Red Hat Advanced Cluster Management Demonstration Script

Hands-on part:

Exercise 1: Environment Preparation

Following the RHPDS request for the OCP4 ACM Hub item you can connect to the ACM console. See the email for details on the URLl and access details.

An example of the email looks like this, the URL for the ACM Console is at the bottom of it:

THESE ENVIRONMENTS WILL ONLY BE AVAILABLE 1 DAY AFTER PROVISIONING. THERE ARE NO EXTENSIONS.

Troubleshooting and Access issues:

Always refer to the instructions to understand how to properly access and navigate your environment. If you need help using the SSH client on your computer you can consult http://www.opentlc.com/ssh.html.

NOTICE: Your environment will expire and be deleted at 2021-02-18 00:00:00 -0500.

In order to conserve resources, we cannot archive or restore any data in this environment. All data will be lost upon expiration.

Here is some important information about your environment:

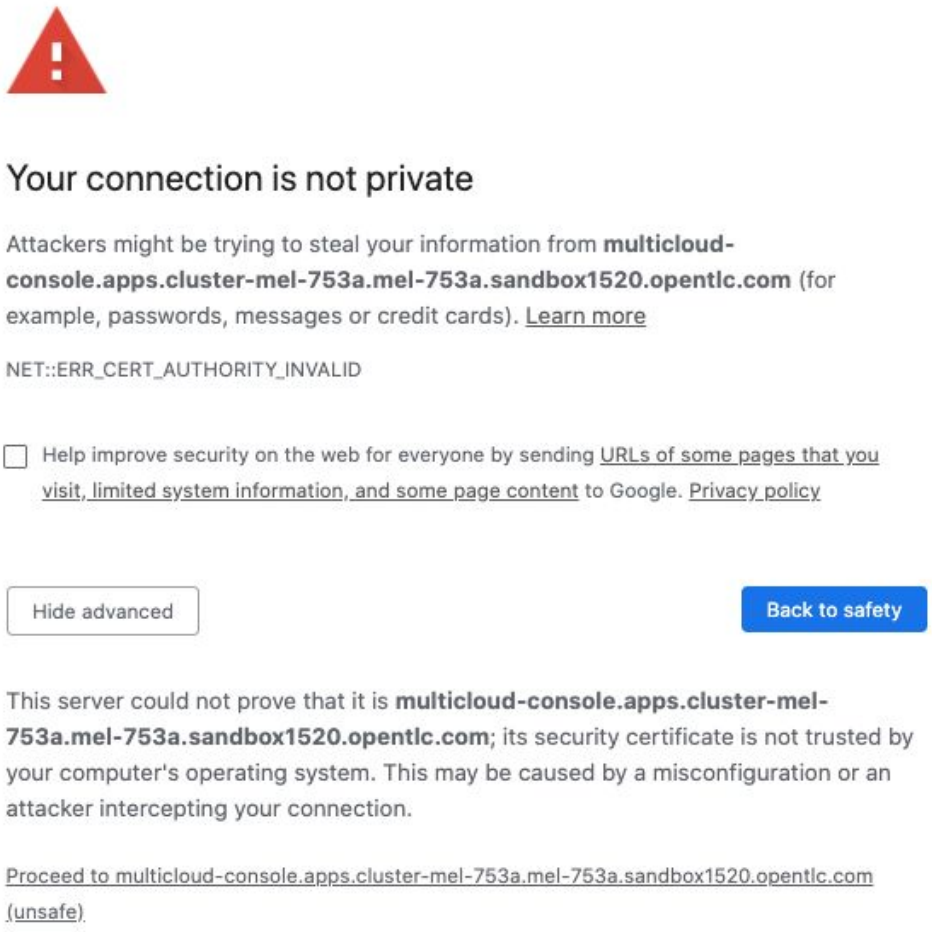
Openshift Master Console: https://console-openshift-console.apps.cluster-mel-753a.mel-753a.sandbox1520.opentlc.com
Openshift API for command line 'oc' client: https://api.cluster-mel-753a.mel-753a.sandbox1520.opentlc.com:6443
Download oc client from http://d3s3zqyaz8cp2d.cloudfront.net/pub/openshift-v4/clients/ocp/4.6.12/openshift-client-linux-4.6.12.tar.gz

HTPasswd Authentication is enabled on this cluster.
Users user1 .. user5 are created with password `openshift`
User `admin` with password `v85oooYWh0foyaEp` is cluster admin.

