

Kube-spawn

A tool to manage Kubernetes clusters on Linux hosts

Dongsu Park

Kinvolk

All Systems Go! 2017

Kube-spawn

- A tool to manage multi-node K8s clusters on Linux hosts
- Allows us to create Kubernetes clusters in an efficient way
- Based on open source projects
 - kubeadm
 - systemd-nspawn
 - CoreOS Container Linux

Requirements

- Systemd v233 or newer
 - e.g. Fedora 26 or Ubuntu 17.10
- Kubernetes v1.7.x or newer
- qemu-img, btrfs-progs
- libselinux-utils

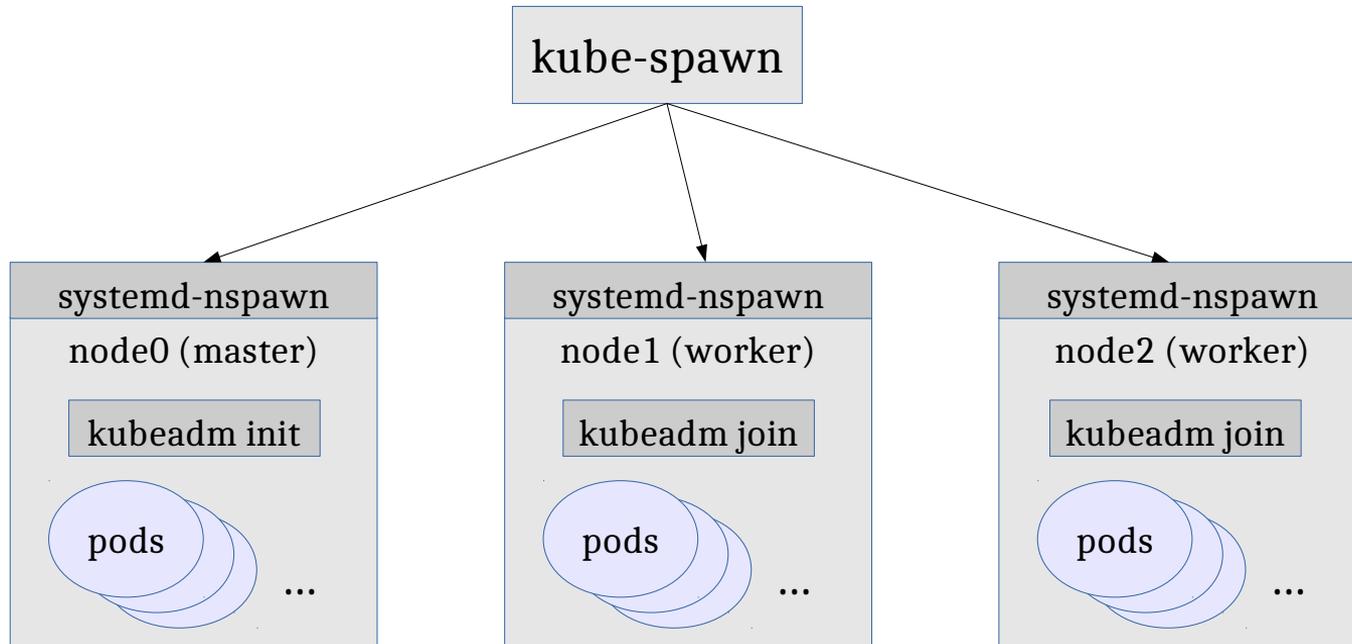
Features

- Makes use of systemd
 - systemd-nspawn, systemd-machined
- Supports multiple container runtimes
 - docker
 - rkt / rktlet
 - cri-o (planned)

Related projects

- minikube
 - launches a virtual machine
 - works on a single node
- kops
 - works on cloud providers

Overview



Demo

Command line interface

- Create
 - Generates the environment for a cluster to store locally under `/var/lib/kube-spawn`
 - Per-cluster, per-node
 - Downloads mandatory files, checks for requirements
- Start
 - Starts the nodes of a generated cluster.
 - “kubeadm init” on a master node
 - “kubeadm join” on worker nodes

Command line interface

- Stop
 - Stop nodes
 - “machinectl poweroff” && “machinectl remove”
- Destroy
 - Destroy created profiles as well as nodes under /var/lib/kube-spawn
 - Delete every persistent info about clusters

Cluster definition format

```
cluster-name = "default"  
container-runtime = "docker"  
image = "coreos"  
kubernetes-version = "v1.7.5"  
nodes = 2  
token = "f080cd.47e9e26e768b2f09"
```

```
[bindmount]
```

```
[[bindmount.read-only]]
```

```
dst = "/opt/cni/bin"
```

```
src = "/home/dpark/go/bin"
```

```
[runtime-config]
```

```
cgroup-per-qos = false
```

```
fail-swap-on = false
```

```
timeout = "15m"
```

```
use-legacy-cgroup-driver = true
```

```
[[machines]]
```

```
ip = "10.22.1.171"
```

```
name = "kubespawn0"
```

```
running = true
```

```
[machines.bindmount]
```

```
[[machines.bindmount.read-write]]
```

```
dst = "/var/lib/docker"
```

```
src = "/var/lib/kube-spawn/default/kubespawn0/mount"
```

(/var/lib/kube-spawn/default/kspawn.toml)

Integration issues

- Changes of Kubernetes interface
 - “--fail-swap-on=false” of kubelet
 - “--kubernetes-version” vs “kubernetesVersion” in kubeadm.yml
- Systemd-machined
 - Makes use of btrfs for storage pool under /var/lib/machines
 - Btrfs filesystem becomes full during extracting OS images
 - Workaround: on each node creation, enlarge storage pool

Integration issues

- Storage full inside nspawn containers
 - Each container is based on Container Linux: not much free space
 - As container runtime deploys apps, /var/lib becomes full
 - Workaround: mount a host directory into /var/lib/{docker,rktlet} inside containers
- Missing socat in Container Linux
 - Kubelet port-forward requires socat, which is missing in Container Linux
 - Workaround: download a static binary to be inserted into containers

Links

- Github: <https://github.com/kinvolk/kube-spawn>
 - Release v0.1.1: <https://github.com/kinvolk/kube-spawn/releases/tag/v0.1.1>
 - Planned release v0.2: <https://github.com/kinvolk/kube-spawn/tree/robertgxr/v02-refactor>
- Blog: <https://kinvolk.io/blog/2017/08/introducing-kube-spawn-a-tool-to-create-local-multi-node-kubernetes-clusters/>

Thanks!