

Inception of Things

42 School Project — Full Architecture Overview

P1 K3s Cluster

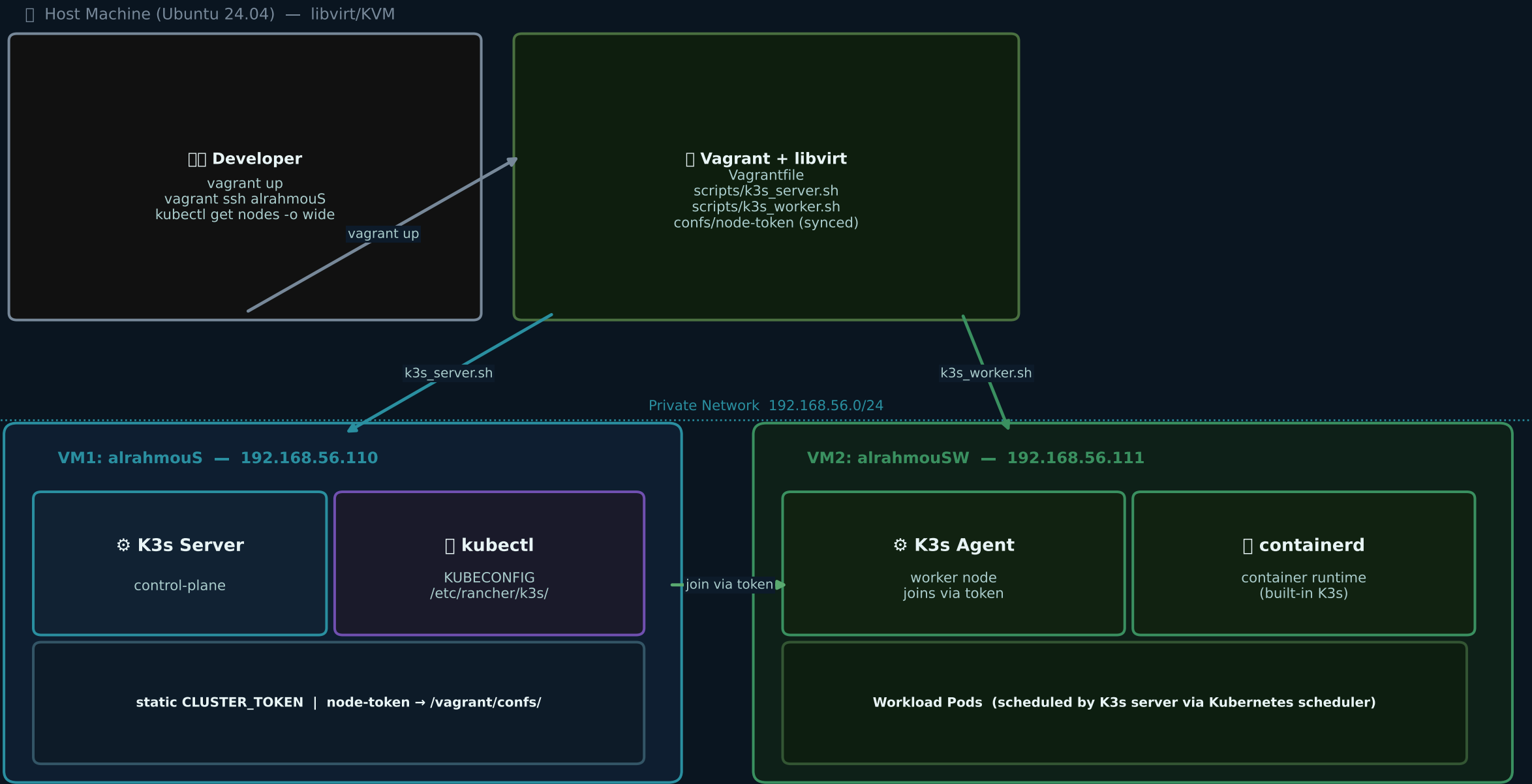
P2 Ingress Apps

P3 GitOps / ArgoCD

Bonus GitLab + ArgoCD

Part 1 — K3s Cluster with Vagrant

2 VMs · libvirt/KVM · K3s server + worker



Part 1 — Key Facts & Defense Commands

K3s · Vagrant · 2 VMs

VMs

2 VMs: alrahmouS (server) + alrahmouSW (worker)

IPs

192.168.56.110 (server) 192.168.56.111 (worker)

