No More Chaos - Inspect Kubernetes at Scale

Jie Chen - Alibaba Cloud
Jim Ma - Ant Financial
Jie Chen
• Senior Engineer at Alibaba Cloud
WeChat: anole2010
Email: hantang.cj@alibaba-inc.com

Jim Ma
• Engineer II at Ant Financial
WeChat: majinjing3
Email: chuxian.mjj@alipay.com
• Why We Need Inspection on Kubernetes Clusters?
• Considerations on an Inspection Service
• Build A Self-Healing Inspection System
• The Future
We have A Large Scale Kubernetes Cluster

- Nodes (>10k)
  - Types of Hardware
  - Types of OS
  - Types of Kubelet
- Pods (>100k)
  - Jobs (>10k)
- Controllers (>100)
  - Custom Resource Types (>100)
So many Kubernetes resources and custom resources

Does the cluster really keep eventual consistency?
Cases

- IP Conflict
  - Service Interrupt
- Dangling Resources
  - Pods
  - PVs
  - CRDs
- Inconsistent with External System
  - CMDB
  - IPAM
We need to dive into our Kubernetes cluster to inspect all the resources
Considerations on An Inspection Service

• Efficiency
  • Capable to inspect ten of thousands of nodes
  • Capable to inspect hundreds of thousands of pods
  • Capable to inspect hundreds of custom resource (* pods)

• Scalability
  • Support more teams/groups
  • Setup hundreds of inspecting points

• Observation
  • Auditing
  • Monitoring
What We Build in Version 1

- Providers
- Triggers
- Pod
- CRDs
- Task Queue
- IP
- Output
Problems in V1

- One pod per cluster
  - Limited performance per node
  - Resource conflicts among tasks
  - Unable to scale horizontally
- Lack of generality
  - Only an internal closed-loop
Knative
Knative: The Killer Component

- Use Serverless to horizontally scale inspections
- Use CloudEvent to be more scalable
- To be a Framework not a internal service any more
We Build V2 in Serverless with Knative

Event -> Knative Broker Trigger

Send inspect result

Tasks
  Pod Inspector
  IPAM Inspector

Actions
  Self-healing

Providers

RPC

Sub&Pub

DingTalk
How to Build A Complete Inspection system
Some Problems in Business Stability

1) A huge gap between infrastructure and business

2) Lack of global perspective of status inspection

**IP conflict**
- IPAM
- APP
- OPS
- CMDB
- K8S cluster
- Mon
- Container
- Node
- ...
Changes When We Moving to Cloud Native and Kubernetes?

- Imperative
  - Giant Container like VM
  - Just Resource Allocation
- Declarative
  - Flexible Containers in Pod
  - Works more
How to Handle These Problems and Changes?
Boundary of Inspection System

- Application Inspection
  - Trade
  - Search
  - Alipay

- Scheduler System Inspection
  - node
  - Pouchd
  - K8S Core
  - External Dependency
  - K8s Event
  - Ops Event

- Infra Inspection
  - Server
  - IDC
  - Network

- Alerting System

- Monitoring System
We Can Do More about Analysis and Make Decision
Self-healing Inspection System - Overview

Control
- Grafana
- Control Platform
- Observation Platform

Tenant Mgmt
- Gateway
- API Server

Security Ctrl
- Alert Manager
- Self-healing Engine

Inspection Mgmt
- Inspector Engine
- Diagnose

Self-healing Mgmt
- tsdb
- SLS
- MNS
- MongoDB

Product Layer
- K8S Export
- NPD
- CMDB
- IPAM

Access Layer
- Monitoring
- Data Analyze
- Store
- Scrape Source
How to Enable Users do Inspect?
User-driven System Inspecting

Inspection Metrics | Alert Metrics | Self-healing Detail | Overall Optimization Metrics

UI(Graph) | Inspection & Self-healing Control | Monitoring Control | query

Users
Configure & Develop

Configure

Rule & Image

Inspector

Event Center

event

Self-healing Engine

monitor

event
Practices
Core Engine for Self-Healing Inspection System
1) Observation

2) Healing
Future: AIOps-driven Self-Healing Inspection System

Data, Schedule, Control, Ops, Closed Loop

Scheduler System Data Platform
- System Metrics
- Nodes Metrics
- Pods Metrics
- Cluster Events

Control Data
- Logging Info
- Operation Events
- Network Events

Rules Center
- Rules Update

Control Platform
- Cluster Update

Inspection Engine
- Biz scenario
  - Exception Detect
  - Root Case Analyze

Algorithm Framework Services
- Machine Learning
- Combined Computing
- Realtime Data Analyze
- Batch Data Analyze

Event Center
- Event

Self-healing Engine
- Data Feedback
- Self-healing Execute
- Serving

K8S cluster
- Cronjob
- Realtime Process

User UDF
- Building
Contact Us

Jie Chen 陈杰

Jim Ma 马金晶
CNCF x Alibaba
云原生技术公开课

免登陆听课 动手实践课后自测

CKA课程内容同步 阿里云原生最佳实践
Thanks