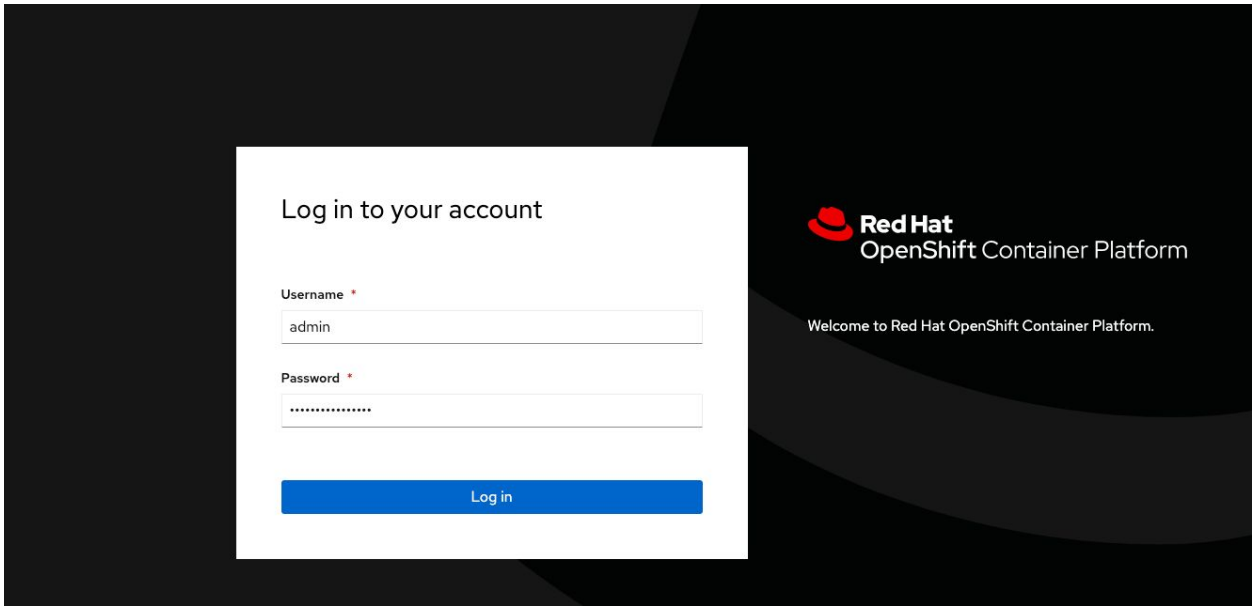
Your ACM console is available at:
https://multicloud-console.apps.cluster-mel-753a.mel-753a.sandbox1520.opentlc.com
You can access your bastion via SSH:
ssh sdelord-redhat.com@bastion.mel-753a.sandbox1520.opentlc.com

Make sure you use the username 'sdelord-redhat.com' and the password 'wcPTNpF688Ye' when prompted.

Navigate to the ACM URL and you will be prompted with the following screen

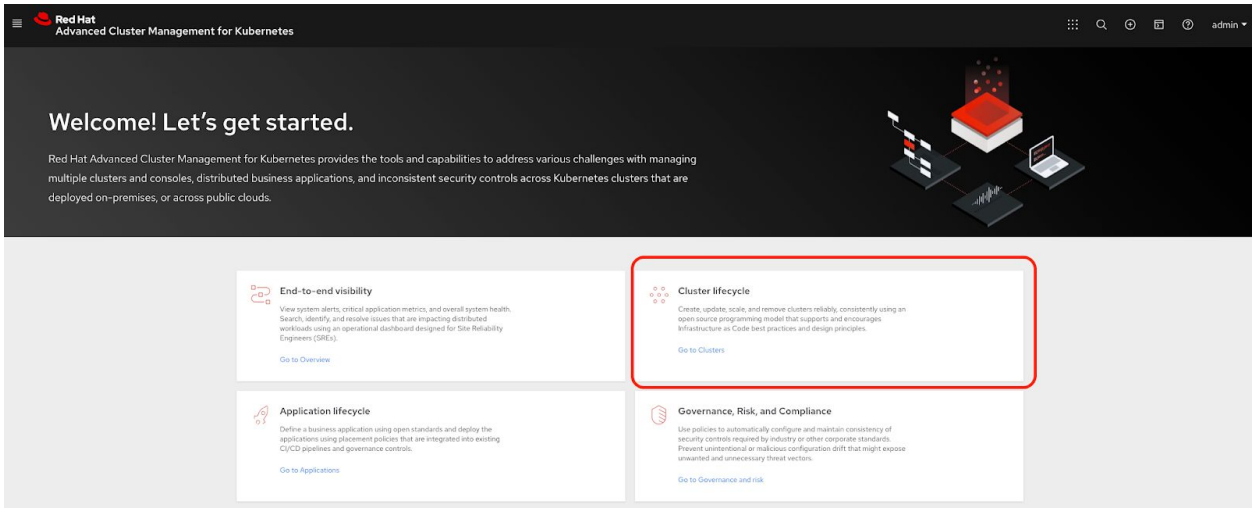


Click on proceed to multicloud-console.apps.<clustername>.opentlc.com to enter your credentials.

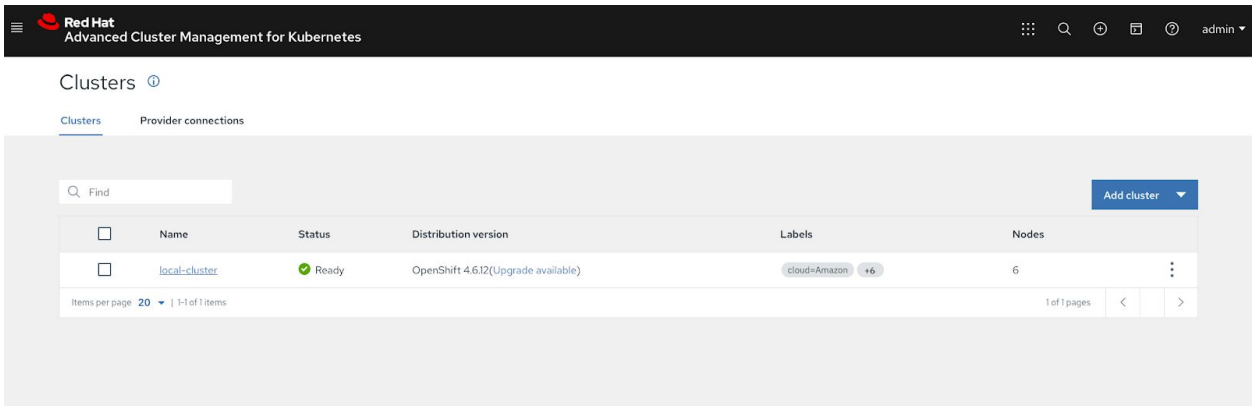


Once logged in to ACM, you will be presented with the Home page of ACM.

