K8S / helm basic commands for ONAP integration

Frankfurt Notes

In addition to the mechanisms from El Alto for deploy/undeply the standard helm commands also work. For example in the case to upgrade a specific component like AAI without having to do a helm delete/heml deploy you can use the following:

Example with aai charts

Precondition:

> helm deploy has previously been run with override files on the command line so that the .helm/plugins/deploy/cache has been populated
>
> git clone aai to an aai_oom directory since it is a recursive submodule of oom and upgrade using these newly cloned charts

Helm command:

> helm upgrade -i onap-aai ./aai_oom --namespace onap --timeout 900 -f ${HOME}/.helm/plugins/deploy/cache/onap/global-overrides.yaml -f ${HOME}/.helm/plugins/deploy/cache/onap-subcharts/aai/subchart-overrides.yaml

El Alto Notes with new Deploy/Undeploy plugin from OOM Team

helm deploy dev local/onap -f /root/integration-override.yaml --namespace onap

For slower cloud environment use this to use longer interval for readiness

helm deploy dev local/onap -f /root/oom/kubernetes/onap/resources/environments/public-cloud.yaml -f /root/integration-override.yaml --namespace onap

Example per prodjest with SO:

helm deploy dev-so local/onap -f /root/oom/kubernetes/onap/resources/environments/public-cloud.yaml -f /root/integration-override.yaml --namespace onap --verbose

If you are using the SNAPSHOT image override file:

helm deploy dev-sdnc local/onap -f /root/oom/kubernetes/onap/resources/environments/public-cloud.yaml -f /root/integration-override.yaml -f /root/integration/deployment/heat/onap-rke/staging-image-override.yaml --namespace onap --verbose

1. After editing a chart
    a. cd /root/oom/kubernetes
    b. make project
        i. note that for cds/sdnc you need to do make cds; make sdnc
    c. make onap
2. helm del project --purge
    a. helm list -a to confirm its gone
    b. also check pvc's for applications like sdnc/appc and kubectl -n onap delete pvc any remaining ones
        i. kubectl -n onap get pv | grep project
        ii. kubectl -n onap get pvc | grep project
        iii. ...
        iv. "delete /dockerdata-nfs/dev-project"
    c. Cleanup shared cassandra (aai, sdc) and shared maiadb (sdnc, so)
    d. /root/integration/deployment/heat/onap-rke/cleanup.sh project(without dev-)
        i. example: ./cleanup.sh sdc
        ii. this script cleans up the shared cassandra and mariadb as well as pvc, pv, jobs etc.
        iii. if you get an error when doing aai or sdc check to make sure cassandra cleaned up correctly. We have known problem where the cluster does not let schema's to be replicated and you get a Timeout back to cleanup.sh
3. Rebuild helm charts as necessary
    a. cd /root/oom/kubernetes
    b. make project
    c. make onap
4. helm deploy dev local/onap -f /root/oom/kubernetes/onap/resources/environments/public-cloud.yaml -f /root/integration-override.yaml --namespace onap --verbose
5. list pods and ports (with k8 host)
    a. kubectl -n onap get pods -o=wide
    b. kubectl -n onap get services
6. Find out why pod is stuck in initializing or crash loopback
    a. kubectl -n onap describe pod dev-blah-blah-blah

b. kubectl -n onap logs dev-blah-blah-blah

complete removal steps (same as Beijing)

Faster method to do a delete for reinstall

kubectl delete namespace onap

kubectl delete pods -n onap --all

kubectl delete secrets -n onap --all

kubectl delete persistentvolumes -n onap --all

~~kubectl -n onap delete clusterrolebindings --all~~

helm del --purge dev

helm list -a

helm del --purge dev-[project] use this if helm list -a shows lingering releases in DELETED state

if you have pods stuck terminating for a long time

kubectl delete pod --grace-period=0 --force --namespace onap --all

CDS Specific Notes (**Dublin**) - In Dublin release, the CDS charts are added as a subchat in OOM. However, the deployment of CDS charts is achieved as part of the SDN-C deployment in Dublin release. Thus, if any changes required to be made in the CDS chart the following steps taking: **"make cds**; **make sdnc** ; **make onap"**

Merge "Tune OpenDaylight parameters"

[oom.git] / kubernetes /

drwxr-xr-x		..	
-rw-r--r--	902	LICENSE	blob | history | raw
-rw-r--r--	2007	Makefile	blob | history | raw
-rw-r--r--	3033	README.md	blob | history | raw
drwxr-xr-x	-	aaf	tree | history
m---------	-	aai	history
drwxr-xr-x	-	appc	tree | history
drwxr-xr-x	-	cds	tree | history
drwxr-xr-x	-	clamp	tree | history
drwxr-xr-x	-	cl1	tree | history
drwxr-xr-x	-	common	tree | history
drwxr-xr-x	-	config	tree | history

