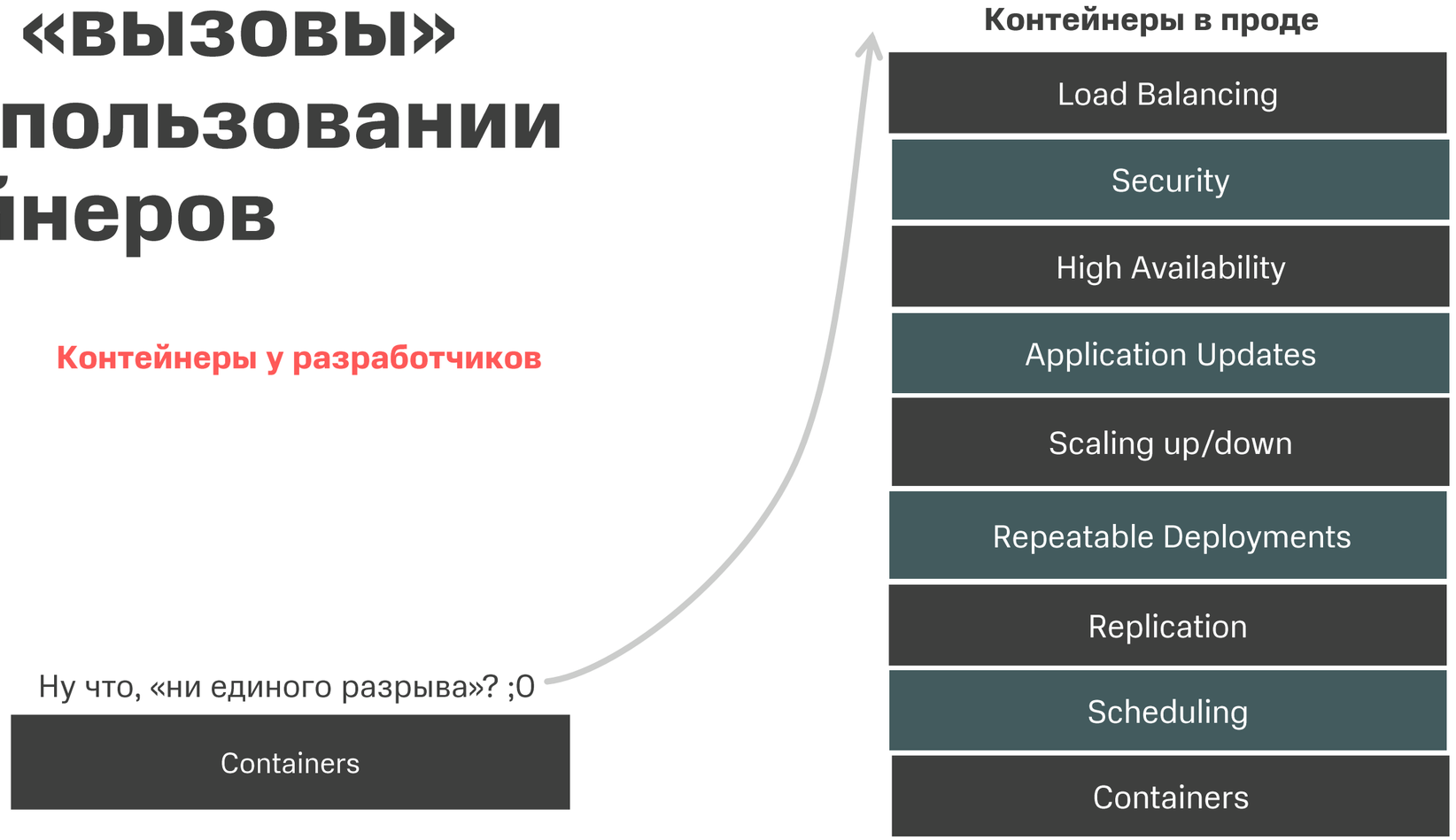
```
version: '3'
# Other docker-compose examples in /examples

services:
minecraft:
  image: itzg/minecraft-server
  ports:
    - "25565:25565"
  volumes:
    - "mc:/data"
  environment:
    EULA: "TRUE"
    CONSOLE: "false"
    ENABLE_RCON: "true"
    RCON_PASSWORD: "testing"
    RCON_PORT: 28016
  restart: always
rcon:
  image: itzg/rcon
  ports:
    - "4326:4326"
    - "4327:4327"
  volumes:
    - "rcon:/opt/rcon-web-admin/db"

volumes:
```

Новые «вызовы» при использовании контейнеров

Контейнеры у разработчиков

Ну что, «ни единого разрыва»? ;0

Containers



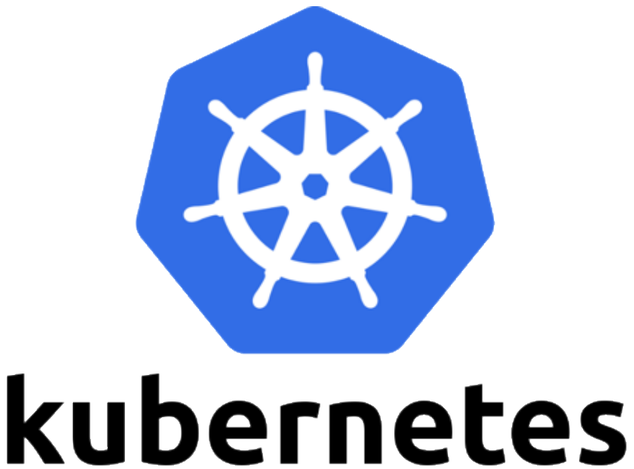

RENA
MONROVIA

Что такое Kubernetes (K8s)

Kubernetes – это open-source-платформа для управления и автоматизации развертывания, масштабирования и управления контейнеризированными приложениями кластера рабочих узлов (worker nodes).