Navigate to the **Clusters lifecycle** screen (by clicking on Go to Clusters in the Cluster lifecycle Box - highlighted in Red in the screenshot below)



This will take you to another screen in which you will notice that the **HUB cluster** is already imported (local-cluster), we will use this later in the guide.

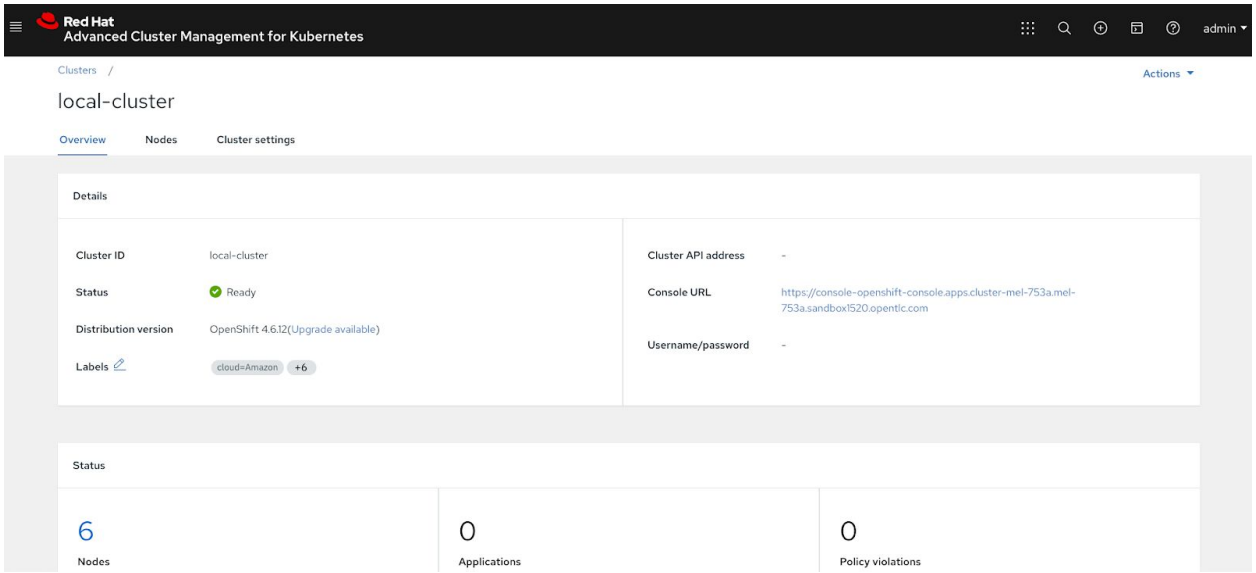


This is the end of the first activity. You can do the set of extra activities below to familiarise yourself with the environment.

Extra activities:

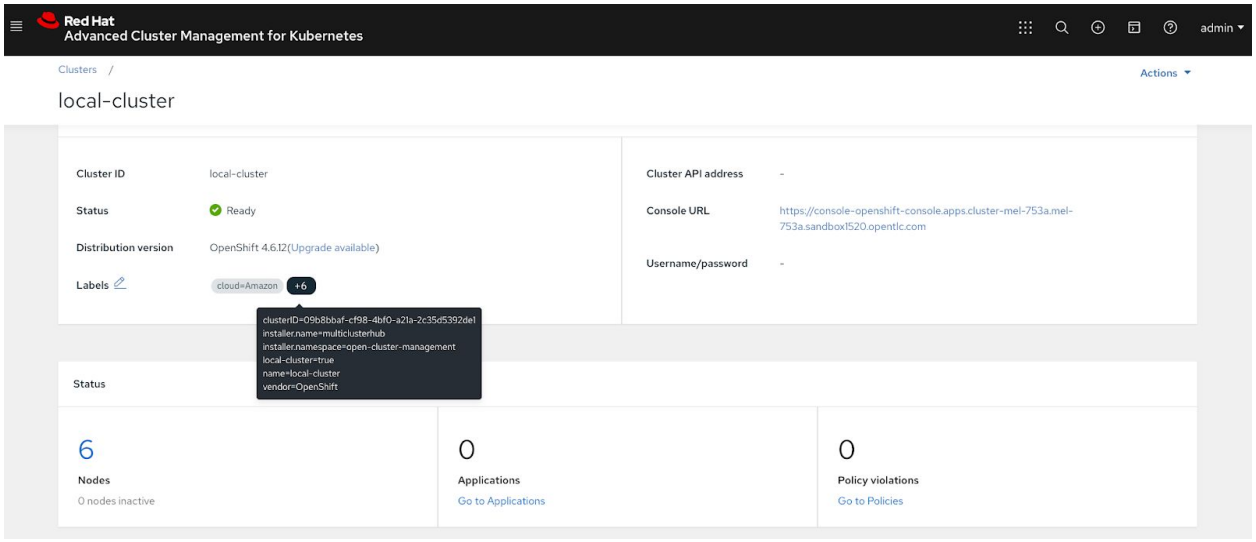
Extra activity 1- Discover the local-cluster

In the previous screenshot, click on local-cluster. This will take you to the detailed view of the local-cluster (Hub cluster where ACM is deployed).



