

// Jivops

VKS en Producción

Guía completa — 20 lecciones

Fundamentos · Red y almacenamiento · Seguridad y plataforma · Operación

Contenido

Módulo 1: Fundamentos y arquitectura

- Lección 01 — Qué es VKS y qué exige producción
- Lección 02 — El Supervisor y la arquitectura
- Lección 03 — vSphere Namespaces, cuotas y permisos
- Lección 04 — Clústeres VKS y ClusterClass
- Lección 05 — Versiones y releases de Kubernetes

Módulo 2: Red y almacenamiento

- Lección 06 — Redes del Supervisor y de los clústeres
- Lección 07 — CNI: Antrea y Calico
- Lección 08 — Almacenamiento y CSI
- Lección 09 — Balanceo e ingress
- Lección 10 — VM classes y dimensionamiento

Módulo 3: Seguridad y plataforma

- Lección 11 — Identidad y acceso
- Lección 12 — Seguridad de pods
- Lección 13 — Registro de imágenes y cadena de suministro
- Lección 14 — Paquetes estándar de VKS
- Lección 15 — Copias de seguridad y recuperación

Módulo 4: Operación en producción

- Lección 16 — Monitorización y registros
- Lección 17 — Actualizaciones
- Lección 18 — Escalado y alta disponibilidad
- Lección 19 — Resolución de problemas
- Lección 20 — Proyecto final: plataforma Kubernetes de producción sobre vSphere

Módulo 1: Fundamentos y arquitectura

Lección 01 — Qué es VKS y qué exige producción

El servicio de Kubernetes integrado en vSphere de VMware (antes Tanzu Kubernetes Grid Service): permite crear y operar clústeres Kubernetes conformes desde un Supervisor, gestionando sus nodos como máquinas virtuales y aprovechando la red, el almacenamiento y la identidad de vSphere.

Alta disponibilidad, seguridad, copias de seguridad y monitorización, además de actualizaciones controladas.

Lección 02 — El Supervisor y la arquitectura

El clúster de Kubernetes que se habilita sobre un clúster de vSphere (Workload Management) y actúa como plano de control de gestión: ejecuta los controladores que crean y administran los clústeres VKS y los vSphere Namespaces, normalmente con tres máquinas virtuales de control.

Para alta disponibilidad: el Supervisor sigue funcionando (con quórum) si falla un nodo, algo necesario en producción.

Lección 03 — vSphere Namespaces, cuotas y permisos

Una unidad de aislamiento y asignación de recursos en el Supervisor: define cuotas de CPU, memoria y almacenamiento, las políticas de almacenamiento y clases de VM permitidas, y los permisos de usuarios (por ejemplo ver, editar o propietario) para un equipo o proyecto.

Aísla recursos y permisos y limita el consumo de cada equipo con cuotas.

Lección 04 — Clústeres VKS y ClusterClass

Con un objeto declarativo basado en Cluster API (con ClusterClass) dentro de un vSphere Namespace: versión de Kubernetes, clase y número de nodos de control y de los grupos de trabajadores (worker), política de almacenamiento, red y variables; el Supervisor los crea como máquinas virtuales.

Es reproducible y versionable (GitOps) y se automatiza el ciclo de vida.

Lección 05 — Versiones y releases de Kubernetes

Las versiones disponibles se publican como releases (TKr) en una Content Library suscrita; cada release de VKS soporta un rango de versiones de Kubernetes, y hay que respetar la compatibilidad entre vSphere, Supervisor y clústeres antes de actualizar.

Porque deja de recibir soporte y parches de seguridad y la actualización será más difícil.

Módulo 2: Red y almacenamiento

Lección 06 — Redes del Supervisor y de los clústeres

Con NSX, que ofrece redes por namespace, balanceo, microsegmentación y enrutamiento, o con redes de VDS (distributed switch) y un balanceador externo (como NSX Advanced Load Balancer/Avi o HAProxy), más sencillo pero con menos funciones.

Aporta microsegmentación, redes por namespace y más funciones de aislamiento.

Lección 07 — CNI: Antrea y Calico

Antrea es el CNI por defecto y Calico es una alternativa soportada; el CNI da IP a los pods, implementa las network policies y la comunicación entre nodos, y se elige al crear el clúster.

Para aislar pods y permitir solo el tráfico necesario, aplicando mínimo acceso.

Lección 08 — Almacenamiento y CSI

Mediante el driver CSI de vSphere: las políticas de almacenamiento del namespace (por ejemplo vSAN) se exponen como StorageClasses, y los volúmenes persistentes se crean como volúmenes nativos de contenedor visibles en vCenter.

Para fijar rendimiento y disponibilidad (por ejemplo FTT en vSAN) según la carga.