Ключевые возможности:

- Скорость и повторяемость развертывания
- Масштабирование «на лету»
- Простой выкат новых релизов
- Оптимизация затрат ресурсов



Сущности K8s

Namespace

Pod

ReplicaSet

Deployment

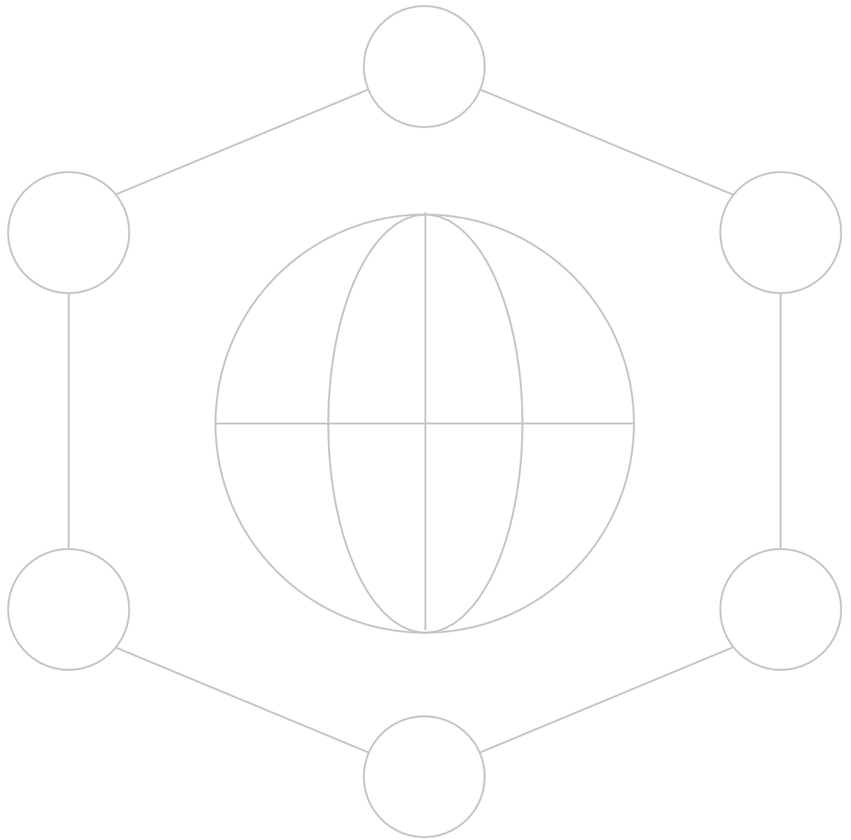
DeamonSet

Job / CronJob

ConfigMap

Secret

- RBAC
- ServiceAccount
- User
- Group
- Role
- RoleBinding
- ClusterRole
- ClusterRoleBinding

