Introducing Metal³

Kubernetes Native Bare Metal Host Management

Russell Bryant
Distinguished Engineer

Doug Hellmann
Sr. Principal Software
Engineer

Why another provisioning tool?

Kubernetes Native

Manage your underlying infrastructure through Kubernetes APIs using the Metal[3] and machine-api CRDs.

Self-Hosted

Metal[3] runs inside your cluster, avoiding the need for an external dependency.

Self-Managed

Integration with the Machine API from SIG Cluster-Lifecycle allows your bare metal cluster to manage its own growth, just like your cloud-based clusters.

Metal^3 Components

Cluster

Metal^3

BareMetalHost

Secret
(Config Drive)

baremetal-operator

Secret
(Credentials)

Ironic

OS Image
(httpd)

Host

BMC

Deploy
Ramdisk

Local Disk
(OS Image)

Config
Drive
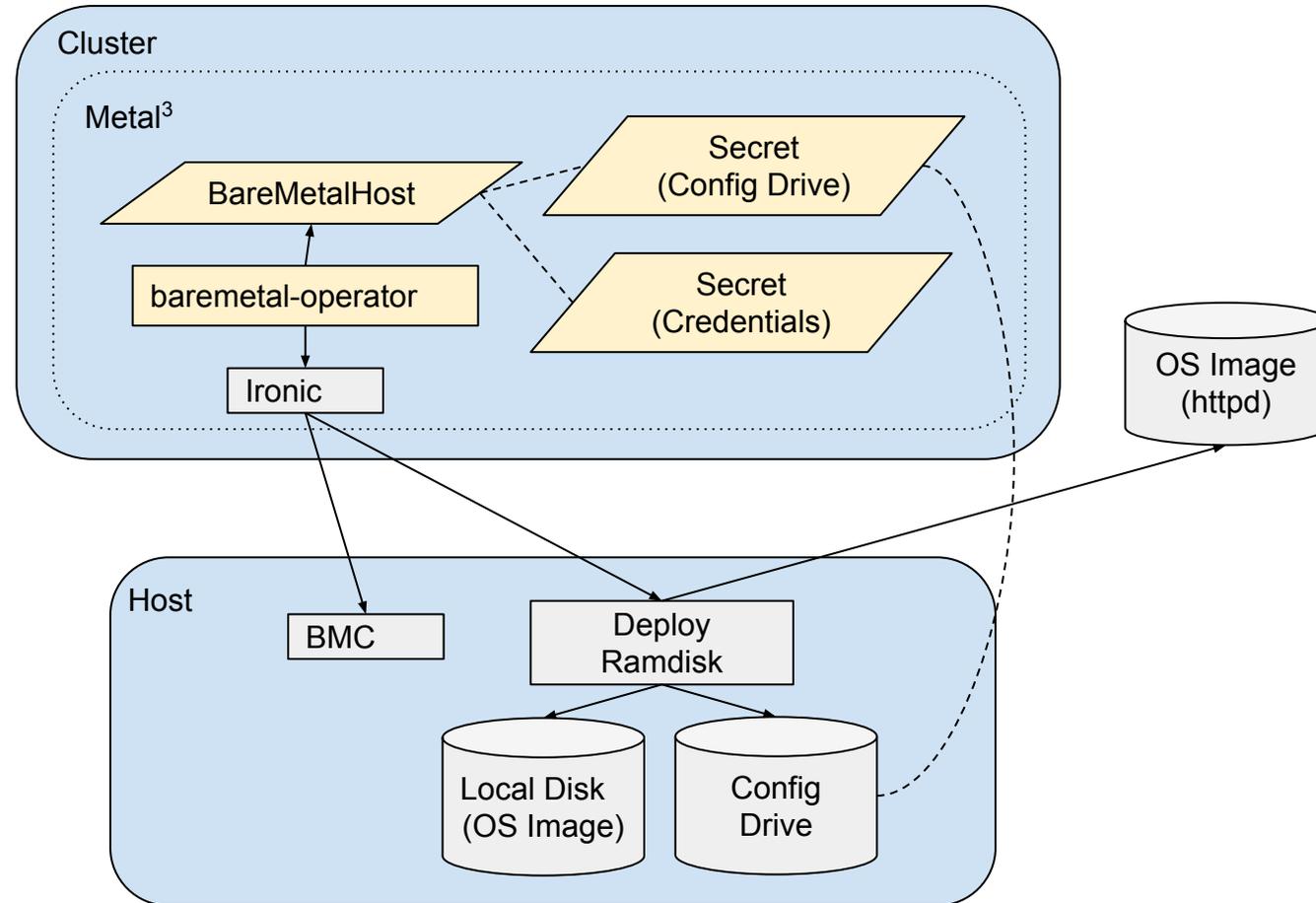
Metal^3 Integration with Machine API
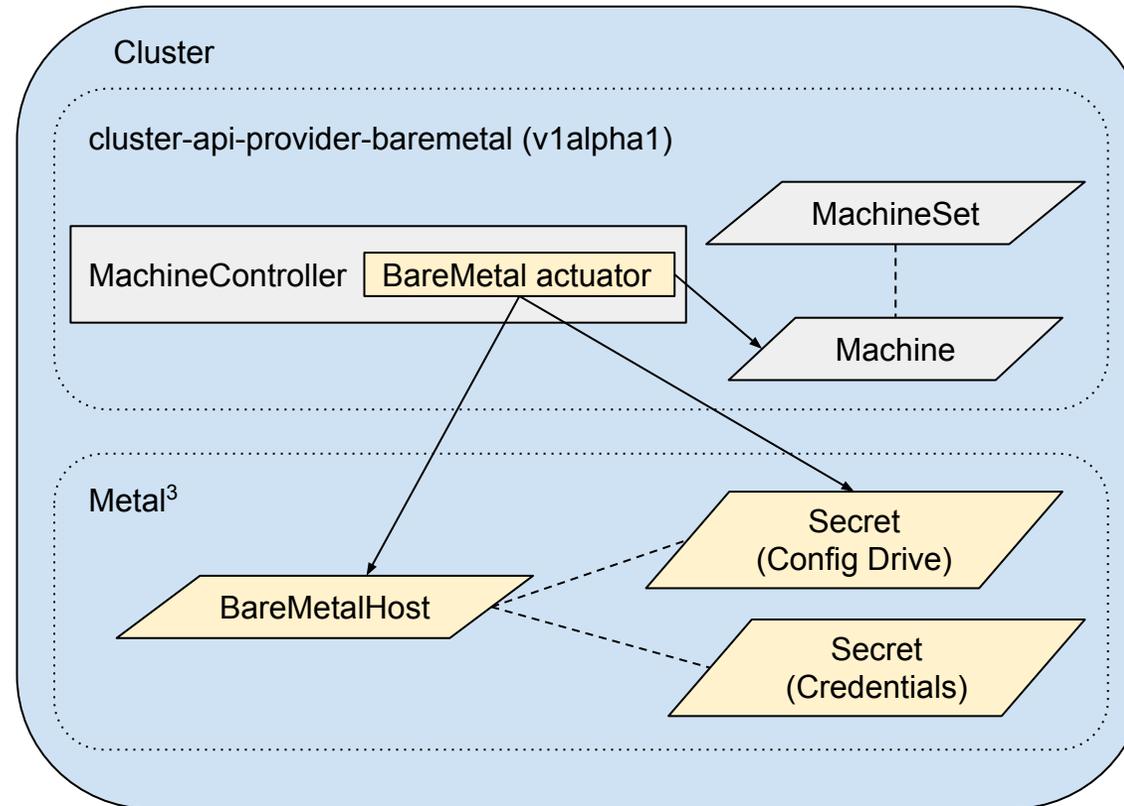
Why 3 Representations of Cluster Members?

