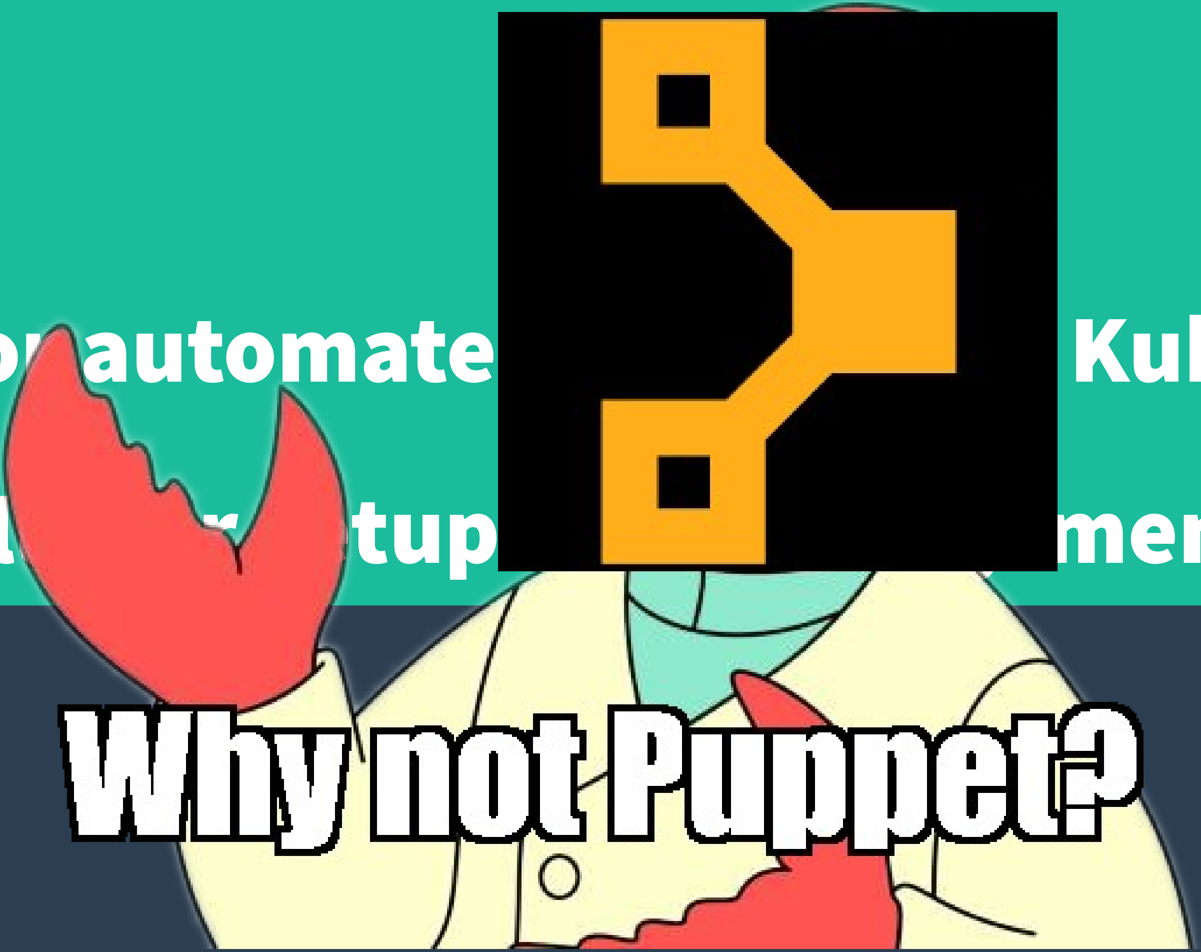
Tools for automated bare-metal Kubernetes cluster setup and management?

Alexander Olofsson

Tools fo automate Kubernetes cl r tup ment?



Who am I

LINKÖPING
UNIVERSITY

alexander.olofsson@liu.se
@ace:kittenface.studio
#ananace on Freenode
@ananace on Telegram

github.com/ananace
(github.com/ace13)

Who am I

LINKÖPING
UNIVERSITY

alexander.olofsson@liu.se
@ace:kittenface.studio
#ananace on Freenode
@ananace on Telegram

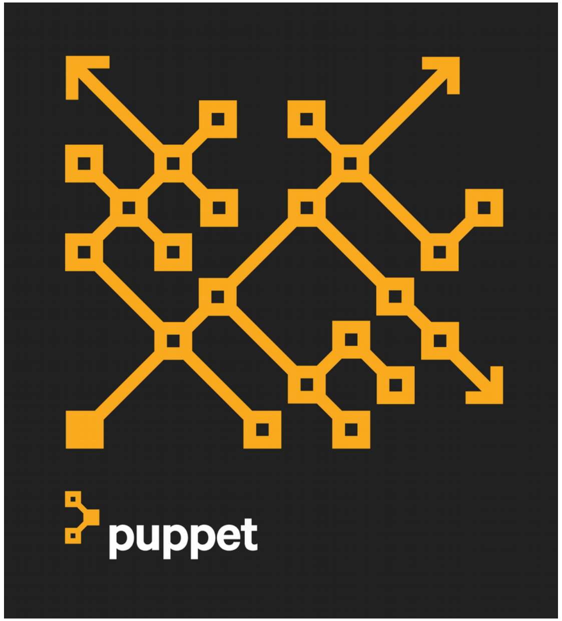
github.com/ananace
(github.com/ace13)

Disclaimer

There are plenty of tools for manually deploying Kubernetes today.

