OpenStack Magnum: Hands-on

Get Presentation Tarball

- Compiled slides (recommended)
  - *Internal URL omitted*
  - *Internal URL omitted*
- odpdown sources only (not merged, yet)
  - *Internal URL omitted*
- Commands and configuration files for hands-on are in the `cmd/` directory

Pre-checks for deploying a cluster using magnum

- On the Controller node

```
source .openrc

magnum service-list

openstack image show openstack-magnum-k8s-image
```

- Package for the magnum image

```
zypper in openstack-magnum-k8s-image-x86_64
```

- Create the glance image if required
- [cmd/create-glance-image.sh]

```
glance image-create --name openstack-magnum-k8s-image \
                    --visibility public \
                    --disk-format qcow2 \
                    --os-distro opensuse \
                    --container-format bare\
                    --file /srv/tftpboot/files/openstack-magnum-k8s-
image/openstack-magnum-k8s-image.x86_64.qcow2
```

Pre-checks for deploying a cluster using magnum (cont.)

```
openstack keypair show default
openstack flavor show m1.magnum
openstack network show floating
```

- [cmd/create-keypair.sh]

```
openstack keypair create --public-key
~/.ssh/id_rsa.pub default
```

- [cmd/create-flavor.sh]

```
openstack flavor create --ram 1024 --disk 10
--vcpus 1 m1.magnum
```

Commands we will use frequently

```
magnum cluster-template-list
magnum cluster-template-show
k8s_template
magnum cluster-list
magnum cluster-show k8s_cluster
```

Steps to deploy a Kubernetes cluster using magnum CLI

- Create the cluster template
- [cmd/create-cluster-template.sh]

```
magnum cluster-template-create \
                          \
  --name k8s_template \
  --image-id openstack-magnum-k8s-image \
  --keypair-id default \
  --external-network-id floating \
  --dns-nameserver 8.8.8.8 \
  --flavor-id m1.magnum \
  --master-flavor-id m1.magnum \
  --docker-volume-size 5 \
  --network-driver flannel \
  --coe kubernetes \
  --tls-disabled
```

Interlude: `cluster-template-show` Output

```
# magnum cluster-template-show k8s_template
+-----------------------+--------------------------------------+
| Property              | Value                                |
+-----------------------+--------------------------------------+
| insecure_registry     | -                                    |
| labels                | {}                                   |
| updated_at            | -                                    |
| floating_ip_enabled   | True                                 |
| fixed_subnet          | -                                    |
| master_flavor_id      | m1.magnum                            |
| uuid                  | 0f1fb1bf-7e34-4c05-8a66-062392817c5f |
| no_proxy              | -                                    |
| https_proxy           | -                                    |
| tls_disabled          | True                                 |
| keypair_id            | default                              |
| public                | False                                |
| http_proxy            | -                                    |
| docker_volume_size    | 5                                    |
| server_type           | vm                                   |
| external_network_id   | floating                             |
| cluster_distro        | opensuse                             |
| image_id              | openstack-magnum-k8s-image           |
| volume_driver         | -                                    |
| registry_enabled      | False                                |
| docker_storage_driver | devicemapper                         |
| apiserver_port        | -                                    |
| name                  | k8s_template                         |
| created_at            | 2017-05-17T13:53:10+00:00            |
| network_driver        | flannel                              |
| fixed_network         | -                                    |
| coe                   | kubernetes                           |
| flavor_id             | m1.magnum                            |
| master_lb_enabled     | False                                |
| dns_nameserver        | 8.8.8.8                              |
+-----------------------+--------------------------------------+
```

Steps to deploy a Kubernetes cluster using magnum CLI(cont.)