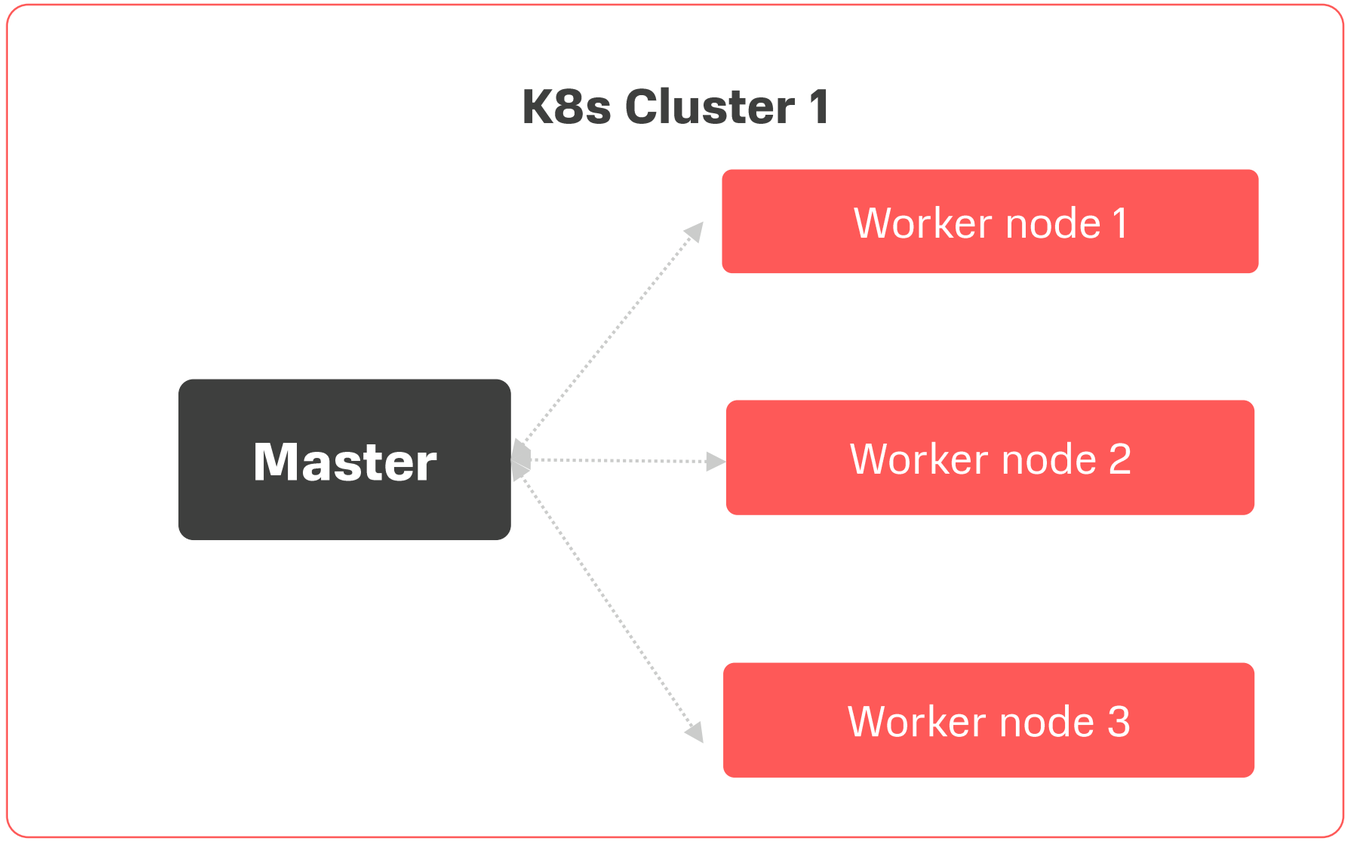


K8s Cluster

Master + worker nodes

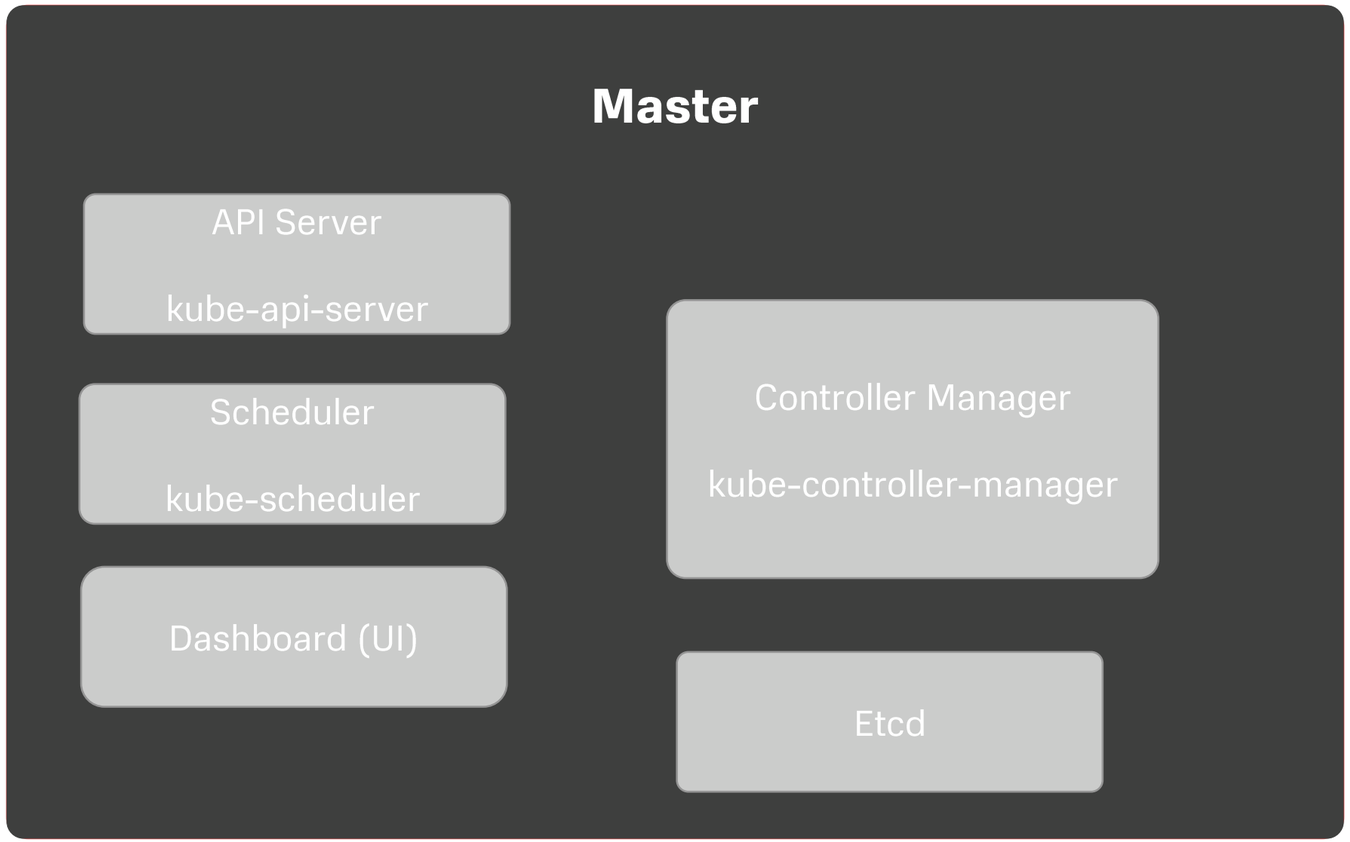
Namespaces

Механизм разделения ресурсов в логические группы нагрузок



K8s Master

API Server
Scheduler
Dashboard (UI)
Controller Manager
Replication Controller



K8s etcd

- Используется как распределенное хранение ключей/секретов в Kubernetes
- Хранит конфигурационные данные для каждого узла кластера
- Может быть распределенным на несколько узлов
- Используется для service discovery
- Представляет состояние кластера