K3s mode

Server: --cluster-init Agent: K3S_URL + K3S_TOKEN

Token

Static CLUSTER_TOKEN in Vagrantfile — written to /vagrant/confs/node-token

Provisioning

Fully automated via shell provisioners (k3s_server.sh / k3s_worker.sh)

Verify

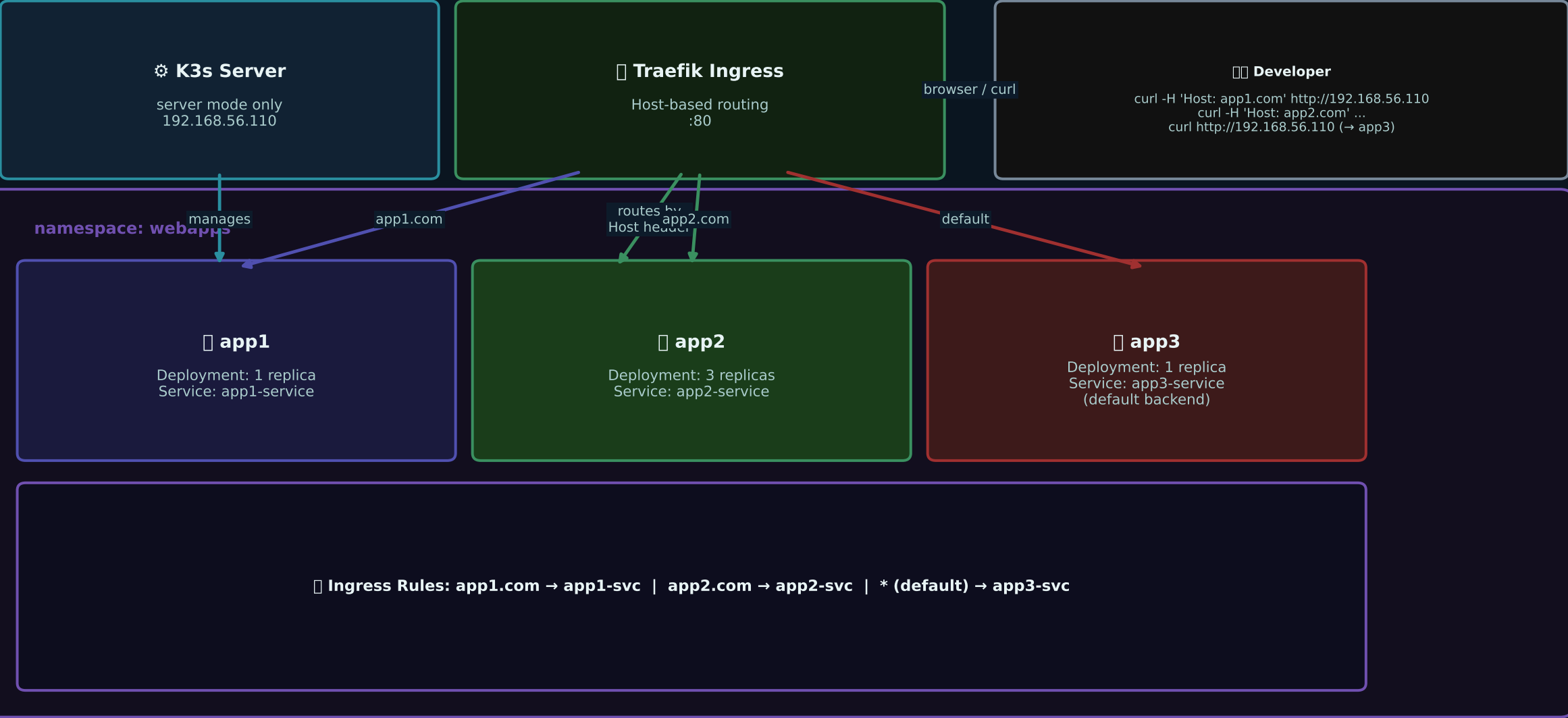
kubectl get nodes -o wide → 2 nodes, both Ready