- Create the cluster
- [cmd/create-cluster.sh]

```
magnum cluster-create \
  --name k8s_cluster \
  --cluster-template k8s_template \
  --node-count 1
```

Interlude: `cluster-show` Output

```
magnum cluster-show k8s_cluster
 +---------------------+------------------------------------------------------------------+
 | Property            | Value                                                            |
 +---------------------+------------------------------------------------------------------+
 | status              | CREATE_COMPLETE                                                  |
 | cluster_template_id | 0f1fb1bf-7e34-4c05-8a66-062392817c5f                             |
 | uuid                | 3ffe8404-a801-4c3f-9a7b-16a297f5351a                             |
 | stack_id            | a4fcbd98-b7d0-4f74-8d88-51bbd5d229d4                             |
 | status_reason       | Stack CREATE completed successfully                              |
 | created_at          | 2017-05-17T14:41:29+00:00                                        |
 | name                | k8s_cluster                                                      |
 | updated_at          | 2017-05-17T15:05:30+00:00                                        |
 | discovery_url       | https://discovery.etcd.io/8a5dc783ca8c17109a386d6d690d2f2f       |
 | api_address         | http://192.168.226.138:8080                                      |
 | coe_version         | v1.3.7                                                           |
 | master_addresses    | ['192.168.226.138']                                              |
 | create_timeout      | 60                                                               |
 | node_addresses      | ['10.0.0.4']                                                     |
 | master_count        | 1                                                                |
 | container_version   | 1.12.3                                                           |
 | node_count          | 1                                                                |
 +---------------------+------------------------------------------------------------------+
```

Monitoring and updating the kubernetes cluster

- List Heat stacks

```
openstack stack list
```

- Get cluster's stack ID:
- [cmd/stack-uuid.sh]

```
stack=$(magnum cluster-show k8s_cluster \
        | grep -w stack_id \
        | awk '{print $4}')
export stack
```

- Monitor stack status

```
watch openstack stack resource list -n 5 $stack \| grep -v
CREATE_COMPLETE
```

- Update the cluster
- [cmd/update-cluster.sh]

```
magnum cluster-update k8s_cluster replace node_count=2
```

Accessing the Kubernetes cluster using magnum CLI

- Package needed: openstack/python-magnumclient
- Generate and export the kubernetes cluster config to use it from anywhere

```
magnum cluster-config --dir $HOME k8s_cluster

export KUBECONFIG=$HOME/config
```

- Install kubectl (https://kubernetes.io/docs/tasks/tools/install-kubectl/)
- [cmd/install-kubectl.sh]

```
curl -LO https://storage.googleapis.com/kubernetes-release/release/$
(curl -s https://storage.googleapis.com/kubernetes-
release/release/stable.txt)/bin/linux/amd64/kubectl

chmod +x ./kubectl

sudo mv ./kubectl /usr/local/bin/kubectl
```

Kubernetes commands we will frequently use

NOTE: Can be run directly on the master node or from anywhere by exporting the config like above

```
kubectl version

kubectl cluster-info dump

kubectl explain <element>

kubectl get nodes

kubectl get pods

kubectl get svc
```

Running our first pod

- Create pod: `kubectl create -f nginx.yaml`
- Create a file (e.g nginx.yaml) describing a pod running nginx:
- [cmd/nginx.yaml]

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
   app: nginx
spec:
  containers:
  - name: nginx
    image: nginx
    ports:
    - containerPort: 80
```

Running our first service

- Create service: `kubectl create -f nginx-service.yaml`
- Create a file (e.g nginx-service.yaml) describing a service for the nginx pod:
- [cmd/nginx-service.yaml]

```
apiVersion: v1
kind: Service
metadata:
  name: nginxservice
  labels:
    app: nginx
spec:
  ports:
  - port: 80
    targetPort: 80
    protocol: TCP
  selector:
    app: nginx
```

Work in progress

Making the service available to the internet

- Enabling external loadbalancer feature
  - Upstream Patch
- Login to master node and modify /etc/sysconfig/kubernetes_openstack_config to add username and password
- Restart service

```
systemctl restart kube-controller-manager
```

Running modified service

- Create service: `kubectl create -f nginx-service1.yaml`
- Create a file (e.g nginx-service1.yaml) describing a service for the nginx pod:
- [cmd/nginx-service1.yaml]

```
apiVersion: v1
kind: Service
metadata:
  name: nginxservice1
  labels:
    app: nginx
spec:
  ports:
  - port: 80
    targetPort: 80
    protocol: TCP
  selector:
    app: nginx
  type: LoadBalancer
```

Demo: Deploying kubernetes cluster from horizon

Troubleshooting

Preparation: Some Sabotage

- No Heat API access for floating IPs
- [cmd/iptables-break.sh]

