Сетевые возможности
Deckhouse Kubernetes Platform

Онлайн-курс | 5 дней

Аудитория курса

- DevOps-инженеры
- Системные инженеры Kubernetes
- Сетевые инженеры

Цели курса

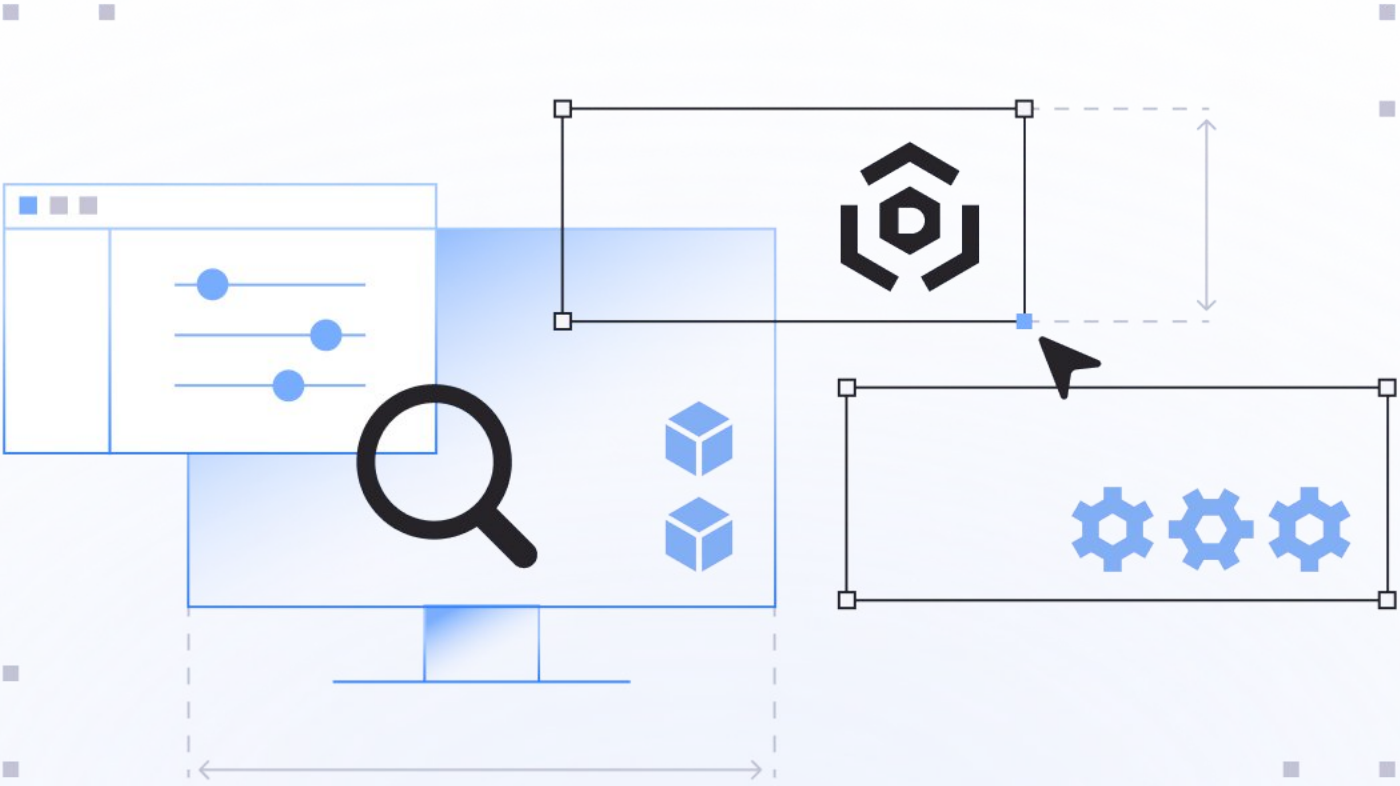
- Получить знания о принципах работы и применении сетевых возможностей Deckhouse Kubernetes Platform (DKP)
- Приобрести базовые навыки для администрирования и эксплуатации сетевых инструментов DKP

Требования к участникам

- Знать Linux на уровне пользователя
- Знать основные понятия и сущности Kubernetes (pod, deployment, service, ingress)
- Знать принцип работы и основные понятия TCP/IP (уровень CCNA)
- Уметь работать с утилитой kubectl

Формат

- Курс состоит из теоретического материала и практической части с выполнением лабораторных работ на учебном стенде
- Теоретический материал включает вебинары и онлайн-демонстрации работы в кластере