K8s Worker Node

Kubelet

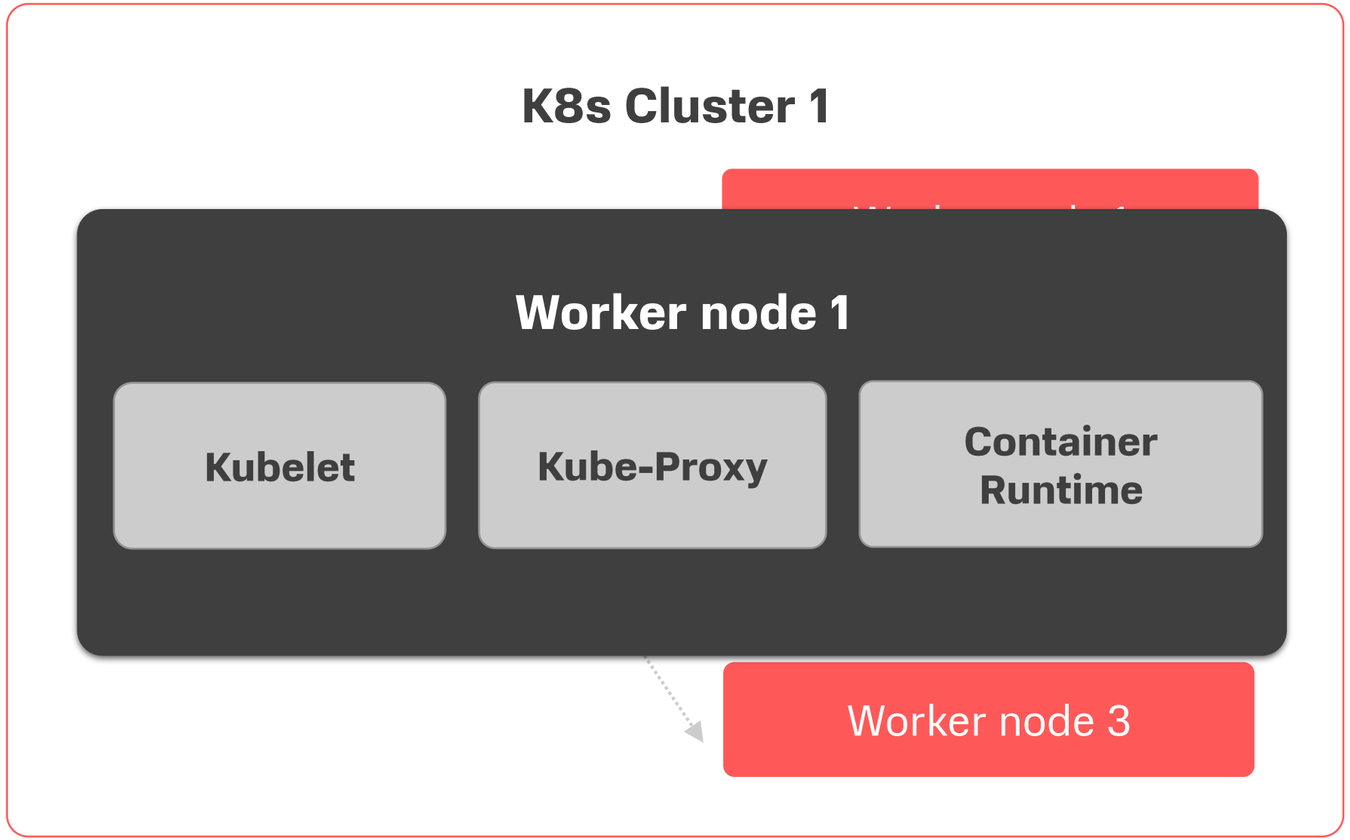
Агент на узлах, наблюдающий за PodSpecs, для определения, что требуется запустить

Kube-Proxy

Сервис, наблюдающий за конфигурацией services на API Server и применяющий east/west балансировку нагрузки на узлах, используя NAT в IPTables

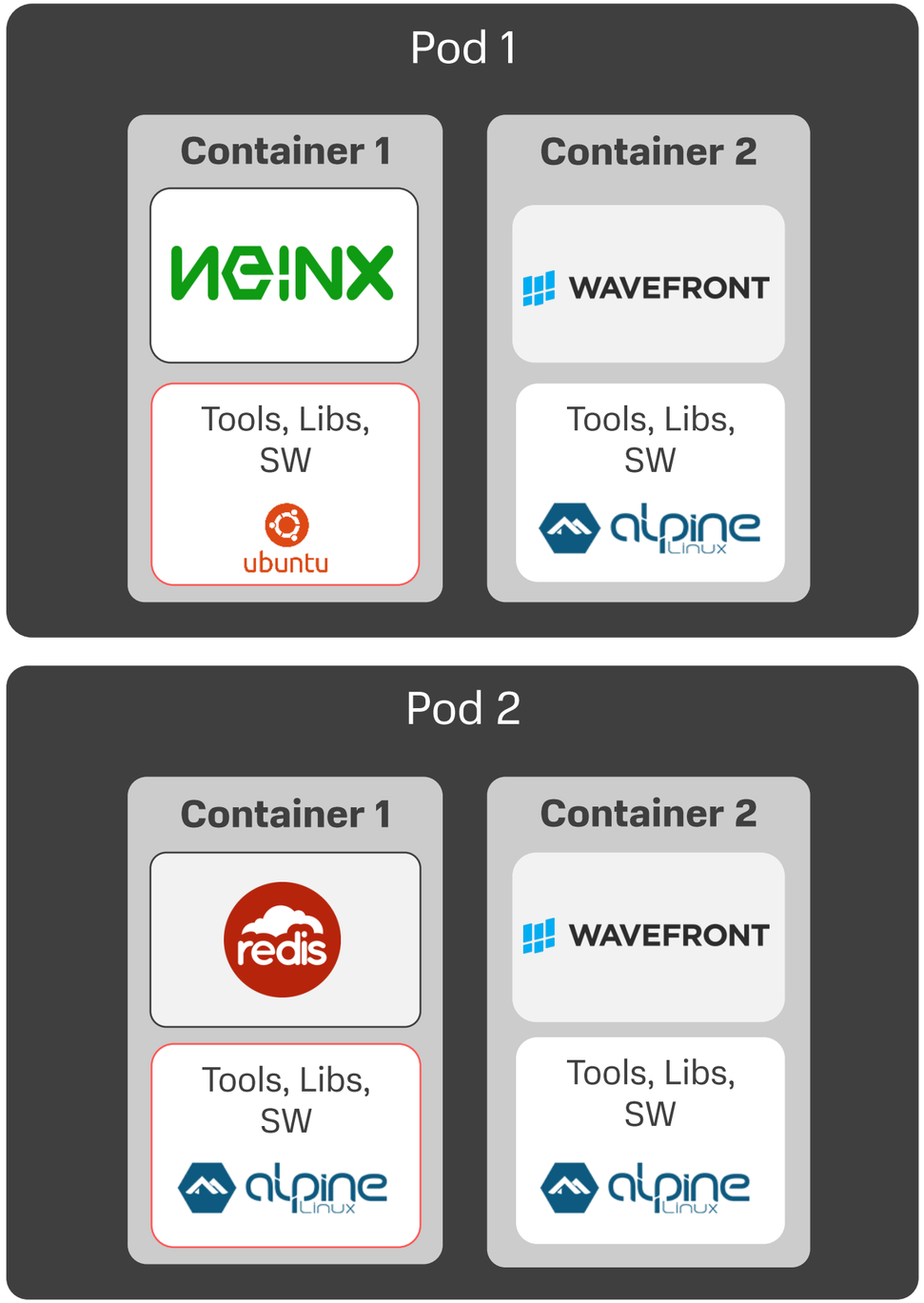
Container Runtime

Docker/Rkt/...