vagrant up → vagrant ssh alrahmouS → kubectl get nodes -o wide

Part 2 — K3s + Traefik Ingress + 3 Apps

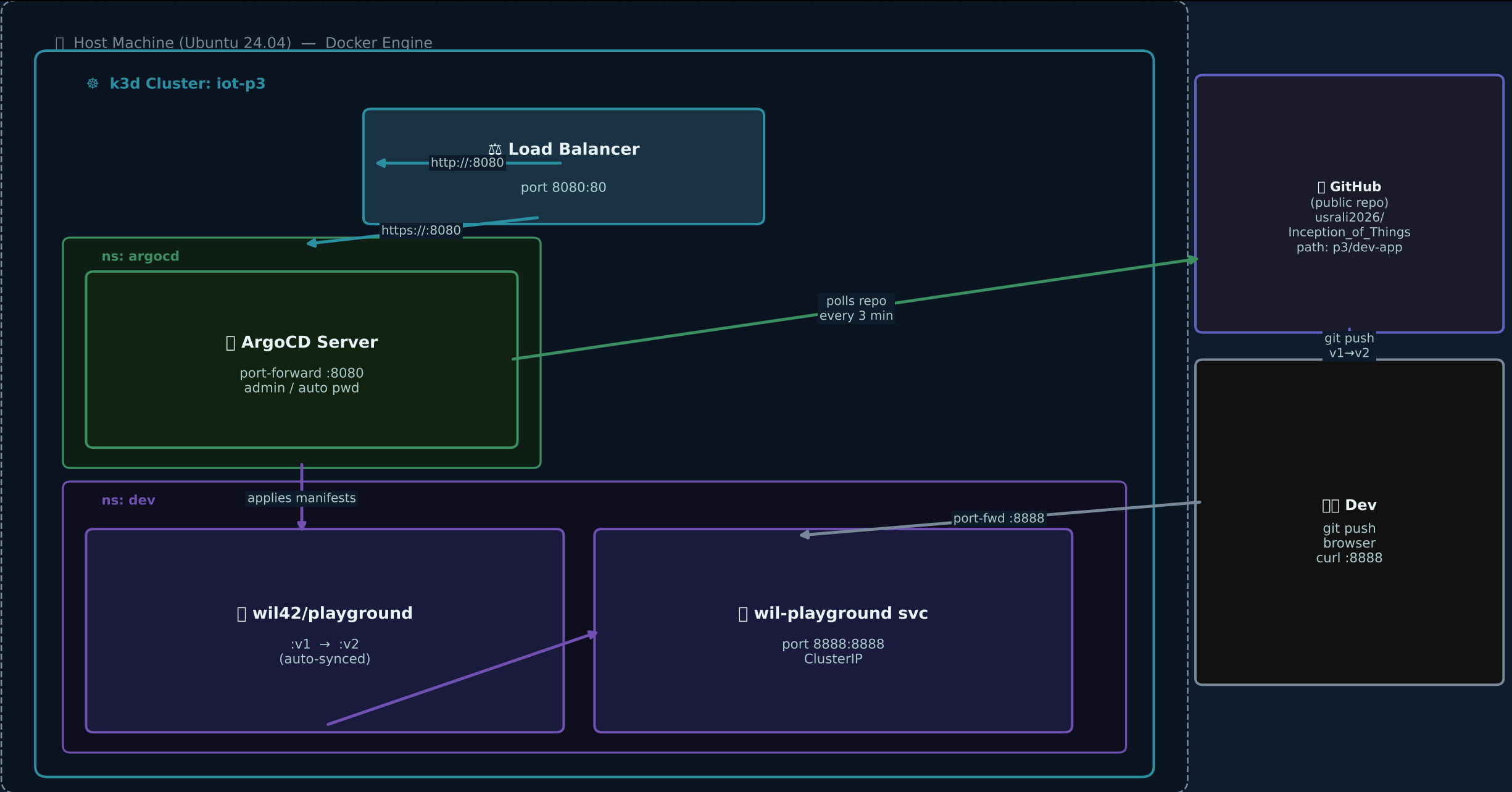
1 VM · webapps namespace · Host-based routing