```
iptables -A INPUT \
         -s $(ip addr sh br-public | grep -w inet \
              | awk '{print $2}' | sed 's#/.*##') \
         -p tcp --dport 8004 -j ACCEPT

iptables -A INPUT \
         -s $(neutron net-list | grep floating \
              | awk '{print $7}') \
         -p tcp --dport 8004 \
         -j REJECT --reject-with tcp-reset
```

Preparation: Create a Fresh Cluster

- Delete existing cluster
- [cmd/delete-cluster.sh]

```
magnum cluster-delete k8s_cluster
```

- Create a new cluster
- [cmd/create-cluster.sh]

```
magnum cluster-create \
  --name k8s_cluster \
  --cluster-template k8s_template \
  --node-count 1
```

When is a Cluster Complete?

- Signalling mechanism: Heat WaitCondition
  - One for each instance
  - Instance curls a magic Heat API URL after `cloud-init` completes
- Possible failure modes:
  - Heat API unreachable
  - cloud-init fails somewhere along the way
  - Cluster takes longer than WaitCondition timeout to complete

1: Heat API Unreachable / Timeout

- Step 1: obtain cluster's main Heat stack UUID
- [cmd/stack-uuid.sh]

```
stack=$(magnum cluster-show k8s_cluster \
        | grep -w stack_id \
        | awk '{print $4}')
export stack
```

- Step 2: find failing WaitCondition's nested stack name
- [cmd/nested-name.sh]

```
nested=$(openstack stack resource list -n 5 $stack \
                | grep CREATE_FAILED \
                | grep OS::Heat::WaitCondition \
                | awk '{print $11}')
export nested
```

Interlude: `cluster-show` Output

```
root@d52-54-77-77-01-01:~ # magnum cluster-show k8s_cluster
+---------------------+--------------------------------------------------------------------------------------------------------------------------------------------+
| Property            | Value                                                                                                                                      |
+---------------------+--------------------------------------------------------------------------------------------------------------------------------------------+
| status              | CREATE_FAILED                                                                                                                              |
| cluster_template_id | ee9d182c-6987-46f6-8e6a-c873e2b10f6c                                                                                                       |
| uuid                | bab88e18-a0dc-4db0-9be2-c1a9bf23104a                                                                                                       |
| stack_id            | f59404cd-14c9-4b62-b62b-cb0beb057514                                                                                                       |
| status_reason       | Resource CREATE failed: WaitConditionTimeout: resources.kube_masters.resources[0].resources.master_wait_condition:                         |
|                     | 0 of 1 received                                                                                                                            |
| created_at          | 2017-05-16T08:54:39+00:00                                                                                                                  |
| name                | k8s_cluster                                                                                                                                |
| updated_at          | 2017-05-16T09:06:55+00:00                                                                                                                  |
| discovery_url       | https://discovery.etcd.io/e025ff3b532c3f5845824b36671faca0                                                                                 |
| faults              | {'0': 'WaitConditionTimeout: resources[0].resources.master_wait_condition: 0 of 1 received', 'kube_masters':                               |
|                     | 'WaitConditionTimeout: resources.kube_masters.resources[0].resources.master_wait_condition: 0 of 1 received',                              |
|                     | 'master_wait_condition': 'WaitConditionTimeout: resources.master_wait_condition: 0 of 1 received'}                                         |
| api_address         | http://:8080                                                                                                                               |
| coe_version         | v1.3.7                                                                                                                                     |
| master_addresses    | ['192.168.232.129']                                                                                                                        |
| create_timeout      | 60                                                                                                                                         |
| node_addresses      | []                                                                                                                                         |
| master_count        | 1                                                                                                                                          |
| container_version   | 1.12.3                                                                                                                                     |
| node_count          | 1                                                                                                                                          |
+---------------------+--------------------------------------------------------------------------------------------------------------------------------------------+
```

1: Heat API Unreachable / Timeout

- Step 1: obtain cluster's main Heat stack UUID
- [cmd/stack-uuid.sh]

```
stack=$(magnum cluster-show k8s_cluster \
        | grep -w stack_id \
        | awk '{print $4}')
export stack
```

- Step 2: find failing WaitCondition's nested stack name
- [cmd/nested-name.sh]