The view displayed is the Overview tab, that shows the status of the cluster, how many nodes it has, how many Applications are deployed on it and how many Policy violations can be found on it.

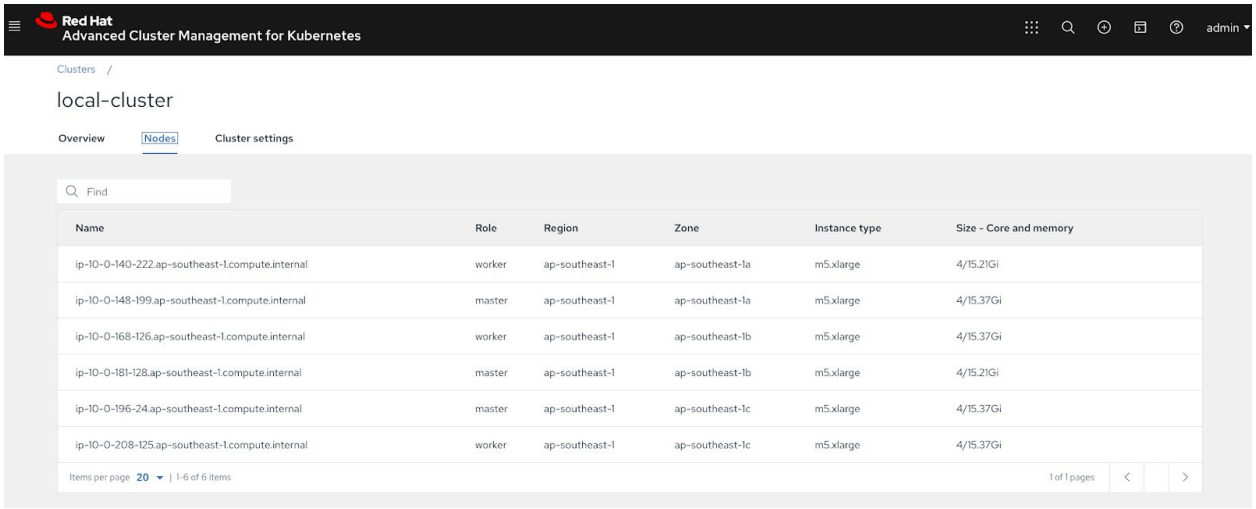
Please also notice the Labels associated with the local-cluster. For this, just drag your mouse over the cloud=Amazon + 6 bubble. Labels play an important role for placement of applications and policies, they will be created, updated and used as part of the next exercises.



Explore each of the other tabs (Nodes, Cluster settings).

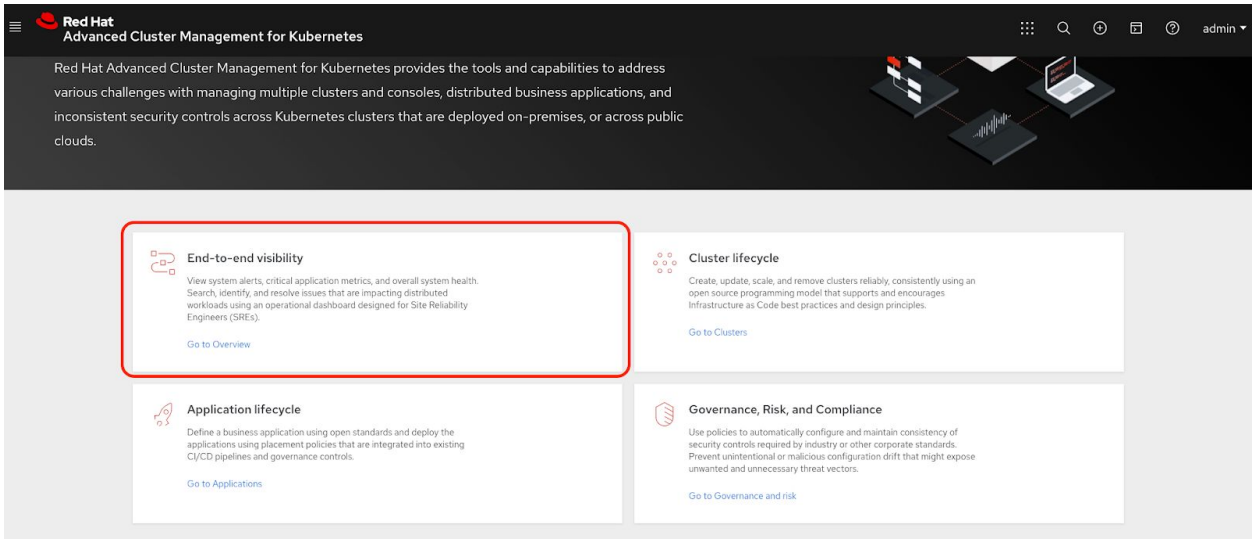
For example when you click on the Nodes tab, you will see the various types of Nodes, their role, where they are located, their size.

From the screen below, you can see that this OpenShift cluster is sitting in AWS in the ap-southeast-1 region and that all 6 nodes (workers and masters) are of the same instance type (m5.xlarge) and split across 3 different zones (1a, 1b and 1c) to provide resiliency within the cluster.

