All about Kubernetes and Cloud Native

Anjul Sahu

2025-09-27

Agenda

- About me
- Some history of Kubernetes
- Basics
- More than basics
- Q&A time

About Me

- Anjul Sahu
- Founder and CEO, CloudRaft
- Started in December 2022
- Specializing in Kubernetes and Cloud Native technologies, AI Infrastructure, Databases and Observability
- 2008 - 2022: Worked at Accenture, InfraCloud, and Lummo

Excited to learn about Kubernetes?

What is Kubernetes?

“*Kubernetes is the orchestration system.*”

“*Kubernetes is becoming like Linux. It is everywhere but abstracted, invisible to you.*”

“*Kubernetes is the operating system of Cloud.*”

Why everyone love Kubernetes?

Why everyone love Kubernetes?

- Most popular and biggest open source project after Linux
- Autoscaling (horizontal, vertical, more complex strategies with keda, custom)
- scheduling is easy and managed
- operational automation is easy
- proven at scale and global standard, battle tested at Google (Borg)
- customization flexibility
- AI/ML ecosystem is growing, Kubernetes is go to orchestration layer

Some basics

- 2023 talk

Beyond Basics

- **Packaging and Deployment** - Helm Charts, Kustomize, ArgoCD (GitOps)
- **Monitoring and Observability** - Prometheus, Grafana, Loki (PLG) stack, or Elastic stack, OpenTelemetry
- **Service Mesh** - Istio, Linkerd, Envoy - for Reliability, Security, and Control
- **Security** - Network Policies (need support from CNI), Dynamic runtime security (tetragon, falco)
- **Storage** - CSI - both block storage and object storage is available
- **Extensibility** - Extend it via custom resource and Operator pattern

Helm

Templating engine that can render Kubernetes YAMLs. Control things using a `values.yaml`

```
.
├── Chart.yaml
├── templates
│   ├── _helpers.tpl
│   ├── configmap.yaml
│   ├── deployment.yaml
│   ├── hpa.yaml
│   ├── ingress.yaml
│   ├── NOTES.txt
│   ├── service.yaml
│   └── tests
│       └── test-connection.yaml
└── values.yaml
```

Example template for Service

```
apiVersion: v1
kind: Service
metadata:
  name: {{ include "service.fullname" . }}
  labels:
    {{- include "service.labels" . | nindent 4 }}
spec:
  type: {{ .Values.service.type }}
  ports:
    - port: {{ .Values.service.port }}
      targetPort: http
      protocol: TCP
      name: http
  selector:
    {{- include "service.selectorLabels" . | nindent
4 }}
```

Kustomize

- To solve the mess of YAMLs, Kustomize was created.
- Learn more at kustomize.io

```yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
namespace: ad-adapter

helmGlobals:
    chartHome: ../../../helm-charts
helmCharts:
  - name: fancy-service
    releaseName: ad-adapter
    version: 0.1.0
    valuesFile: values.yaml
```

ArgoCD (GitOps)

- Save the configuration in git and sync it with a branch
- Any change in cluster is reverted, will sync it back with git
- Idea is to have all infra changes must go through a review process
- Decoupled architecture:
  - CI build the image and stores it in the repo
  - image updater (automatically pull the latest image or filtered image from the repo)
- Advanced rollout strategies: canary and blue-green
- ArgoCD vs FluxCD - Read more at https://www.cloudraft.io/blog/argocd-vs-fluxcd

Monitoring and Observability

- Monitoring vs Observability
- **Observability** = Metrics + Logs + Traces + Profiling ; treat like a black box
- **Metrics**: Prometheus has been a standard timeseries database used to store metrics
- **Logs** - Elastic or Loki etc
- **Traces** - Tempo, Jaeger
- **Profiling** - pyroscope, parca
- ***OpenTelemetry*** is a *industry standard*
  - Manually instrument or auto-instrument, agent sends to backend
  - An OTel pipeline has receivers, processors, and exporters or sinks

There are many options in the industry, choose what is right for your use case.

Read more - https://www.cloudraft.io/blog/guide-to-observability

Service Mesh and API Gateway

- Istio, Linkerd are go to options
- 3 main purposes of **Service Mesh**
  - **observability** - apm like features
  - **security** - mTLS, zero trust
  - **traffic control / reliability** - rate limiting, circuit breaker, timeouts, retries
- **API Gateway** - traefik, kong, envoy api gateway

Read more: https://www.cloudraft.io/blog/kubernetes-api-gateway-comparison

Security

Role-based access

- start with least and keep it granular and use service account

Network policy

- multi-tenancy, isolation, need cni support

Runtime threat detection

- falco or tetragon based on eBPF
- cryptomining
- privilege escalation
- unexpected network connection etc

Policy-as-code - opa, kyverno

- disallow latest tag in images
- enforce requests and limits
- restrict external IPs

Supply chain security

- signing, provenance and authorization

Storage

- Various solutions: longhorn, openebs, rook-ceph, portworx
- storage class
  - different tiers/classes
  - encryption etc
- PV and PVC
- need csi driver
- Backup & recovery: Velero, Kasten K10

Extensibility

- custom resources - extension of k8s API
- operators - control loop to manage CRs
  - kubebuilder is popular
  - operator framework
- **Example**: I want to run Postgres on Kubernetes, there is a CloudNativePG operator. That automates most of the day to day operations.

Kubernetes and cloud native is an ocean. It is a journey and best time to start is NOW.

Questions

Feel free to ask any question around Kubernetes and cloud native.

Follow me on LinkedIn

Keep yourself updated at www.cloudraft.io/blog

Thank you!