Host Machine — VM: alrahmouS 192.168.56.110



VM	Single VM alrahmouS at 192.168.56.110 (libvirt)
K3s	Server mode only — no worker node needed
Ingress	Traefik (built-in K3s) — Host-header based routing
app1	1 replica — Host: app1.com → app1-service:80
app2	3 replicas — Host: app2.com → app2-service:80
app3	1 replica — default backend (any other Host)
/etc/hosts	192.168.56.110 app1.com 192.168.56.110 app2.com (on host)

```
curl -H 'Host: app1.com' http://192.168.56.110 → app1
curl -H 'Host: app2.com' http://192.168.56.110 → app2
curl http://192.168.56.110 → app3
```

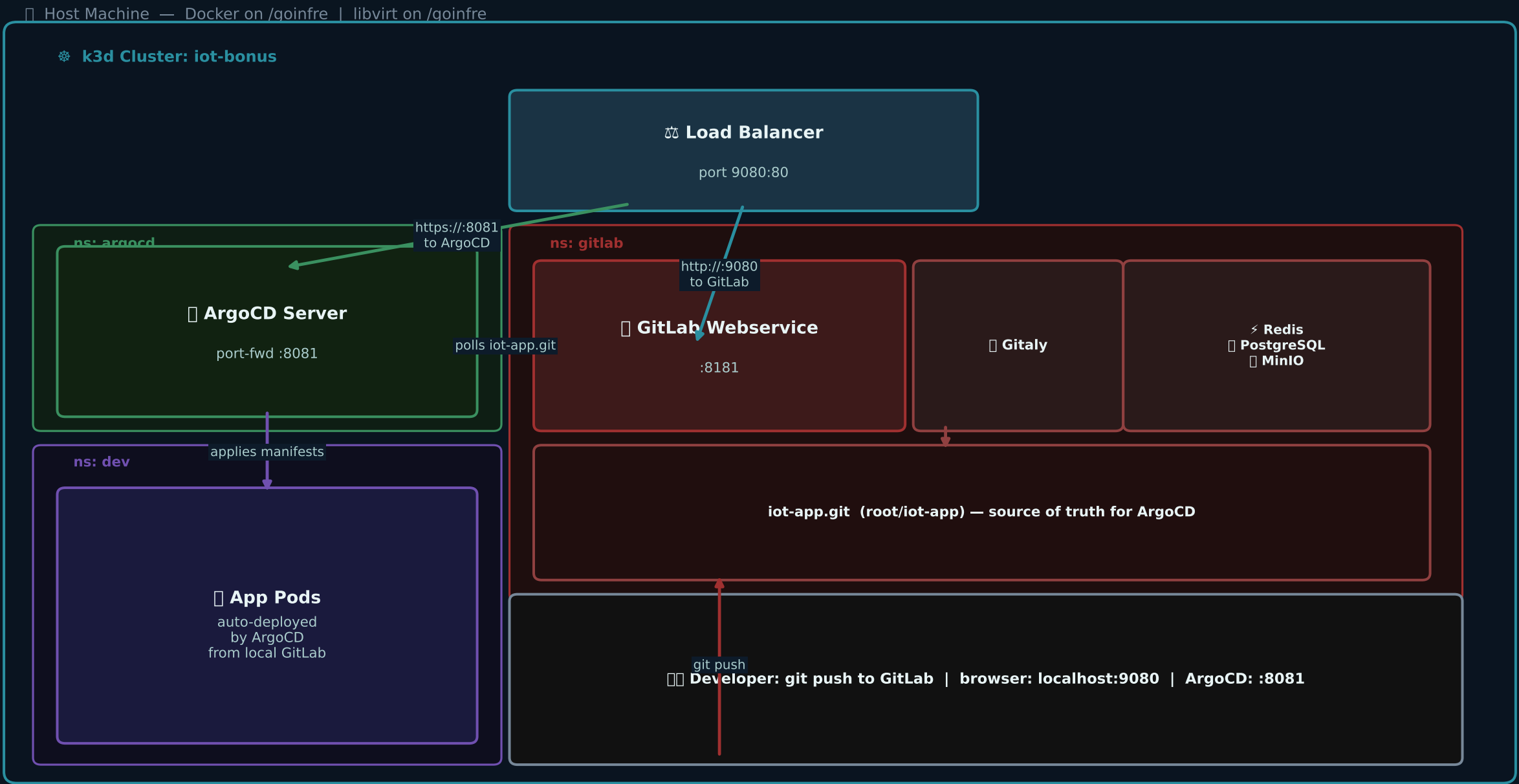


Cluster	k3d iot-p3 — 1 server + 2 agents (Docker containers)
Namespaces	argocd (ArgoCD) + dev (app pods)
App image	wil42/playground:v1 → change to :v2 in Git to prove GitOps
Repo	Public GitHub: usrali2026/Inception_of_Things path: p3/dev-app
ArgoCD sync	Auto-sync enabled — detects Git drift every ~3 min, reconciles
Access	ArgoCD: port-forward svc/argocd-server 8080:443 App: port-fwd :8888
GitOps flow	git push tag v1→v2 → ArgoCD detects drift → pod rolls over

```
kubectl get pods -n argocd | kubectl get pods -n dev
kubectl port-forward svc/argocd-server -n argocd 8080:443
kubectl -n dev port-forward svc/wil-playground 8888:8888
curl http://127.0.0.1:8888 → v1 (edit Git → v2) curl again → v2
```

Bonus — Self-Hosted GitLab + ArgoCD