SDNC Values.yaml chart in OOM

```
# dependency / sub-chart configuration
cds:
  enabled: true
```

Beijing Notes

```
kubectl config get-contexts
```

helm list

root@k8s:~# helm list

NAME	REVISION	UPDATED	STATUS	CHART	NAMESPACE
dev	2	Mon Apr 16 23:01:06 2018	FAILED	onap-2.0.0	onap
dev	9	Tue Apr 17 12:59:25 2018	DEPLOYED	onap-2.0.0	onap

helm repo list

NAME	URL
stable	https://kubernetes-charts.storage.googleapis.com
local	http://127.0.0.1:8879

#helm upgrade -i dev local/onap --namespace onap -f onap/resources/environments/integration.yaml

```
helm upgrade -i dev local/onap --namespace onap -f integration-override.yaml
```

to upgrade robot

a config upgrade should use the local/onap syntax to let K8 decide based on the parent chart (local/onap)

```
helm upgrade -i dev local/onap --namespace onap -f integration-override.yaml
```

if docker container changes use the enable:false/true

```
helm upgrade -i dev local/onap --namespace onap -f integration-override.yaml --set robot.enabled=false
helm upgrade -i dev local/onap --namespace onap -f integration-override.yaml --set robot.enabled=true
```

 # if both the config and the docker container changes use the enable:false, do the make component, make onap then enable:true

```
helm upgrade -i dev local/onap --namespace onap -f /root/integration-override.yaml --set robot.enabled=false
```

Confirm the assets are removed with get pods , get pv, get pvc, get secret, get configmap for those pieces you dont want to preserve

cd /root/oom/kubernetes

make robot

make onap

```
helm upgrade -i dev local/onap --namespace onap -f /root/integration-override.yaml --set robot.enabled=true
kubectl get pods --all-namespaces -o=wide
```

to check status of a pod like robots pod

```
kubectl -n onap describe pod dev-robot-5cfddf87fb-65zvv
pullPolicy: Always IfNotPresent option to allow us to
```

Faster method to do a delete for reinstall

kubectl delete namespace onap

kubectl delete pods -n onap --all

kubectl delete secrets -n onap --all

kubectl delete persistentvolumes -n onap --all

kubectl -n onap delete clusterrolebindings --all

helm del --purge dev

helm list -a

helm del --purge dev-[project] use this if helm list -a shows lingering releases in DELETED state

if you have pods stuck terminating for a long time

kubectl delete pod --grace-period=0 --force --namespace onap --all

# of NAME=dev release

```
helm upgrade -i dev local/onap --namespace onap -f integration-override.yaml
```

To test with a smaller ConfigMap try to disable some things like:

helm upgrade -i dev local/onap --namespace onap -f /root/integration-override.yaml --set log.enabled=false --set clamp.enabled=false --set pomba.enabled=false --set vnfsdk.enabled=false

(aaf is needed by alot of modules in Casablanca but this is a near equivalent)

helm upgrade -i dev local/onap --namespace onap -f /root/integration-override.yaml --set log.enabled=false --set aaf.enabled=false --set pomba.enabled=false --set vnfsdk.enabled=false

Note: setting log.enabled=false means that you will need to hunt down /var/log/onap logs on each docker container - instead of using the kibana search on the ELK stack deployed to port 30253 that consolidates all onap logs

Slower method to delete full deploy

helm del dev --purge

kubectl get pods --all-namespaces -o=wide

look for all Terminating to be gone and wait till they are

kubectl -n onap get pvc

look for persistant volumes that have not been removed.

```
kubectl -n onap delete pvc  dev-sdnc-db-data-dev-sdnc-db-0
```

dev-sdnc is the name from the left of the get pvc command

same for pv (persistant volumes)

```
kubectl -n onap get pv
```

```
kubectl -n onap delete  pv  pvc-c0180abd-4251-11e8-b07c-02ee3a27e357
```

#same for pv, pvc, secret, configmap, services

```
kubectl get pods --all-namespaces -o=wide
```

```
kubectl delete  pod dev-sms-857f6dbd87-6lh9k -n onap (stuck terminating pod )
```

full install

of NAME=dev instane

```
helm upgrade -i dev local/onap --namespace onap -f integration-override.yaml
```

```
# update vm_properties.py
```

```
# robot/resources/config/eteshare/vm_properties.py

# cd to oom/kuberneties
```

Remember: Do the enabled=false BEFORE doing the make onap so that the kubectl processing will use the old chart to delete the POD

```
#

# helm upgrade -i dev local/onap --namespace onap -f integration-override.yaml - this would just redeploy robot
becuase its configMap only
```

Container debugging commands

kubectl -n onap logs pod/dev-sdnc-0 -c sdnc