

OOM Helm (un)Deploy plugins

- [Overview](#)
- [Install Helm Plugins \(deploy & undeploy\)](#)
- [Deploying ONAP](#)
 - [Deploy from public Helm Chart Repository](#)
 - [Deploy from cloned OOM codebase](#)
 - [Advanced Options](#)
 - [Customizing ONAP deployment using overrides *](#)
 - [Update \(or install\) a specific ONAP component *](#)
 - [Helm Deploy plugin logs](#)
- [Undeploying ONAP](#)
 - [Undeploy entire ONAP deployment](#)
 - [Undeploy specific ONAP component](#)

Overview

In the Casablanca release ONAP has become very large with many Helm charts (~6 x that of Amsterdam). Every Helm chart contains some amount of external configuration and, unfortunately, there is a limit to the amount of configuration that can exist in a Helm chart (1 MB). As of Casablanca, we in ONAP have exceeded this limit.

ONAP is installed as an umbrella (ie. parent) chart containing many subcharts, each with configuration. The total amount of configuration that resides in configmaps within K8s (exceeding 1MB) causes an installation of ONAP to fail. To work around this issue, a Helm plugin has been introduced that will install/upgrade ONAP by deploying the parent chart and each subchart within its own Helm "release". It is important to note that all releases must be deployed within the same Kubernetes namespace in order for communication between the components to succeed.

Disclaimer:

The plugins described here were introduced to address config map limitations in the [Casablanca release](#). There are [alternative](#) projects, such as <https://github.com/roboll/helmfile>, that can also solve this problem.

The decision to not use an existing project was due to:

- not wanting to introduce a new project-specific deployment specification this late in the release cycle
- the desire to stay as close to an existing Helm solution as possible (we anticipate significant improvements in Helm 3)

That said, the use of **deploy** and **undeploy** plugins can be viewed as a temporary solution. In their current state, they have not been hardened (eg. resilient to networking errors) but are made available to unblock installation of ONAP. To avoid networking errors that can cause some of the sub-charts to fail to deploy, it is recommend that Helm **deploy** and **undeploy** commands execute from within the same network (e.g Rancher node or Jumpnode) as the K8s cluster you are deploying to.

Install Helm Plugins (deploy & undeploy)

Clone [oom repository](#)

Copy [oom/kubernetes/helm/plugins](#) directory into your local `~/.helm/` folder.

Manual Installation

```
# ubuntu
sudo cp -R ~/oom/kubernetes/helm/plugins/ ~/.helm
# mac
sudo cp -R ~/oom/kubernetes/helm/plugins/* ~/.helm/plugins
```

Verify plugins installed correctly

Execute '**helm**' (no arguments) should show both '**deploy**' and '**undeploy**' in list of available commands.

```
>sudo helm
```

```
.....
```

Usage:
helm [command]

Available Commands:
...

dependency manage a chart's dependencies
deploy install (upgrade if release exists) parent chart and subcharts as separate but related releases
fetch download a chart from a repository and (optionally) unpack it in local directory
...

template locally render templates
test test a release
undeploy delete parent chart and subcharts that were deployed as separate releases
upgrade upgrade a release
version print the client/server version information

Deploying ONAP

Deploy from public Helm Chart Repository

Example is using one of the public Helm Chart repos <https://nexus.onap.org/content/sites/> (ie. staging).

Prerequisite:

1. helm repo add staging <https://nexus.onap.org/content/sites/oom-helm-staging/>

Deploy ONAP from staging repository (using default configuration values defined in onap/values.yaml)

```
helm deploy demo staging/onap --namespace onap
```

Deploy from cloned OOM codebase

Prerequisites:

1. clone [oom repository](#)
2. cd oom/kubernetes
3. make repo
4. make; make onap
5. use sudo for helm commands except when running root

Deploy ONAP from OOM codebase using local Helm Chart Repository (default configuration values defined in onap/values.yaml)

```
sudo helm deploy demo local/onap --namespace onap
```

or

Deploy ONAP from OOM codebase using local file changes (default configuration values defined in onap/values.yaml)

```
sudo helm deploy demo ./onap --namespace onap
```

Results

ONAP pods deployed into the 'onap' Kubernetes namespace. Each application project is installed as a Helm Release that is prefixed by the parent release name (e.g. "demo-")