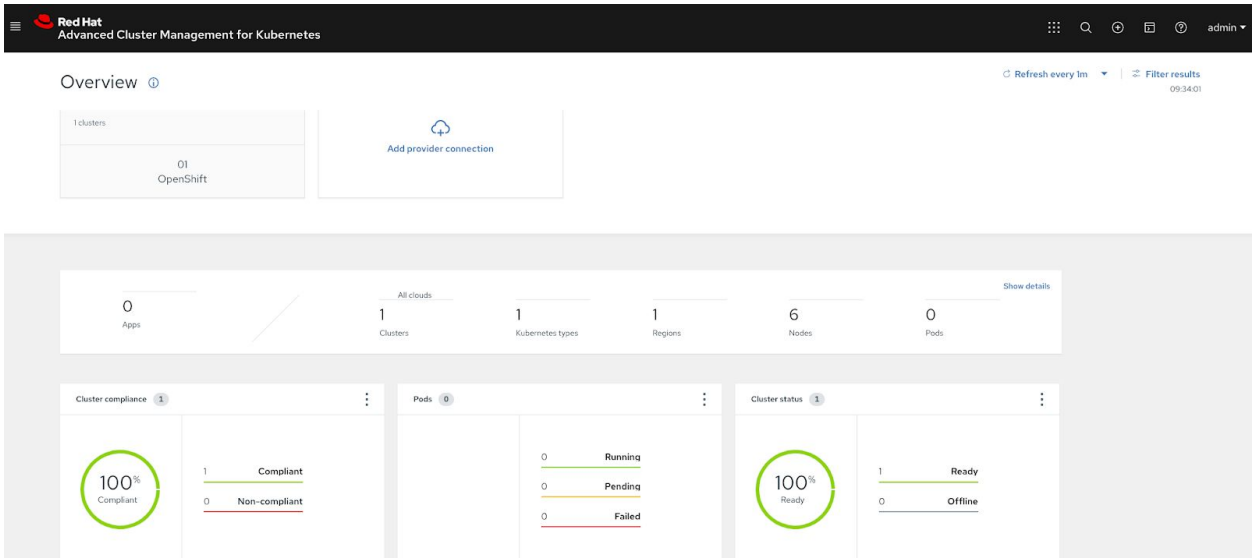


Extra activity 2 - Discover the Overview Screen

Go back to the ACM home page and click this time on the End-to-end visibility (Go to Overview) box.



This will then take you to another screen that looks like the screenshot below



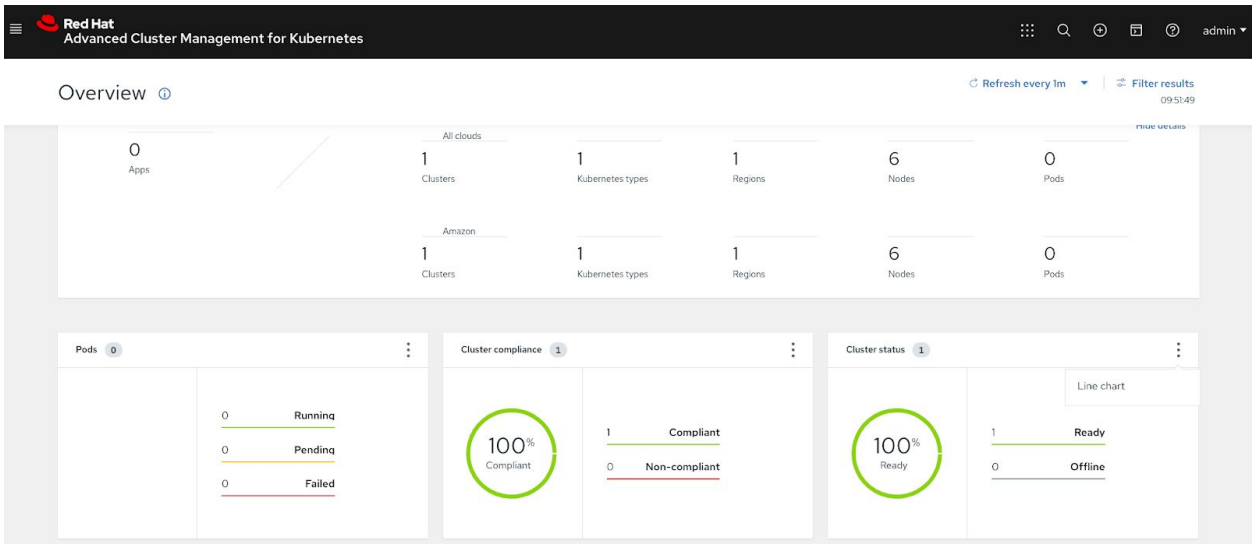
You can view the following information about your clusters on the Overview dashboard:

- Cluster, node, and pod counts across all clusters and for each provider
- Cluster status