Node

Represents a running instance of kubelet

Defined by Kubernetes Core

Managed via Kubernetes

Machine

Represents a request for a place to run kubelet

Defined by cluster-api

Managed via a driver called an actuator, running within a controller

BareMetalHost

Represents a physical computer, via profile and inventory data

Defined by Metal[3]

Managed via controller in the baremetal-operator

Worker Deployment

Set up Hosts	Manage Inventory	Create Machines	Provision	Operations
Connect hosts to the control plane and provisioning networks	Collect IPMI credentials and MAC addresses for hosts on the provisioning network.	Create Machine CRs (either individually or by scaling a MachineSet)	The Baremetal Operator uses Ironic hosted on the cluster to install an image on the worker hosts.	Nodes join the cluster. Ironic monitors power state and metrics.
	Create BareMetalHost CRs in the cluster that reflect available nodes		Workers boot and receive configuration data from the existing masters.	

BareMetalHost API

```yaml
kind: BareMetalHost
metadata:
  name: bmo-host-0
spec:
  bmc:
    address: ipmi://10.10.57.19
    credentialsName: bmo-host-0-bmc-secret
  bootMACAddress: 98:03:9b:61:80:48
  consumerRef:
    apiVersion: machine.openshift.io/v1beta1
    kind: Machine
    name: bmo-machine-0
    namespace: bmo-project
  image:
    checksum: http://172.16.1.100/images/myOSv1/myOS.qcow2.md5sum
    url: http://172.16.1.100/images/myOSv1/myOS.qcow2
  online: true
  userData:
    name: bmo-host-user-data
    namespace: bmo-project
```

BareMetalHost API

```
status:
  hardware:
    cpu:
      arch: x86_64
      clockMegahertz: 2000
      count: 40
      model: Intel(R) Xeon(R) Gold 6138 CPU @ 2.00GHz
    nics:
    - ip: 172.22.135.105
      mac: "98:03:9b:61:80:58"
      name: eno1
      pxe: true
      speedGbps: 25
      vlanId: 0
    ramMebibytes: 131740
  operationalStatus: OK
  provisioning:
    ID: a4438010-3fc6-4c5c-b570-900bbe85da57
    image:
      checksum: http://172.16.1.100/images/myOSv1/myOS.qcow2.md5sum
      url: http://172.16.1.100/images/myOSv1/myOS.qcow2
    state: provisioned
```

Demo

See https://metal3.io/blog/index.html for demo resources.

Future Work

BIOS
Manage BIOS settings during deployment.

RAID
Create RAID volumes during deployment.

cluster-api v1alpha2/3
v1alpha2 support is in development now.

Thank you

We are looking for contributors interested in

collaborating on the design and implementation of

the next generation of Kubernetes-native

infrastructure management.

Existing contributors from:

- Red Hat, Ericsson, AT&T, Fujitsu, Mirantis

Infrastructure provided by:

- Packet.net, Nordix, Netlify, travis-ci

https://metal3.io

metal3-io

@metal3_io

Red Hat