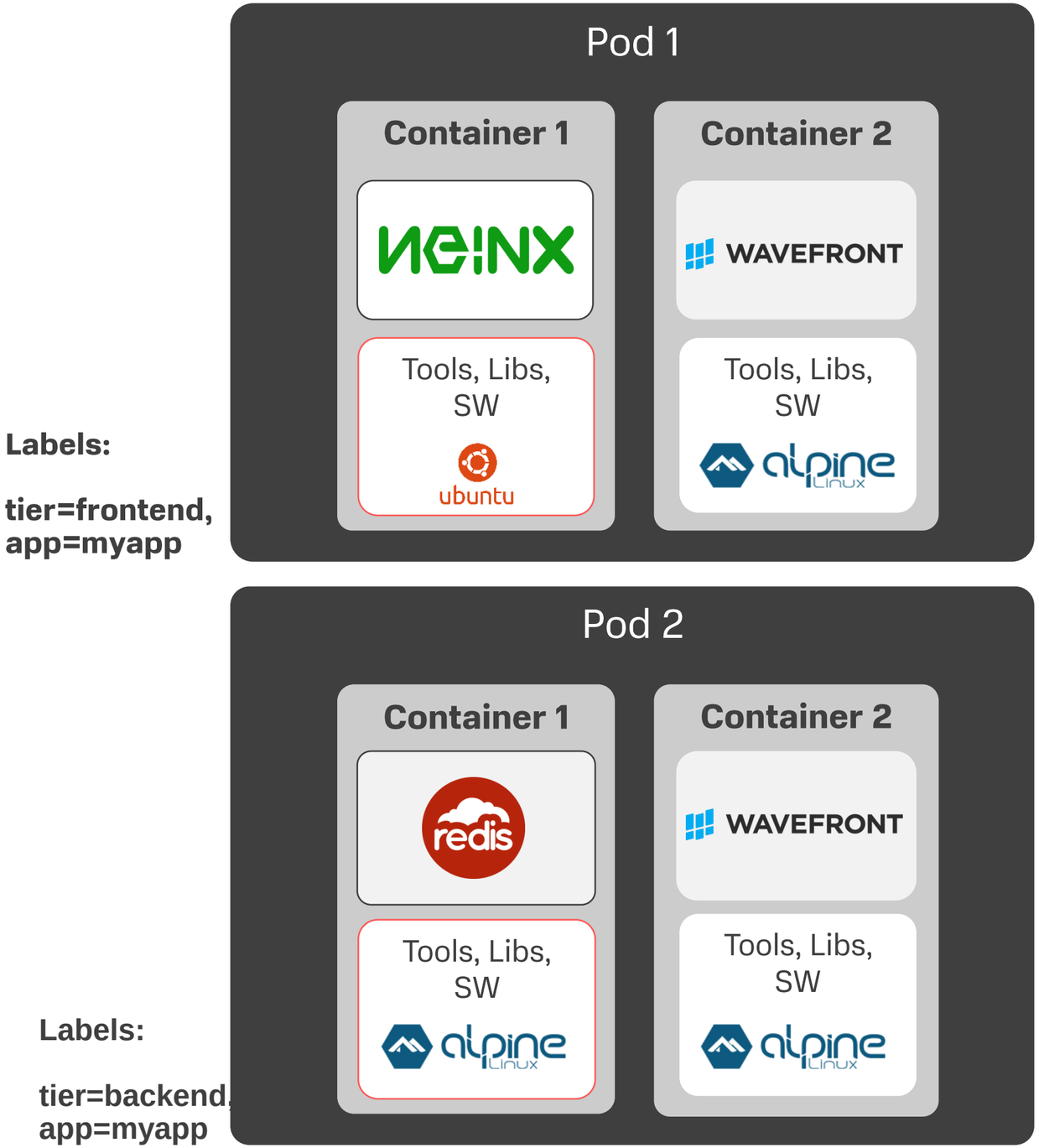
Pods

- Узел (под) – это группа из одного или более контейнеров
- Контейнеры в поде разделяют один IP address и список портов и могут взаимодействовать друг с другом через localhost
- Контейнеры в поде также разделяют данные в volumes