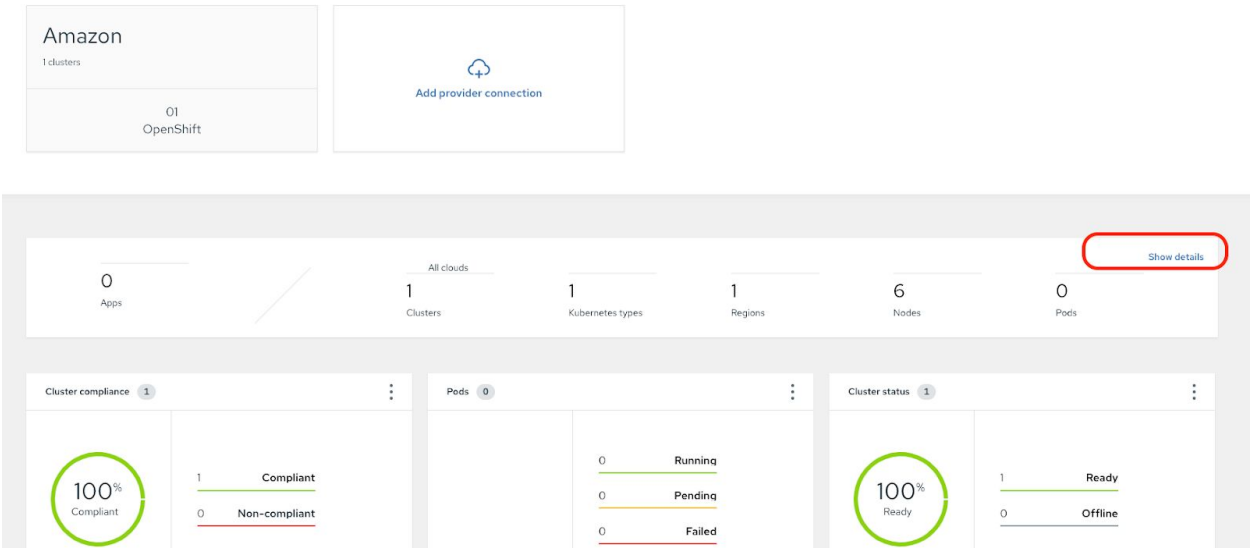
- Cluster compliance
- Pod status

You can personalize your view of the Overview dashboard by clicking and dragging to reorganize the cards.

For example in the screenshot below, I have changed the cluster status from Pie chart to Line chart (click on the 3 grey dots in the Cluster status box) and swapped around the Pods box and Cluster compliance box.



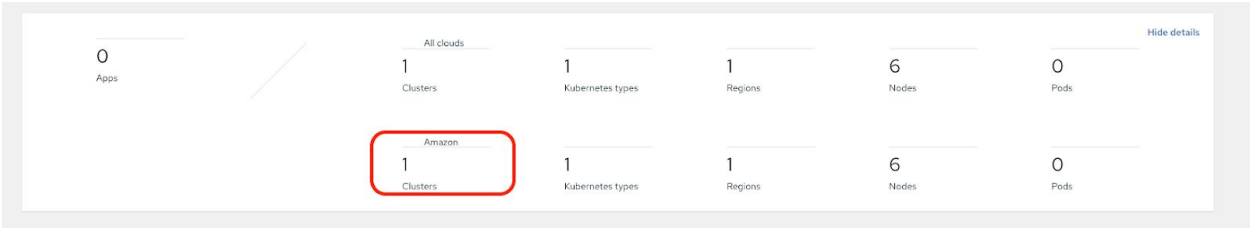
On the central box, click on Show details

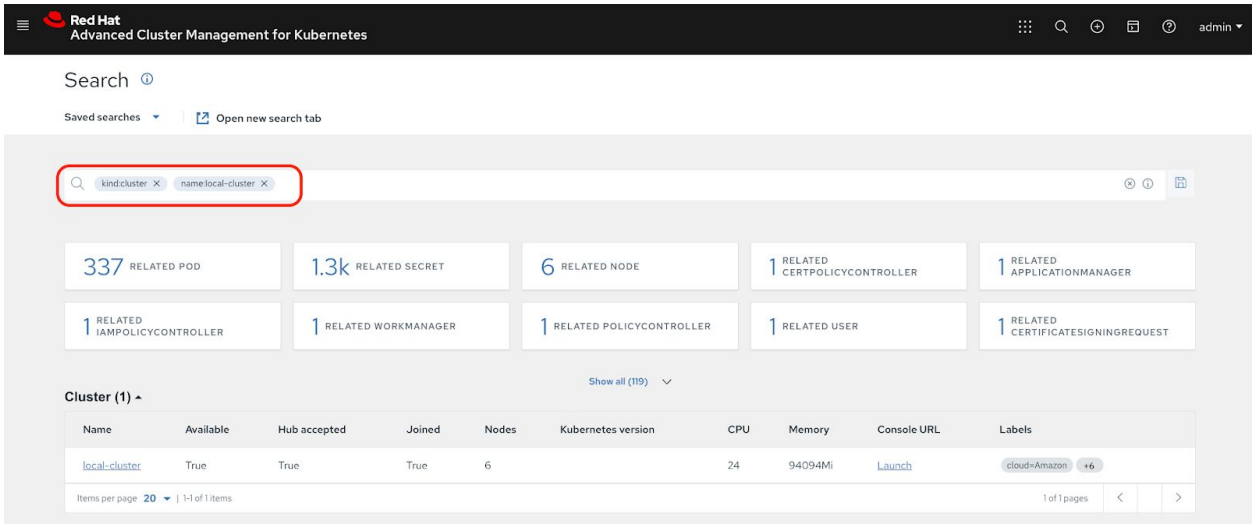


This will expand the box with a summary of the various clusters.

Click on the Amazon 1 Clusters box.

It will take you to the search view (e.g clicking performed a search on cluster & local-cluster).