At least for virtualized / Cloud environments.

What is “puppet”?



What is “ puppet”?

```
class profiles::kube::node(
  String $kube_cluster                                 = 'default',
  Boolean $allow_privileged                            = false,
  Optional[Variant[Array[String],String]] $api_servers = undef,
  Optional[Hash[String,String]] $node_labels           = undef,
) {
  $kube_master = query_resources("Class['profiles::kube::master']{ kube_cluster = '${kube_cluster}' }", "Class['profiles::kube::master']")[0]
  $kube_masters_apiserv = query_resources("Class['profiles::kube::master']{ kube_cluster = '${kube_cluster}' }",
"Class['hyperkube::control_plane::apiserver']")
  $master_flannel = query_resources("Class['profiles::kube::master']{ kube_cluster = '${kube_cluster}' }", "Class['flannel']")[0]
  $master_flannel_backend = query_resources("Class['profiles::kube::master']{ kube_cluster = '${kube_cluster}' }", "Class['flannel::etcd_key']")[0]

  $etcd_nodes = sort(query_resources("Class['profiles::etcd']{ cluster = '${kube_master[parameters][etcd_cluster]}' }", "Class['etcd']").map |$n| {
      if $n[certname] == $trusted[certname] {
        split($n[parameters][listen_client_urls], ',')[1]
      } else {
        $n[parameters][advertise_client_urls]
      }
  })

  $node_cert = '/etc/etcd/cert.pem'
  $node_key = '/etc/etcd/key.pem'
  $node_ca = '/etc/etcd/ca.pem'

  class { '::profiles::etcd::client':
    cluster => $kube_master[parameters][etcd_cluster],
  }
  if !defined(Class['::flannel']) {
    class { '::flannel':
      etcd_endpoints  => $etcd_nodes,
      ...
```

What does puppet give us?

- Configuration Management

  Reuse of configuration modules through the Puppet Forge

- System Inventory w/ PuppetDB
- Automated tasks w/ Bolt
- (Orchestration w/ MCollective)

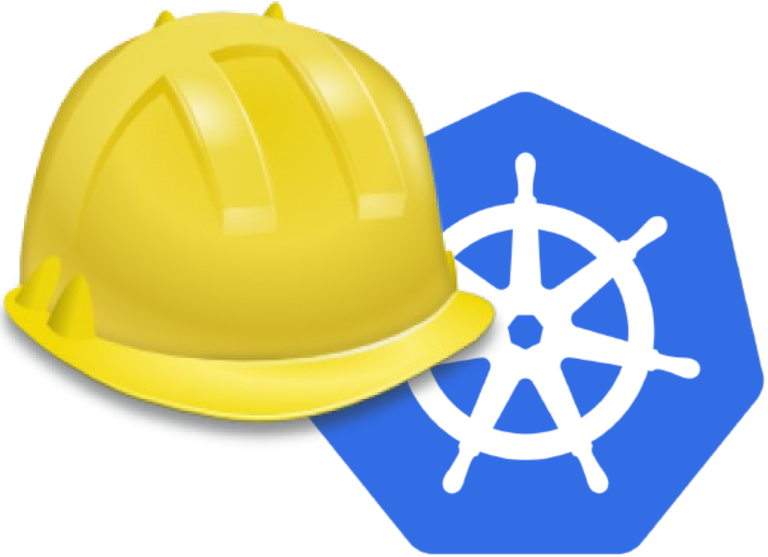
Which parts do we actually need?

- Puppetmaster
- PuppetDB
- One or more nodes that are registered with Puppet

  (Or can be provisioned with Puppet installed)

NB: The hyperkube module will work without a master/DB

Demo



What is happening now?

What is happening now?

MAGIC

What is happening now?

- The Foreman lifecycle-management tool, being used as a Puppet ENC
- Bare-metal nodes being configured as K8s Kubelets

  (Provisioning of Bare-Metal nodes)
- Magic behind the scenes

The ‘Magic behind the scenes’

- Puppet `hyperkube` module
- Etcd, flannel, firewalld
- Smoke and mirrors

Demo nr. 2

Current state

- Module can be used for production
  - Not on Forge yet, sorry
- Full automation is possible with PuppetDB
- Currently cheats with K8s Addon Manager
  - There’s a module for orchestrating w/ Puppet
- Lacking a few good-to-have addons

Questions?

alexander.olofsson@liu.se
@ace:kittenface.studio
#ananace on Freenode
@ananace on Telegram

github.com/ananace

github.com/ananace/hyperkube