Lección 09 — Balanceo e ingress

Los servicios tipo LoadBalancer obtienen una IP del balanceador de la plataforma (NSX, Avi o HAProxy); para HTTP se añade un ingress controller (por ejemplo Contour) con certificados, y se limitan rangos de IP y acceso.

Permite publicar varios servicios HTTP detrás de una sola IP, con rutas y TLS.

Lección 10 — VM classes y dimensionamiento

Con VM classes que definen CPU y memoria de las máquinas virtuales de control y de trabajo, asignadas al namespace; en producción se separan nodos de control y workers, se usan reservas de recursos para cargas críticas y se calcula el espacio según pods y márgenes de fallo.

Margen para el fallo de un nodo, las actualizaciones por rodado, picos y crecimiento.

Módulo 3: Seguridad y plataforma

Lección 11 — Identidad y acceso

Con vCenter Single Sign-On (o un proveedor de identidad externo vía federación), usando el plugin de login para obtener acceso a los namespaces y clústeres; los permisos en el namespace se traducen a RBAC de Kubernetes, y las cuentas de servicio cubren la automatización.

Centraliza altas, bajas y grupos con auditoría clara.

Lección 12 — Seguridad de pods

Aplicando Pod Security Admission con perfiles baseline o restricted por namespace, ejecutando sin privilegios, con contextos de seguridad mínimos, network policies y límites de recursos, y controlando qué imágenes se pueden ejecutar.

Porque un compromiso puede permitir acceso al nodo y escalar privilegios.

Lección 13 — Registro de imágenes y cadena de suministro

Con un registro privado como Harbor (como servicio del Supervisor o propio) con escaneo de vulnerabilidades, firma de imágenes y control de acceso por proyecto, y fijando las imágenes por digest para despliegues reproducibles.

Permite detectar vulnerabilidades antes de desplegar y parchear a tiempo.

Lección 14 — Paquetes estándar de VKS

Un catálogo de software de plataforma instalable y gestionado sobre los clústeres: cert-manager, Contour, external-dns, Fluent Bit, Prometheus, Grafana y otros, con versiones compatibles con la release de Kubernetes y configuración declarativa.

Porque están probados con la versión de Kubernetes, tienen soporte y simplifican actualizar.

Lección 15 — Copias de seguridad y recuperación

Con copias de seguridad de los objetos de Kubernetes y los volúmenes (por ejemplo con Velero y el plugin de datos de vSphere), copias del Supervisor y de vCenter, y pruebas periódicas de restauración incluso a otro clúster.

Para comprobar que realmente funcionan y medir cuánto tardan en recuperar.

Módulo 4: Operación en producción

Lección 16 — Monitorización y registros

Combinando métricas de la infraestructura vSphere (hosts, datastores, vSAN) con las del clúster (Prometheus y Grafana) y logs enviados con Fluent Bit a un sistema central, más alertas sobre nodos, pods, almacenamiento y la salud del Supervisor.

Nodos no listos, falta de espacio en almacenamiento, pods en error y salud del Supervisor.

Lección 17 — Actualizaciones

En orden: comprobar compatibilidad, actualizar vCenter y ESXi, luego el Supervisor y después los clústeres VKS mediante un rodado de nodos (rolling update), probando antes en un entorno no productivo y teniendo copias recientes.

La matriz de compatibilidad, tener copia, probar en un entorno previo y fijar ventana.

Lección 18 — Escalado y alta disponibilidad

Escalando el número de workers manualmente o con Cluster Autoscaler, repartiendo los nodos por zonas de vSphere (vSphere Zones) y dominios de fallo, con tres nodos de control, y usando afinidad y presupuestos de interrupción para las aplicaciones.

Para mantener un mínimo de réplicas disponibles durante actualizaciones o mantenimientos.

Lección 19 — Resolución de problemas

El estado de los objetos de Cluster API en el Supervisor (cluster, máquinas y eventos), la cuota y las clases del namespace, la disponibilidad de recursos en vSphere (CPU, memoria, datastore), la red y el balanceador, y la Content Library con la imagen del release.

Falta de cuota o de recursos en vSphere, como CPU, memoria o espacio en el datastore.

Lección 20 — Proyecto final: plataforma Kubernetes de producción sobre vSphere

Supervisor de tres nodos en tres zonas, red con NSX o VDS más balanceador, namespaces por equipo con cuotas y permisos vía SSO, clústeres declarativos con tres nodos de control y workers por zonas, almacenamiento por políticas con CSI, Harbor, Pod Security, paquetes de plataforma, copias con Velero, monitorización y actualizaciones por fases.

Crear el namespace con cuotas, clases de VM y políticas de almacenamiento, y asignar permisos mediante SSO.