Introduction to Virtual Kubelet
Ria Bhatia, Product Manager, Microsoft
Sravan Rengarajan, Product Manager, Microsoft
Who are we

Ria Bhatia
Product Manager, Microsoft Azure
@rbitia
Maintainer of Virtual Kubelet Core
Who are we

Sravan Rengarajan

Product Manager, Microsoft Azure Container Compute
Provider Maintainer, Virtual Kubelet
Azure Kubernetes Service
Azure Container Instances
• Open Sourced by Microsoft in 2017
• Extend the Kubernetes API to serverless container platforms like ACI, AWS Fargate, Alibaba Cloud ECI.
Virtual Kubelet

Virtual Kubelet is an open source Kubernetes kubelet implementation that mirrors Kubernetes as a kubelet.

@VirtualKubelet

Virtual Kubelet on Twitter: “We’re so excited to finally share this with everyone – we’re now a CNCF project 😊”

Virtual Kubelet

Virtual Kubelet is an open source Kubernetes kubelet implementation that mirrors Kubernetes as a kubelet.

@VirtualKubelet

S/18 AM - 4 Dec 2018

29 Retweets 51 Likes

Nishant rhymes with Steve @nshantanil_pds - 4 Dec 2018

Replying to @virtualkubelet

I am very excited to be learning all about Virtual Kubelet at the Microsoft Open workshop next week (before KubeCon)
Announcing v1.0!
Kubernetes Architecture

In two slides.....
A Long Time Ago.....
Typical kubelets implement the pod and container operations for each node as usual.

Virtual kubelet registers itself as a “node” and allows developers to deploy pods and containers with their own APIs.

- Capacity
- OperatingSystem
- CreatePod
- UpdatePod
- NodeConditions
- GetPods
- GetPod
- GetPodStatus
• Kubernetes is fundamentally a node-based scheduling system
• A node comes with a number of conventions and constraints
  • Pods are run in the same address space
  • Pods are executed using low-level operating system primitives (cgroups / namespaces)
  • Pods share node resources - kernel, ephemeral disk, container images
  • Pods inherit certain aspects of node configuration
  • Node is a static size
  • Privileged containers expect to interact directly with node resources
  • A node is a single fault domain - affinity consideration
Anatomy of a Node: Node Agent

K8S API Server

Asynchronous
- Poll for pod changes
- Poll for secret changes
- Poll for volume changes
- Poll for service changes
- Push events (pod, ctr, node)
- Get CIDR range

Synchronous
- Get node and pod status
- Container interaction (exec)
- Get container logs
- Configure port forwarding

Pod create, update, delete
Volume create, delete
Secret create, update
DNS / routes config
Create Events

Internal
- GC loops (pod, ctr, image)
- Pod eviction QoS/OOM
- Run init containers
- Poll for stats
- Liveness / readiness monitors

Container attach, exec
Container start
Get container logs
Image pull, delete
...
• VK treats the concept of pods and nodes *in the abstract*
  • A node is a bounded amount of resource
  • A pod is a set of containers running within a sandbox
• The specifics of how these are manifested is contained within a pluggable Provider
• Common code in VK is relatively small
  • Registers with the API server as a client
  • Polls for state changes in Objects
  • Reports back state of "node" and "pods"
  • Handles synchronous interaction with containers
• VK can itself be deployed as a Pod
Why Virtual Kubelet?

• Flexibility of abstraction
• Flexibility of resource consumption
  • Greater granularity – pay for pods, not nodes
• Hybrid use-case - traditional cluster with a VK node for bursting / batch / on-demand pricing
• Single cluster high availability - a VK in each availability zone
• IOT edge connector - a VK schedules to a set of IOT devices
• Alternate Kubelet – VK as a node agent in a different kind of node
Provider Interface

// Provider contains the methods required to implement a Virtual Kubelet provider

type Provider interface {

    // Takes a Kubernetes Pod and deploys it within the provider
    CreatePod(ctx context.Context, pod *v1.Pod) error

    // Takes a Kubernetes Pod and updates it within the provider
    UpdatePod(ctx context.Context, pod *v1.Pod) error

    // Takes a Kubernetes Pod and deletes it from the provider
    DeletePod(ctx context.Context, pod *v1.Pod) error

    // Retrieves a pod by name from the provider (can be cached)
    GetPod(ctx context.Context, namespace, name string) (*v1.Pod, error)

    // Retrieves the logs of a container by name from the provider
    GetContainerLogs(ctx context.Context, namespace, podName, containerName string, tail int) (string, error)

    ....
Provider Definition

• Provide the back-end plumbing necessary to support the lifecycle management of pods, containers and supporting resources in the context of Kubernetes.

• Conform to the current API provided by Virtual Kubelet.

• Not have access to the Kubernetes API Server and has a well-defined callback mechanism for getting data like secrets or configmaps.
Extend Kubernetes workloads to any service you like!
## Providers in Virtual Kubelet

<table>
<thead>
<tr>
<th>Provider</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>alibabacloud</td>
<td>Decouple <code>vkubelet/*</code> packages from providers (#626)</td>
<td>3 days ago</td>
</tr>
<tr>
<td>aws</td>
<td>Decouple <code>vkubelet/*</code> packages from providers (#626)</td>
<td>3 days ago</td>
</tr>
<tr>
<td>azure</td>
<td>Decouple <code>vkubelet/*</code> packages from providers (#626)</td>
<td>3 days ago</td>
</tr>
<tr>
<td>azurebatch</td>
<td>Decouple <code>vkubelet/*</code> packages from providers (#626)</td>
<td>3 days ago</td>
</tr>
<tr>
<td>cri</td>
<td>Decouple <code>vkubelet/*</code> packages from providers (#626)</td>
<td>3 days ago</td>
</tr>
<tr>
<td>huawei</td>
<td>Upgrade to Go 1.12 (#630)</td>
<td>4 hours ago</td>
</tr>
<tr>
<td>mock</td>
<td>Decouple <code>vkubelet/*</code> packages from providers (#626)</td>
<td>3 days ago</td>
</tr>
<tr>
<td>nomad</td>
<td>Decouple <code>vkubelet/*</code> packages from providers (#626)</td>
<td>3 days ago</td>
</tr>
<tr>
<td>openstack</td>
<td>Decouple <code>vkubelet/*</code> packages from providers (#626)</td>
<td>3 days ago</td>
</tr>
<tr>
<td>register</td>
<td>Remove VIC provider code.</td>
<td>14 days ago</td>
</tr>
<tr>
<td>web</td>
<td>Decouple <code>vkubelet/*</code> packages from providers (#626)</td>
<td>3 days ago</td>
</tr>
</tbody>
</table>
IoT Edge with Virtual Kubelet
Virtual Node architecture in AKS
Tolerance to Pod Spec

```yaml
spec:
  tolerations:
  - key: virtual-kubelet.io/provider
    operator: Exists
  - effect: NoSchedule
    key: azure.com/aci
```
spec:
  affinity:
    nodeAffinity:
      preferredDuringSchedulingIgnoredDuringExecution:
        - preference:
            matchExpressions:
              - key: type
                operator: NotIn
                values:
                  - virtual-kubelet
Demo – AKS and ACI
Video goes here
Extend Kubernetes to Alibaba

- Viking: Pod lifecycle, networking, service discovery
  - kubelet + kube-proxy + kube-dns
  - Support Service & Ingress

As cloud provider, we offer public cloud product: **Serverless Kubernetes**
Thank you!!
KubeCon | CloudNativeCon
OPEN SOURCE SUMMIT
China 2019