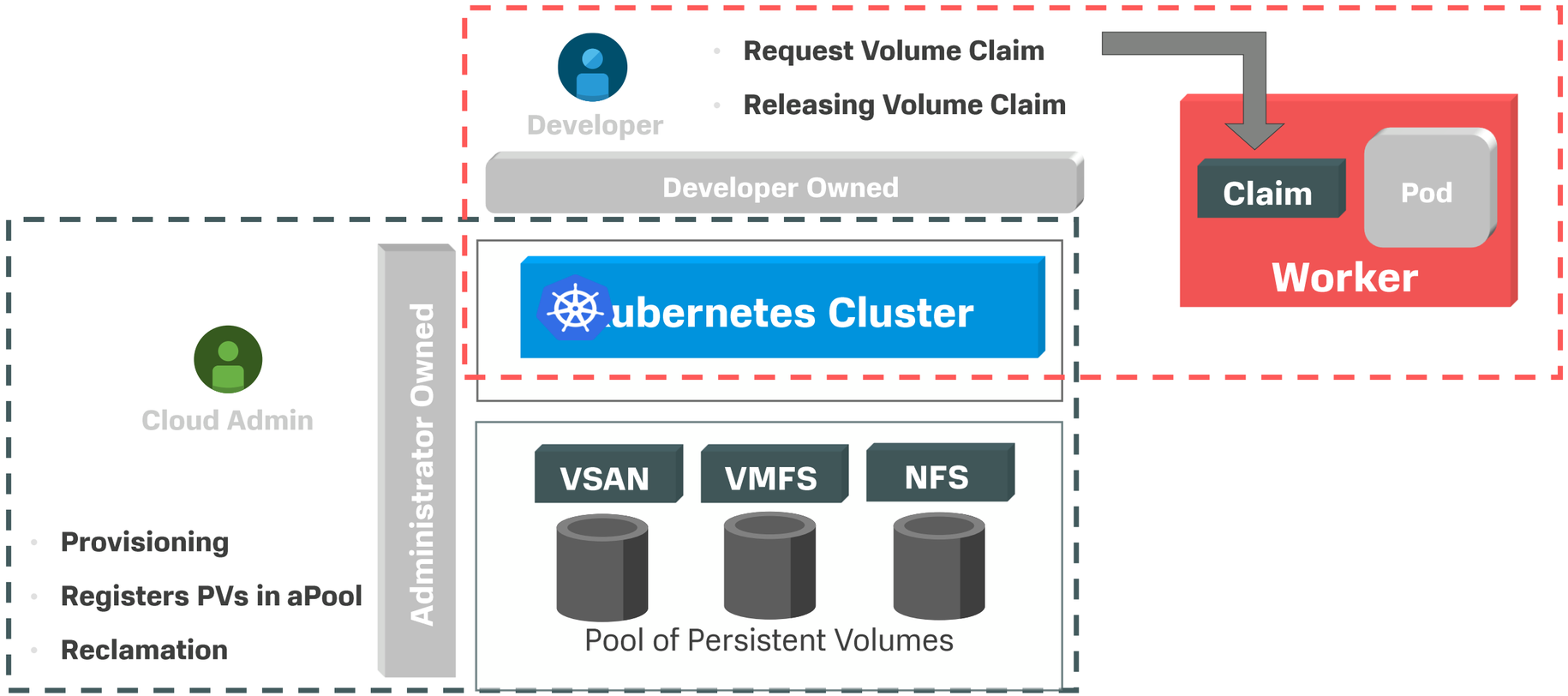


Labels

- Метка – это пара key/value, привязанная к подам и содержащая пользовательские атрибуты
- Можно использовать селекторы меток для выбора нужных подов и применения Services или Replication Controllers к ним
- Метки могут быть добавлены к объектам при создании и изменены в любой момент



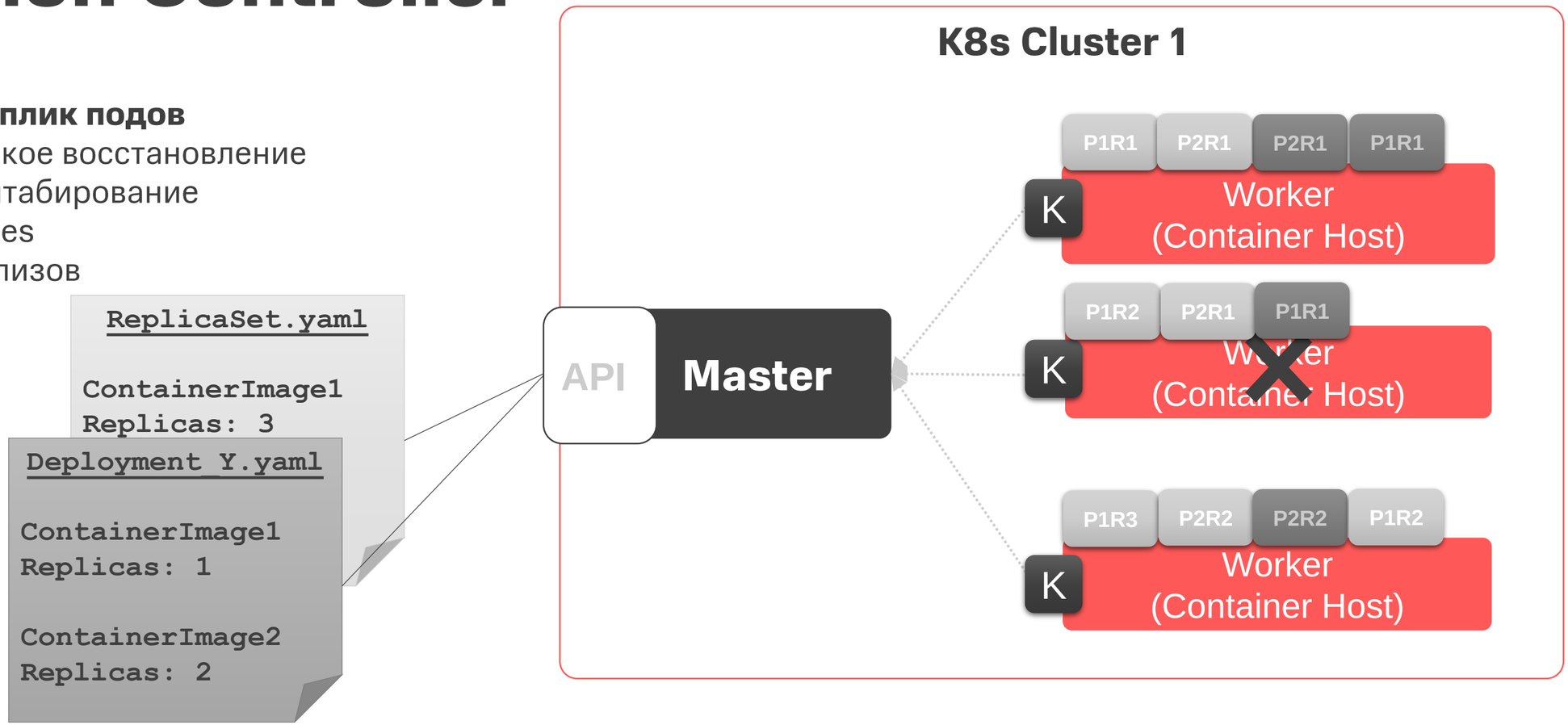
Persistent Volume Claim



Replication Controller

Преимущества реплик подов

- Автоматическое восстановление
- Ручное масштабирование
- Rolling Updates
- Контроль релизов



StatefulSet

Путь запуска **последовательно** реплик подов.
Позволяет работать подам в clustered mode