```
nested=$(openstack stack resource list -n 5 $stack \
                | grep CREATE_FAILED \
                | grep OS::Heat::WaitCondition \
                | awk '{print $11}')
export nested
```

Interlude: `resource list` Output

```
root@d52-54-77-77-01-01:~ # openstack stack resource list -n 5 -c resource_name -c resource_type -c resource_status -c stack_name f59404cd-14c9-4b62-b62b-cb0beb057514
+-----------------------------+-----------------------------------------------------------------------------------------------------+-----------------+-----------------------------------------------------------------------------------------------------------------------------+
| resource_name               | resource_type                                                                                       | resource_status | stack_name                                                                                                                  |
+-----------------------------+-----------------------------------------------------------------------------------------------------+-----------------+-----------------------------------------------------------------------------------------------------------------------------+
| api_monitor                 | Magnum::Optional::Neutron::LBaaS::HealthMonitor                                                     | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| extrouter_inside            | OS::Neutron::RouterInterface                                                                        | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| secgroup_kube_master        | OS::Neutron::SecurityGroup                                                                          | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| kube_master_ports           | OS::Heat::ResourceGroup                                                                             | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| secgroup_kube_minion        | OS::Neutron::SecurityGroup                                                                          | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| etcd_pool                   | Magnum::Optional::Neutron::LBaaS::Pool                                                              | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| api_pool                    | Magnum::Optional::Neutron::LBaaS::Pool                                                              | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| kube_minions                | OS::Heat::ResourceGroup                                                                             | INIT_COMPLETE   | k8s_cluster-jf3atei2jlpu                                                                                                    |
| api_address_floating_switch | Magnum::FloatingIPAddressSwitcher                                                                   | INIT_COMPLETE   | k8s_cluster-jf3atei2jlpu                                                                                                    |
| etcd_loadbalancer           | Magnum::Optional::Neutron::LBaaS::LoadBalancer                                                      | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| api_address_lb_switch       | Magnum::ApiGatewaySwitcher                                                                          | INIT_COMPLETE   | k8s_cluster-jf3atei2jlpu                                                                                                    |
| etcd_address_lb_switch      | Magnum::ApiGatewaySwitcher                                                                          | INIT_COMPLETE   | k8s_cluster-jf3atei2jlpu                                                                                                    |
| fixed_subnet                | OS::Neutron::Subnet                                                                                 | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| kube_masters                | OS::Heat::ResourceGroup                                                                             | CREATE_FAILED   | k8s_cluster-jf3atei2jlpu                                                                                                    |
| etcd_monitor                | Magnum::Optional::Neutron::LBaaS::HealthMonitor                                                     | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| api_pool_floating           | Magnum::Optional::Neutron::FloatingIP                                                               | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| extrouter                   | OS::Neutron::Router                                                                                 | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| fixed_network               | OS::Neutron::Net                                                                                    | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| secgroup_base               | OS::Neutron::SecurityGroup                                                                          | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| api_loadbalancer            | Magnum::Optional::Neutron::LBaaS::LoadBalancer                                                      | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| kube_minion_ports           | OS::Heat::ResourceGroup                                                                             | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| etcd_listener               | Magnum::Optional::Neutron::LBaaS::Listener                                                          | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| api_listener                | Magnum::Optional::Neutron::LBaaS::Listener                                                          | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu                                                                                                    |
| 0                           | file:///usr/lib/python2.7/site-packages/magnum/drivers/k8s_opensuse_v1/templates/kubeport.yaml      | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_master_ports-5p52lkj663n3                                                                     |
| kube_port                   | OS::Neutron::Port                                                                                   | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_master_ports-5p52lkj663n3-0-lajcxcfmpgth                                                      |
| 0                           | file:///usr/lib/python2.7/site-packages/magnum/drivers/k8s_opensuse_v1/templates/kubemaster.yaml    | CREATE_FAILED   | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m                                                                          |
| kube_master_floating        | Magnum::Optional::KubeMaster::Neutron::FloatingIP                                                   | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| api_pool_member             | Magnum::Optional::Neutron::LBaaS::PoolMember                                                        | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| write_heat_params           | OS::Heat::SoftwareConfig                                                                            | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| create_kubernetes_user      | OS::Heat::SoftwareConfig                                                                            | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| kube_master                 | OS::Nova::Server                                                                                    | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| kube_master_init            | OS::Heat::MultipartMime                                                                             | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| master_wait_handle          | OS::Heat::WaitConditionHandle                                                                       | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| etcd_pool_member            | Magnum::Optional::Neutron::LBaaS::PoolMember                                                        | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| master_wait_condition       | OS::Heat::WaitCondition                                                                             | CREATE_FAILED   | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| configure_kubernetes        | OS::Heat::SoftwareConfig                                                                            | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| master_wc_notify            | OS::Heat::SoftwareConfig                                                                            | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| configure_etcd              | OS::Heat::SoftwareConfig                                                                            | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| make_cert                   | OS::Heat::SoftwareConfig                                                                            | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| configure_flanneld          | OS::Heat::SoftwareConfig                                                                            | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_masters-gypg2jphpe3m-0-gqt6az75rjgs                                                           |
| 0                           | file:///usr/lib/python2.7/site-packages/magnum/drivers/k8s_opensuse_v1/templates/kubeport.yaml      | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_minion_ports-u2ny44qx4lko                                                                     |
| kube_port                   | OS::Neutron::Port                                                                                   | CREATE_COMPLETE | k8s_cluster-jf3atei2jlpu-kube_minion_ports-u2ny44qx4lko-0-znm2n4vv5bxq                                                      |
+-----------------------------+-----------------------------------------------------------------------------------------------------+-----------------+-----------------------------------------------------------------------------------------------------------------------------+
```

1: Heat API Unreachable / Timeout (cont.)

- Step 3: check for wait condition signalling in log
- [cmd/signal-log.sh]:

```
grep $nested /var/log/heat/heat-
api.log \
  | grep /signal
```

  - If this yields results but the WaitCondition has failed, deployment was successful, but the WaitCondition already transitioned to `CREATE_FAILED` because it took too long.
  - If this does not yield any results, we'll need to keep looking...

2: Failing user-data script

- Step 1: obtain failing cluster's main Heat stack UUID
- [cmd/stack-uuid.sh]

```
stack=$(magnum cluster-show k8s_cluster \
        | grep -w stack_id \
        | awk '{print $4}')
export stack
```

- Step 2: obtain nested stack name for failing WaitCondition
- [cmd/nested-name.sh]

```
nested=$(openstack stack resource list -n 5 $stack \
                | grep CREATE_FAILED \
                | grep OS::Heat::WaitCondition \
                | awk '{print $11}')
export nested
```

- Step 3: obtain floating IP address for instance with failed WaitCondition
- [cmd/show-floating.sh]:

```
openstack stack output show $nested kube_master_external_ip  # Only run this now
openstack stack output show $nested kube_minion_external_ip  # Run if minion
fails
```

2: Failing user-data script (cont.)

- Step 4: SSH login (as root) to floating IP address
- Step 5: examine `/var/log/cloud-init-output.log`

```
2017-05-12 09:31:09,517 -
util.py[WARNING]: Failed running
/var/lib/cloud/instance/scripts/part-007
```

- Step 6 (optional): locate source script under `/usr/lib/python2.7/magnum/`, find problem, add debug output, fix script...

Interlude: Fix iptables, Break Timeout...

- Fix iptables
- [cmd/iptables-unbreak.sh]

```
iptables -D INPUT \
         -s $(neutron net-list | grep floating \
              | awk '{print $7}') \
         -p tcp --dport 8004 \
         -j REJECT --reject-with tcp-reset
```

- Break WaitCondition timeout
- [cmd/break-timeout.sh]

```
sed -i s/600/60/ \
/usr/lib/python2.7/site-packages/magnum\
/drivers/k8s_opensuse_v1/templates/kubecluster.yaml
```

Interlude: ...and Rebuild Cluster

- Delete existing cluster
- [cmd/delete-cluster.sh]

```
magnum cluster-delete k8s_cluster
```

- Create new cluster
- [cmd/create-cluster.sh]

```
magnum cluster-create \
  --name k8s_cluster \
  --cluster-template k8s_template \
  --node-count 1
```

3: Genuine Timeout

- Step 1: obtain cluster's main Heat stack UUID
- [cmd/stack-uuid.sh]:

```
stack=$(magnum cluster-show k8s_cluster \
        | grep -w stack_id \
        | awk '{print $4}')
export stack
```

- Step 2: find failing WaitCondition's nested stack name
- [cmd/nested-name.sh]:

```
nested=$(openstack stack resource list -n 5 $stack \
                | grep CREATE_FAILED \
                | grep OS::Heat::WaitCondition \
                | awk '{print $11}')
export nested
```

- Step 3: check for wait condition signalling in log
- [cmd/signal-log.sh]:

```
grep $nested /var/log/heat/heat-api.log \
  | grep /signal
```

Interlude: Fix Timeout, Break etcd

- Restore old WaitCondition Timeout
- [cmd/fix-timeout.sh]

```
sed -i s/60/600/ \
/usr/lib/python2.7/site-packages/magnum\
/drivers/k8s_opensuse_v1/templates/kubecluster
.yaml
```

- Add a rogue DNS entry for `discovery.etcd.io`
- [cmd/break-etcd.sh]

```
echo '127.0.0.1    discovery.etcd.io' >>
/etc/hosts
```

Redeploy Cluster Once More

- Delete existing cluster
- [cmd/delete-cluster.sh]

```
magnum cluster-delete k8s_cluster
```

- Create a new cluster
- [cmd/create-cluster.sh]

```
magnum cluster-create \
  --name k8s_cluster \
  --cluster-template k8s_template \
  --node-count 1
```

Failure to Get Discovery URL

```
root@d52-54-77-77-01-01:~ # magnum cluster-show k8s_cluster
+---------------------+--------------------------------------------------------------------------------+
| Property            | Value                                                                          |
+---------------------+--------------------------------------------------------------------------------+
| status              | CREATE_FAILED                                                                  |
| cluster_template_id | ee9d182c-6987-46f6-8e6a-c873e2b10f6c                                           |
| uuid                | 05203920-0c2b-4741-871a-8f298431a769                                           |
| stack_id            | -                                                                              |
| status_reason       | Failed to get discovery url from 'https://discovery.etcd.io/new?size=1'.       |
| created_at          | 2017-05-17T15:04:01+00:00                                                      |
| name                | k8s_cluster                                                                    |
| updated_at          | -                                                                              |
| discovery_url       | -                                                                              |
| faults              | {}                                                                             |
| api_address         | -                                                                              |
| coe_version         | -                                                                              |
| master_addresses    | []                                                                             |
| create_timeout      | 60                                                                             |
| node_addresses      | []                                                                             |
| master_count        | 1                                                                              |
| container_version   | -                                                                              |
| node_count          | 1                                                                              |
+---------------------+--------------------------------------------------------------------------------+
```

Static etcd Cluster to the Rescue

- We have various Provisions for offline (Cloud without Internet access) operation.
  - Only in OpenSUSE Kubernetes driver
  - Includes `etcd` operation without discovery
  - Based on static list of cluster members
- Caveat: cluster update is effectively rebuild of *all* cluster nodes

Rebuild Cluster Without Discovery URL

- Delete existing cluster
- [cmd/delete-cluster.sh]

```
magnum cluster-delete k8s_cluster
```

- Create a new cluster
- [cmd/create-cluster-discovery.sh]

```
magnum cluster-create \
  --name k8s_cluster \
  --cluster-template k8s_template \
  --node-count 1
```

Rebuild Cluster Without Discovery URL

- Delete existing cluster
- [cmd/delete-cluster.sh]

```
magnum cluster-delete k8s_cluster
```

- Create a new cluster
- [cmd/create-cluster-discovery.sh]

```
magnum cluster-create \
  --name k8s_cluster \
  --cluster-template k8s_template \
  --node-count 1 \
  --discovery_url none
```

Undo etcd Sabotage

- Remove `/etc/hosts` entry
- [cmd/fix-etcd.sh]

```
sed -i /discovery.etcd.io/d /etc/hosts
```

- More `etcd` failure modes: https://docs.openstack.org/developer/magnum/troubleshooting-guide.html#etcd-service

Bonus Error: cluster_user_trust

- Patch for CVE-2016-7404 breaks clusters with `registry-enabled` set in ClusterTemplate
- Relevant for Magnum offline scenario (cloud without Internet access)

cluster_user_trust: Create Template

- Create ClusterTemplate with `--registry-enabled`
- [cmd/create-template-registry.sh]