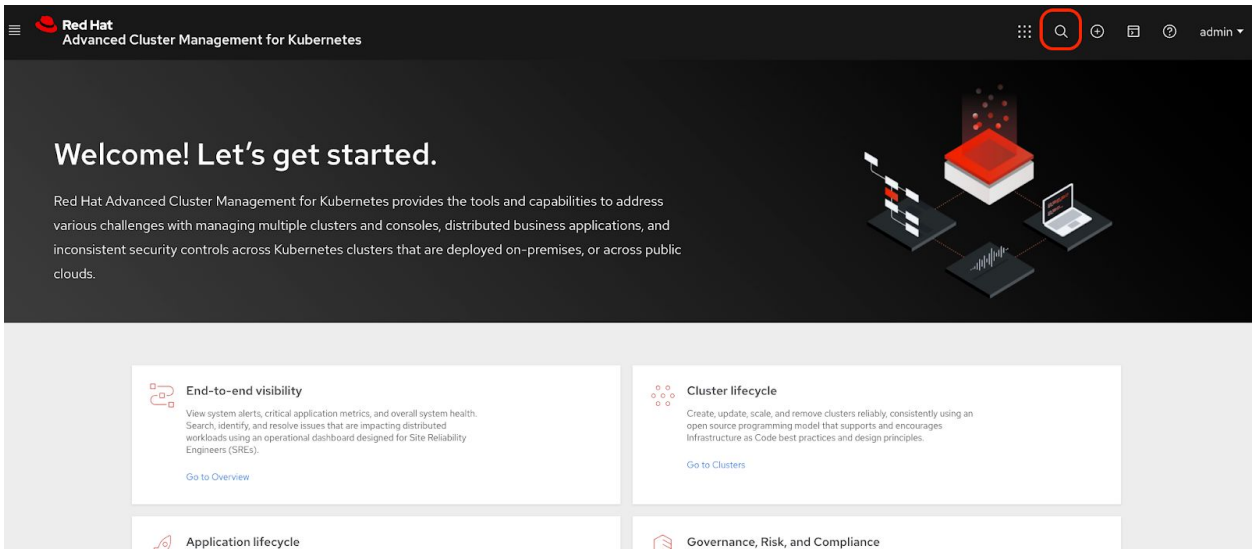


The next extra activity will show you how to perform this search directly and how to explore some of the results for it.

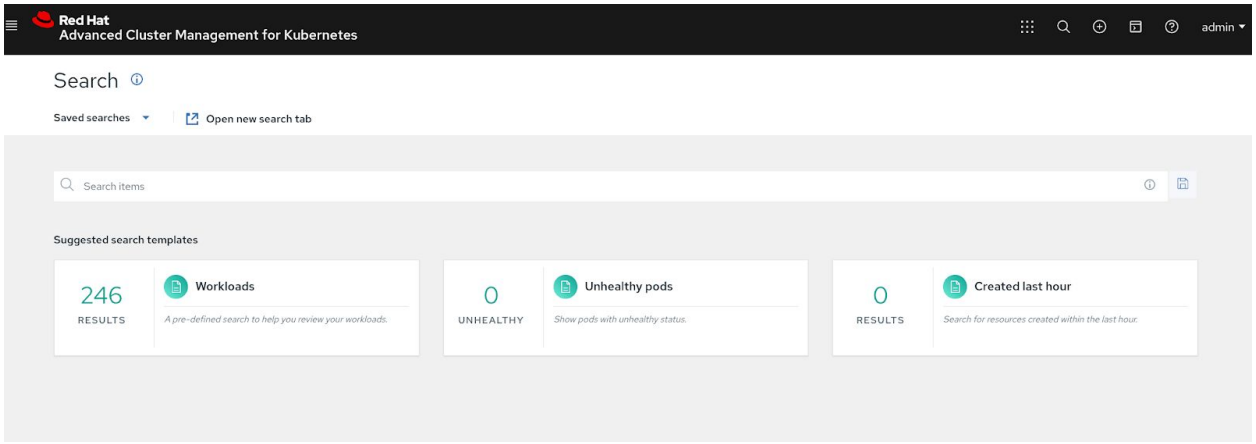
Extra activity 3 - Perform a search & explore results for the search

In the previous extra activity, we performed by clicking on the overview environment a search related to kind:cluster & name:local-cluster.

We can also perform this action directly. Going back to the home screen, you can click on the search icon directly in the top right corner.



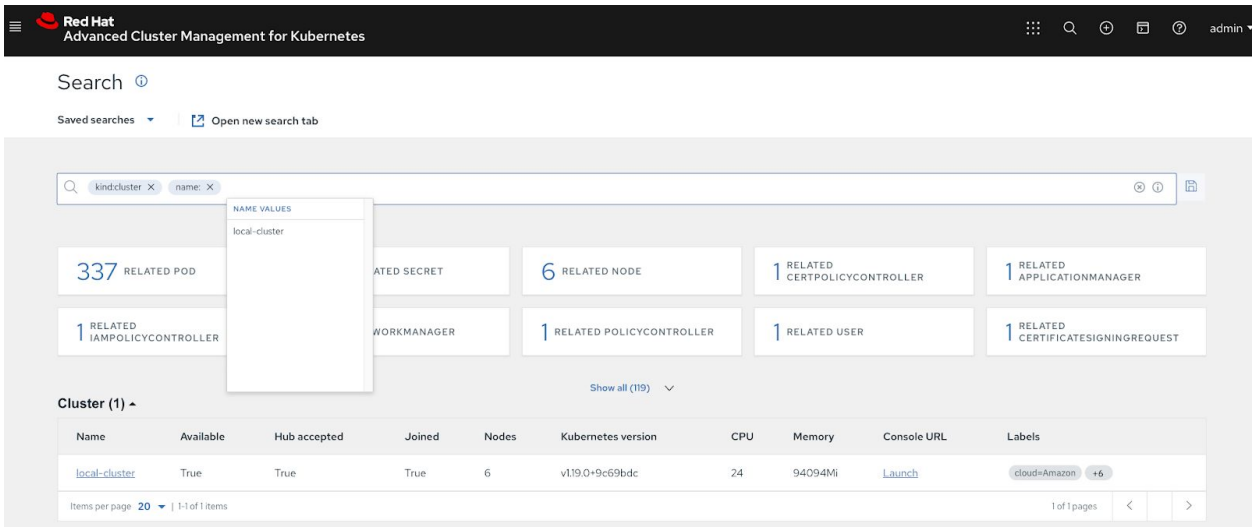
This takes you to the search page.