План работы

Сетевые возможности Deckhouse Kubernetes Platform

Тема	Структура
1. Организация сетевого взаимодействия в инфраструктуре кластера Deckhouse Kubernetes Platform	**Цель:** получить знания об основах организации сетевого взаимодействия для функционирования кластера DKP в статической и облачной инфраструктуре. **Теория:** • Подготовка статической инфраструктуры для DKP: требования к узлам, сетевая связность • Межсетевое экранирование между узлами • Internal-сеть кластера: StaticClusterConfiguration, internalNetworkCIDRs • Общие параметры кластера (ClusterConfiguration) • Управление статическими маршрутами и правилами ip rule • Настройка сетевого шлюза из узлов: модуль network-gateway, DHCP, SNAT • Подготовка облачной инфраструктуры для DKP: clusterType: Cloud, <CLOUD_PROVIDER>ClusterConfiguration • Схемы размещения (layout) для облачных провайдеров **Практика:** инсталляция кластера DKP, настройка сетевых параметров, управление маршрутами. **Продолжительность модуля:** 8 ак. часов
2. Внутренняя сеть. Container Network Interface	**Цель:** получить знания о принципах работы и назначении Container Network Interface (CNI), особенностях и настройках CNI Cilium, внутренней балансировке трафика, а также об особенностях работы DNS в кластерах DKP. **Теория:** • Принципы работы и назначение CNI. CNI Cilium в DKP: архитектура, основные компоненты, сетевые интерфейсы, взаимодействие подов (локальное и межузловое) • Настройки модуля CNI Cilium: режимы туннелей (Disabled, VXLAN), режимы балансировщика eBPF (SNAT, DSR, Hybrid) • Локальная балансировка трафика (Services): типы (ClusterIP, NodePort, LoadBalancer, Headless, ExternalName), Endpoints, EndpointSlice, сервисы без селекторов • Публикация подов за пределы кластера: hostPort, NodePort, LoadBalancer • Модуль service-with-healthchecks DKP • Кластерный DNS: CoreDNS, NodeLocalDNS, DNS-политики в поде, настройки модуля kube-dns

	Практика: создание различных типов сервисов, работа с DNS. **Продолжительность модуля:** 8 ак. часов
3. Сетевая балансировка входящего трафика. MetalLB	**Цель:** научиться управлять входящим в кластер DKP трафиком L4 и осуществлять его балансировку. **Теория:** • Сервисы типа LoadBalancer и NodePort, способы балансировки входящего трафика, особенности использования LoadBalancer в облачной инфраструктуре • Механизм LoadBalancer для сервисов в кластерах bare metal, принцип работы MetalLB • Режим Layer 2 MetalLB в кластерах DKP: анонс, спикеры, выбор узла-владельца • Улучшенный режим L2 от Deckhouse • Режим BGP MetalLB в кластерах DKP: анонс маршрутов, взаимодействие с сетевым оборудованием **Практика:** настройка и использование MetalLB в режиме BGP LoadBalancer, создание и использование сервисов типа LoadBalancer в статической инфраструктуре. **Продолжительность модуля:** 8 ак. часов
4. Прикладная балансировка входящего трафика. Управление исходящим трафиком	**Цель:** научиться управлять прикладным входящим и исходящим трафиком кластера DKP и осуществлять их балансировку. **Теория:** • Прикладная балансировка входящего трафика с помощью Ingress NGINX Controller: назначение, архитектура, IngressNginxController в DKP • Ingress-ресурсы, Ingress Class, правила (rules), аннотации, TLS и работа с cert-manager • Инлеты Ingress NGINX Controller • Gateway API как замена Ingress. Модуль alb DKP • Публикация приложений с Gateway API: HTTP/HTTPS с TLS-терминацией и cert-manager, маршрутизация по path/header/method/query, weighted canary, TLS passthrough, gRPC • Управление исходящим трафиком с помощью Cilium Egress Gateway от DKP: базовый режим, настройка в DKP • Режим Egress Gateway с Virtual IP • Egress Gateway Policy от DKP: политики перенаправления прикладного трафика на определенные egress-шлюзы **Практика:** настройка NGINX Ingress Controller с различными инлетами и работа с ними, создание и настройка Ingress-ресурсов и сертификатов, включение модуля alb и создание шлюзов Gateway API (инлеты LoadBalancer и HostPort), публикация приложений через Gateway API (HTTP, HTTPS с cert-manager, маршрутизация по path/header и weighted canary), настройка

	egress-шлюза и политик перенаправления исходящего трафика. **Продолжительность модуля:** 8 ак. часов
5. Сетевая безопасность и инструменты troubleshooting'а сетевых компонентов Deckhouse Kubernetes Platform	**Цель:** научиться создавать и применять сетевые политики для пространств имен и других объектов DKP, реализовывать шифрование внутрикластерного трафика и диагностировать сетевые компоненты DKP. **Теория:** • Сетевая сегментация: необходимость, Network Policy Kubernetes, CiliumNetworkPolicy • Сегментация Ingress-контроллера, MetalLB, Egress Gateway • Шифрование внутреннего трафика на основе Istio в DKP: архитектура Istio, sidecar-инъекция, подготовка Ingress-контроллера • Активация mTLS, режимы PeerAuthentication (STRICT/PERMISSIVE/DISABLE), иерархия применения • Политики авторизации (AuthorizationPolicy) • Troubleshooting сетевых компонентов DKP: CNI Cilium, MetalLB, Ingress NGINX, Egress Gateway, NetworkPolicy/CiliumNetworkPolicy, Istio, методология диагностики **Практика:** создание сетевых политик для различных объектов кластера, шифрование сетевого трафика с помощью Istio, создание политик авторизации. **Продолжительность модуля:** 8 ак. часов