```
magnum cluster-template-create \
                          \
  --name k8s_template \
  --image-id openstack-magnum-k8s-image \
  --keypair-id default \
  --external-network-id floating \
  --dns-nameserver 8.8.8.8 \
  --flavor-id m1.magnum \
  --master-flavor-id m1.magnum \
  --docker-volume-size 5 \
  --network-driver flannel \
  --coe kubernetes \
  --tls-disabled
```

cluster_user_trust: Create Template

- Create ClusterTemplate with `--registry-enabled`
- [cmd/create-template-registry.sh]

```
magnum cluster-template-create \
  --registry-enabled  \
  --name k8s_registry \
  --image-id openstack-magnum-k8s-image \
  --keypair-id default \
  --external-network-id floating \
  --dns-nameserver 8.8.8.8 \
  --flavor-id m1.magnum \
  --master-flavor-id m1.magnum \
  --docker-volume-size 5 \
  --network-driver flannel \
  --coe kubernetes \
  --tls-disabled
```

cluster_user_trust: Recreate cluster

- Delete old cluster
- [cmd/delete-cluster.sh]

```
magnum cluster-delete k8s_cluster
```

- Create cluster with new template
- [cmd/create-cluster-registry.sh]

```
magnum cluster-create \
  --name k8s_cluster \
  --cluster-template k8s_template \
  --node-count 1
```

cluster_user_trust: Recreate cluster

- Delete old cluster
- [cmd/delete-cluster.sh]

```
magnum cluster-delete k8s_cluster
```

- Create cluster with new template
- [cmd/create-cluster-registry.sh]

```
magnum cluster-create \
  --name k8s_cluster \
  --cluster-template k8s_registry \
  --node-count 1
```

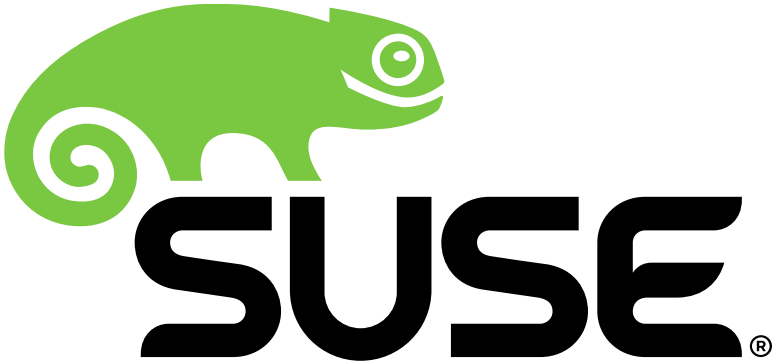
Error: cluster_user_trust=false

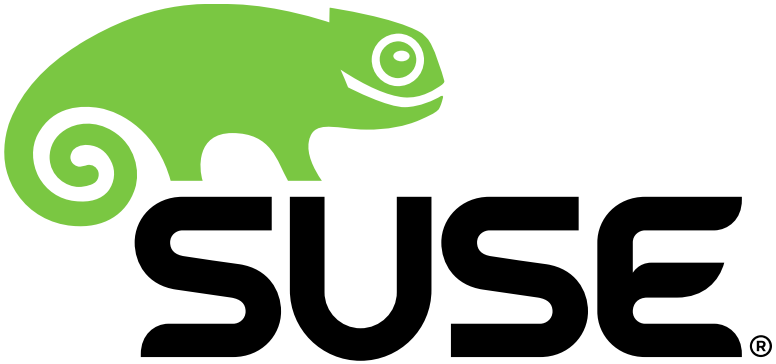
- Error message in `status_reason:

```
This cluster can only be created with
trust/cluster_user_trust = True in
magnum.conf
```

- Fix
  - Set `trustee["cluster_user_trust"] = True` (raw mode setting in Crowbar)
  - Run each cluster in a dedicated project with tight quotas to limit attack surface

Thank you.


SUSE®
We adapt. You succeed.



Corporate Headquarters
Maxfeldstrasse 5
90409 Nuremberg
Germany

+49 911 740 53 0 (Worldwide)
www.suse.com

Join us on:
www.opensuse.org