- Master/Slave приложения

Важно для приложений, которым нужны:

- Фиксированные и уникальные сетевые идентификаторы
- Фиксированное persistent storage
- Развертывание и масштабирование по запросу

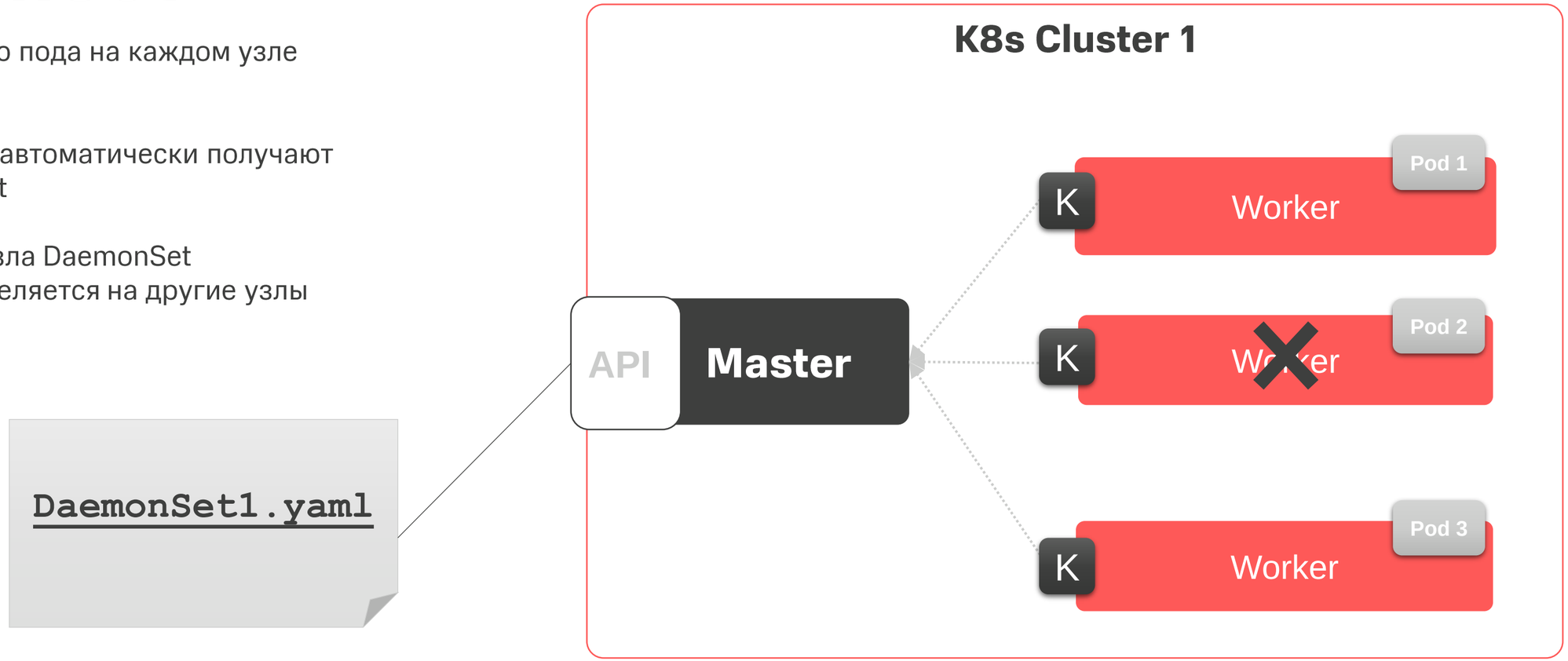
Необходим Headless service

Примеры:

- Zookeeper, Cassandra, etcd, MySQL, etc

DaemonSet

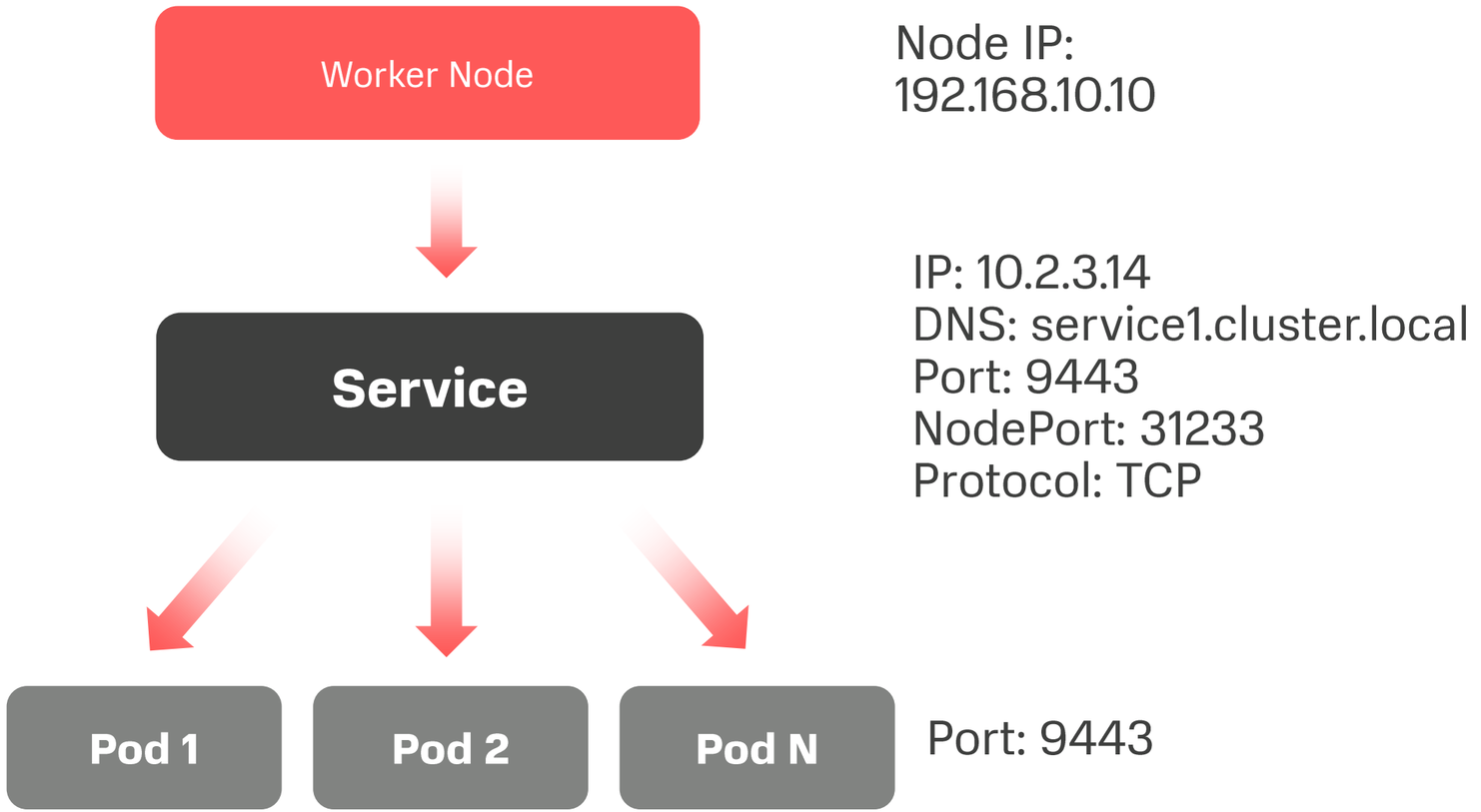
- Запускает копию пода на каждом узле в кластере
- Все новые узлы автоматически получают поды DaemonSet
- При удалении узла DaemonSet не перераспределяется на другие узлы