> helm list

NAME CHART	REVISION NAMESPACE	UPDATED	STATUS	
demo	1	Wed Sep 19 12:04:52 2018	DEPLOYED	onap-
2.0.0	onap			
demo-aaf	1	Wed Sep 19 12:04:57 2018	DEPLOYED	aaf-
2.0.0	onap			
demo-aai	1	Wed Sep 19 12:05:09 2018	DEPLOYED	aai-
2.0.0	onap			
demo-appc	1	Wed Sep 19 12:05:22 2018	DEPLOYED	appc-
2.0.0	onap			
demo-clamp	1	Wed Sep 19 12:05:28 2018	DEPLOYED	clamp-
2.0.0	onap			
demo-cli	1	Wed Sep 19 12:05:33 2018	DEPLOYED	cli-
2.0.0	onap			
demo-consul	1	Wed Sep 19 12:05:38 2018	DEPLOYED	consul-
2.0.0	onap			
demo-dcaegen2	1	Wed Sep 19 12:05:48 2018	DEPLOYED	dcaegen2-
2.0.0	onap			
demo-dmaap	1	Wed Sep 19 12:05:54 2018	DEPLOYED	dmaap-
2.0.0	onap			
demo-esr	1	Wed Sep 19 12:05:58 2018	DEPLOYED	esr-
2.0.0	onap			
demo-log	1	Wed Sep 19 12:06:03 2018	DEPLOYED	log-
3.0.0	onap			
demo-msb	1	Wed Sep 19 12:06:08 2018	DEPLOYED	msb-
2.0.0	onap			
demo-multicloud	1	Wed Sep 19 12:06:13 2018	DEPLOYED	multicloud-
2.0.0	onap			
demo-nbi	1	Wed Sep 19 12:06:18 2018	DEPLOYED	nbi-
2.0.0	onap			
demo-oof	1	Wed Sep 19 12:06:23 2018	DEPLOYED	oof-
2.0.0	onap			
demo-policy	1	Wed Sep 19 12:06:30 2018	DEPLOYED	policy-
2.0.0	onap			
demo-pomba	1	Wed Sep 19 12:06:41 2018	DEPLOYED	pomba-
2.0.0	onap			
demo-portal	1	Wed Sep 19 12:06:49 2018	DEPLOYED	portal-
2.0.0	onap			
demo-robot	1	Wed Sep 19 12:06:55 2018	DEPLOYED	robot-
2.0.0	onap			
demo-sdc	1	Wed Sep 19 12:07:00 2018	DEPLOYED	sd-
2.0.0	onap			
demo-sdnc	1	Wed Sep 19 12:07:08 2018	DEPLOYED	sdnc-
2.0.0	onap			
demo-so	1	Wed Sep 19 12:07:16 2018	DEPLOYED	so-
2.0.0	onap			
demo-uui	1	Wed Sep 19 12:07:23 2018	DEPLOYED	uui-
2.0.0	onap			
demo-vfc	1	Wed Sep 19 12:07:29 2018	DEPLOYED	vfc-
2.0.0	onap			
demo-vid	1	Wed Sep 19 12:07:36 2018	DEPLOYED	vid-
2.0.0	onap			
demo-vnfsdk	1	Wed Sep 19 12:07:41 2018	DEPLOYED	vnfsdk-
2.0.0	onap			

Advanced Options

Note that any overrides that can be used using Helm **install** or **update** can be applied.

Customizing ONAP deployment using overrides *

Deploy using configuration overrides in the form of override file and --set

```
helm deploy demo ./onap --namespace onap -f ~/overrides.yaml --set vid.enabled=false
```

Update (or install) a specific ONAP component *

* Update (or install) a specific ONAP component (e.g. robot)

```
helm deploy demo-robot ./onap --namespace onap -f ~/overrides.yaml --set vid.enabled=false
```

* Note that in order for any changes to a Helm Chart to take affect, a **make** is required (e.g. "make; make onap" or "make robot; make onap")

Helm Deploy plugin logs

```
ubuntu@a-ld0:~$ sudo ls ~/.helm/plugins/deploy/cache/onap/logs/onap-
onap-aaf.log          onap-cli.log          onap-dmaap.log          onap-multicloud.log    onap-portal.
log                  onap-sniro-emulator.log onap-vid.log
onap-aai.log          onap-consul.log        onap-esr.log            onap-oof.log           onap-robot.
log                  onap-so.log           onap-vnfsdk.log         onap-policy.log        onap-sdc.
onap-appc.log         onap-contrib.log       onap-log.log            onap-policy.log        onap-sdc.
log                  onap-uui.log           onap-vvp.log            onap-pomba.log         onap-sdnc.
onap-clamp.log        onap-dcaegen2.log      onap-msb.log            onap-pomba.log         onap-sdnc.
log                  onap-vfc.log
```

Undeploying ONAP

Undeploy entire ONAP deployment

Delete entire ONAP deployment

```
helm undeploy demo --purge
```

Undeploy specific ONAP component

Undeploy a specific ONAP component (e.g. robot)

```
helm undeploy demo-robot --purge
```