Karpenter: simplificando o auto provisionamento de nós K8s na AWS e Azure.

2024-10-04

Sobre mim

- Graduado em Engenharia de Software na UFG.
- Mais de 13 anos trabalhando com TI.
- Ajudo empresas novas no mercado com práticas DevOps, arquitetura na nuvem e observabilidade.

Primeiro

Você precisa entender o comportamento de sua aplicação e adequá-la para tal.

Execução

- Stateless: não guarda dados em tempo de execução
- Stateful: guarda dados, ao iniciar vai precisar desses dados.

Vai influenciar no escalonamento da aplicação.

Recurso

- CPU Bound: Cálculos pesados, muito I/O.
- Memory Bound: Datasets, estruturas de dados grandes (hashmap, vetores).
- General Purpose: Um pouco dos 2.

Vai influenciar no provisionamento do nó correto.

Regras pra evitar dor de cabeça desnecessária

Para Horizontal Pod Autoscaler, prepare sua aplicação para ser Stateless.

E para aplicações Stateful use o Vertical Pod Autoscaler.

Obs.: Existem estratégias para escalonar aplicações Stateful de forma horizontal com PVs em RWX.

Dicas de ouro

- SEMPRE configurar Resource Requests para seus workloads.
- Utilizar o prometheus-adapter a fim de ter métricas de tráfego e latência. E para aplicações que rodam em background tem o Kubernetes Event-Driven Autoscaler (KEDA).
- HPA: Entender limite de tráfego (req/s, ops/s) que sua aplicação suporta. Olhar para saturação pode ser uma falsa métrica de escalonar.
- VPA: Olhar para saturação.

Karpenter

Conceitos

- Configurações para comunicação entre o Controller <-> Provider
- CRD NodePool e NodeClass Exemplos
- Schedulling: Resource Request, Node affinity (gpu, cpu), Topology (hostname, zones)
- NodeClaim: Provisionamento de nós Just-in-time Capacity
- Disruption: Consolidação do cluster (WhenEmpty, WhenEmptyOrUnderutilized, Budgets)
- Também tem suporte para instâncias spot.

Um pouco de mão na massa.

https://github.com/gmtborges/karpenter-demo

Obrigado

- @gmtborges
- linkedin.com/in/gmtborges

Referências

- Karpenter
- EKS Workshop
- Karpenter Github
- Karpenter on AWS
- Karpenter on Azure
- Karpenter on OKE