Services

Services Types
- ClusterIP
- NodePort
- Loadbalancer

Service Discovery
- DNS
- Environmental variables

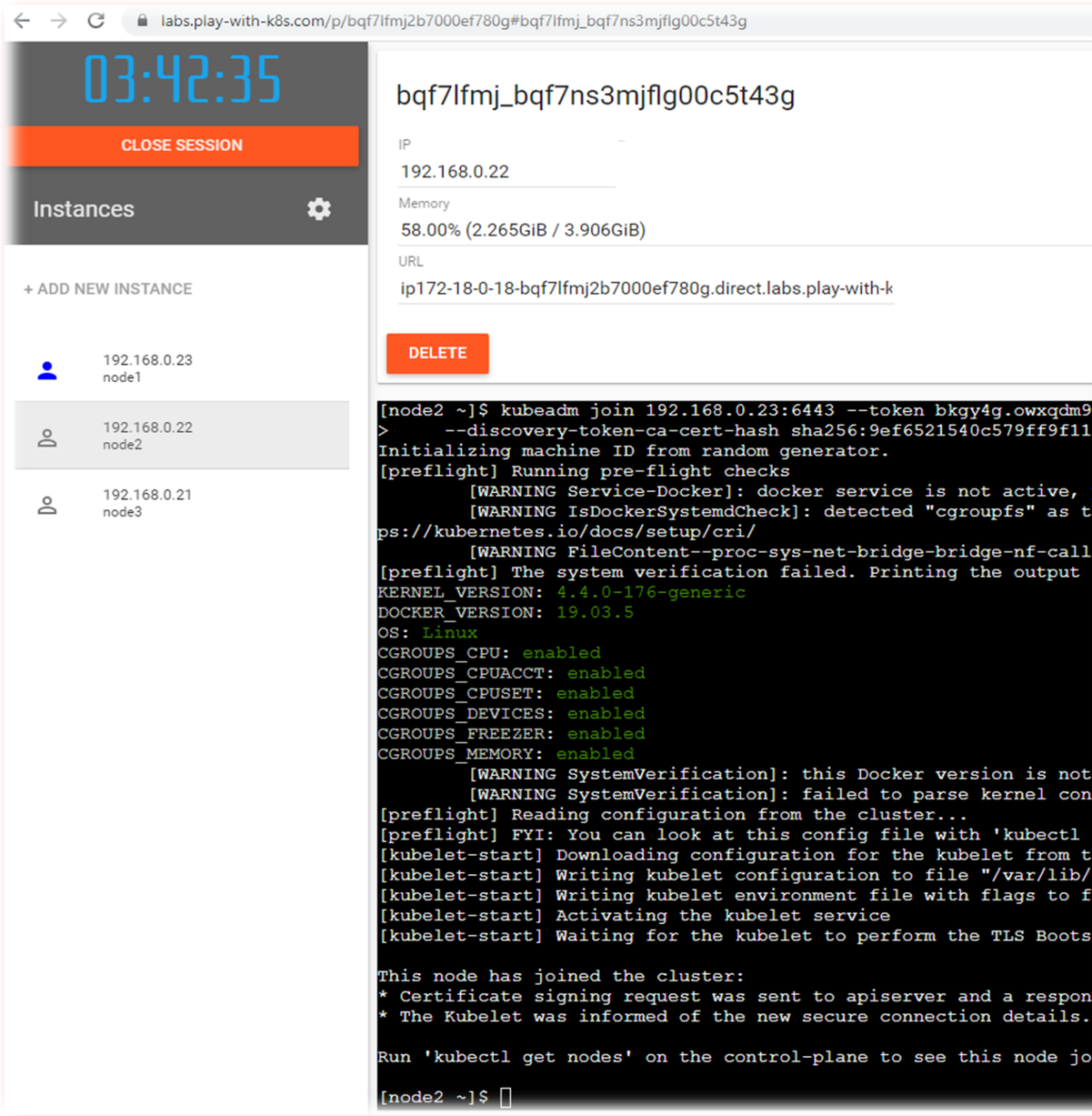


Закрепить материал

Лабораторные работы

Docker/Kubernetes https://labs.play-with-k8s.com/
Hands-On Labs https://labs.hol.vmware.com/
ModernAppsNinja Courses from VMware
https://modernapps.ninja
Learn Kubernetes. From Experts. For Free
https://kube.academy/

K8S https://kubernetes.io/
Cloud Native Apps: https://blogs.vmware.com/cloudnative



Спасибо!