

kind Cluster Baseline Report – ingress-lab

1. Identificación general

- **Tipo de cluster:** kind (Kubernetes in Docker)
 - **Contexto activo kubeconfig:** kind-advanced-lab
 - **Fuente:** `kubectl config current-context`
 - **Ubicación del lab:** `~/repos/ingress-lab`
 - **Rol del cluster:** Platform / Ingress / GitOps lab (baseline reusable)
-

2. Patrón de despliegue (GitOps)

- **Herramienta:** Argo CD
 - **Patrón:** App of Apps
 - **Aplicación root:** ingress-lab-root
 - **Namespace:** argocd
 - **Repositorio Git:**
<https://github.com/leonardocastanodiaz/ingress-lab.git>
 - **Path root:** bootstrap
 - **Modo de sincronización:**
 - Auto-sync: ☒
 - Prune: ☒
 - Self-heal: ☒
-

3. Capas del laboratorio

3.1 Platform Layer

- **ArgoCD App:** platform-ingress-nginx
- **Tipo:** Helm chart
- **Chart:** ingress-nginx
- **Versión:** 4.10.1
- **Namespace:** ingress-nginx
- **Ingress Controller Service:** NodePort
- **IngressClass:**
 - Nombre: nginx

- Default: `true`
-

3.2 Application Layer (Demo)

- **ArgoCD App:** `demo-apps`
- **Namespace:** `demo` (auto-creado)
- **Gestión:** Kustomize
- **Path:** `apps/demo/kustomization.yaml`

Despliegues:

- `app1`
 - Imagen: `hashicorp/http-echo`
 - Service: `app1`
 - `app2`
 - Imagen: `hashicorp/http-echo`
 - Service: `app2`
-

3.3 Routing Layer

- **ArgoCD App:** `demo-routing`
- **Namespace:** `demo`
- **Recurso:** Ingress (`demo-ingress`)
- **Host:** `demo.local`

Rutas configuradas:

- `/app1` → Service `app1`
- `/app2` → Service `app2`

Características:

- Regex habilitado
 - `rewrite-target: /$2`
-

4. Comando base de arranque del lab

`kubectl apply -f bootstrap/root-app.yaml`

Este comando inicializa completamente el laboratorio mediante Argo CD siguiendo el patrón App of Apps.

5. Versiones de herramientas (Baseline técnico)

- **Docker Client:** 28.0.4
 - **Docker Server:** 28.0.4
 - **kind:**
 - v0.31.0
 - Go: 1.25.5
 - OS/Arch: darwin/arm64
 - **kubectI Client:** v1.35.0
 - **Kustomize (integrado en kubectI):** v5.7.1
-

6. Notas de diseño (importantes para GPT / DevX)

- El cluster ya contiene **infraestructura funcional** (Ingress + ArgoCD).
 - Este baseline debe **reutilizarse**, no recrearse.
 - Jenkins, CI y pipelines deben:
 - Integrarse en este cluster existente
 - Usar contenedores efímeros para builds
 - No duplicar componentes ya desplegados
 - Este archivo debe considerarse **fuentes de verdad contextual** para asistentes GPT y automatizaciones futuras.
-

Estado:  Baseline estable

Uso esperado: contexto persistente para labs, entrevistas y diseño DevX / Platform