k3d · gitlab ns · argocd ns · dev ns · GitOps (no GitHub)



Cluster	k3d iot-bonus — 1 server + 2 agents port 9080:80
Namespaces	argocd + dev + gitlab (Webservice, Gitaly, PostgreSQL, Redis, MinIO)
GitLab	Fully local — no GitHub needed root / <initial-password>
ArgoCD repo	http://gitlab-webservice-default.gitlab.svc.cluster.local:8181/root/iot-app.git
GitOps flow	git push to local GitLab → ArgoCD polls → auto-deploys to dev ns
Access GitLab	http://localhost:9080 (k3d port 9080:80)
Access ArgoCD	kubectl port-forward svc/argocd-server -n argocd 8081:443
vs P3	P3 uses GitHub (external). Bonus replaces GitHub with self-hosted GitLab

```
bash bonus/scripts/setup.sh # full deploy
kubectl get secret gitlab-gitlab-initial-root-password -n gitlab -o jsonpath='{.data.password}' | base64 -d
kubectl get pods -n gitlab | kubectl get pods -n argocd | kubectl get pods -n dev
```

Full Comparison — P1 → P2 → P3 → Bonus

Side-by-side overview

	P1	P2	P3	Bonus
VMs	2 (server+worker)	1 (server only)	0 (Docker)	0 (Docker)
K8s distro	K3s bare	K3s bare	K3s in k3d	K3s in k3d
Provisioner	Vagrant/libvirt	Vagrant/libvirt	k3d CLI	k3d CLI
Namespaces	—	webapps	argocd+dev	argocd+dev +gitlab
Apps	None	app1/2/3	wil42/playground	custom app
Ingress	None	Traefik (Host)	LB :8080	LB :9080
GitOps	☐	☐	☐ GitHub	☐ GitLab local
CI/CD	☐	☐	ArgoCD	ArgoCD + GitLab