

Kubemark Pre-existing Provider Guide

Kubemark Master - A set of Kubernetes control plane components running in a VM

Kubernetes Cluster - A real Kubernetes Cluster that has master and nodes. The hollow-node pods are run in this cluster, but appear as nodes to the Kubemark Master

Introduction

Every running Kubemark setup looks like the following: 1) A running Kubernetes cluster pointed to by the local kubeconfig 2) A separate VM where the kubemark master is running 3) Some hollow-nodes that run on the Kubernetes Cluster from #1 4) The hollow-nodes are configured to talk with the kubemark master at #2

When using the pre-existing provider, the developer is responsible for creating #1 and #2. Therefore, the kubemark scripts will not create any infrastructure or start a kubemark master like in other providers. Instead, the existing resources provided by the VM at \$MASTER_IP will serve as the kubemark master.

Use Case

The goal of the pre-existing provider is to use the kubemark tools with an existing kubemark master. It's meant to provide the developer with additional flexibility to customize the cluster infrastructure and still use the kubemark setup tools. The pre-existing provider is an **advanced** use case that requires the developer to have knowledge of setting up a kubemark master.

Requirements

To use the pre-existing provider, the expectation is that there's a kubemark master that is reachable at \$MASTER_IP. The machine that the kubemark master is on has to be ssh able from the host that's executing the kubemark scripts. And the user on that machine has to be 'kubernetes'.

Requirement checklist: - Set MASTER_IP to ip address to the kubemark master - The host where you execute the kubemark scripts must be able to ssh to kubernetes@\$MASTER_IP

Example Configuration

test/kubemark/cloud-provider-config.sh

```
CLOUD_PROVIDER="pre-existing"
KUBEMARK_IMAGE_MAKE_TARGET="push"
CONTAINER_REGISTRY=docker.io
```

```
PROJECT="rthallisey"  
MASTER_IP="192.168.121.29:6443"
```