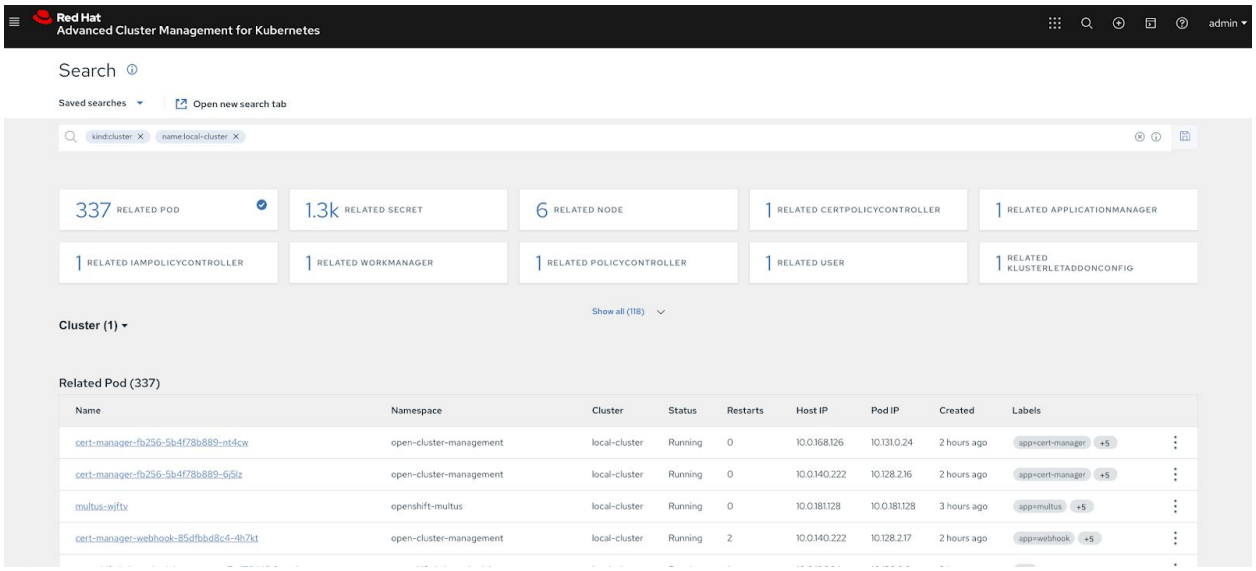
By clicking on the Search items box, you will see a filter tab appearing.

You can choose between many FILTERS and associated FILTER VALUES. Once you've selected the FILTERS, click on it and then click next to the filter and all associated values will appear.

For example, in the screenshot below I have performed the kind (FILTERS) cluster (KIND VALUE) search category and I am now performing the name (FILTER) local-cluster (NAME VALUE)

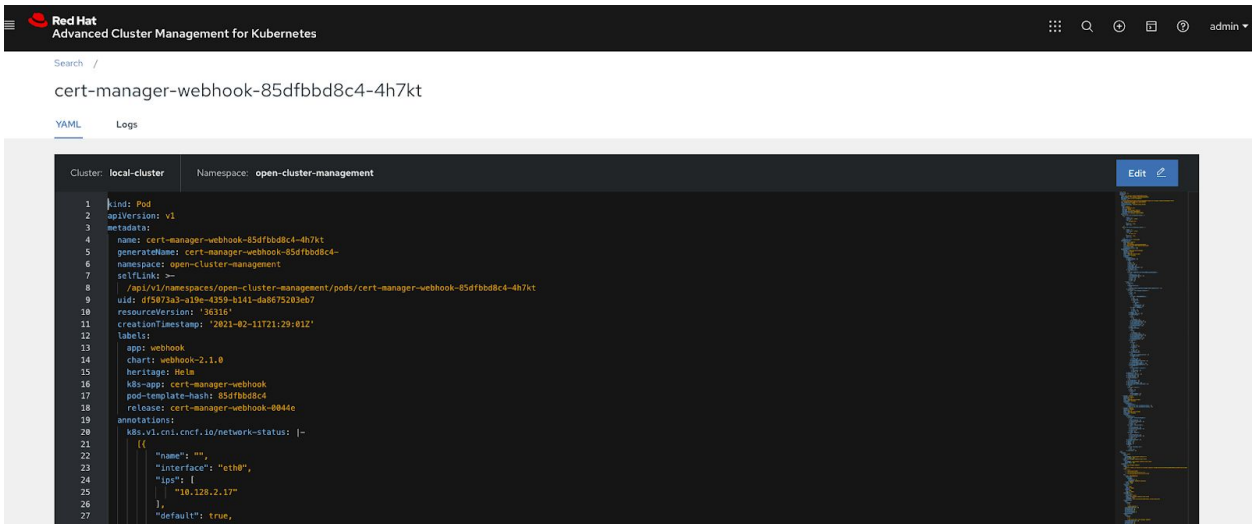


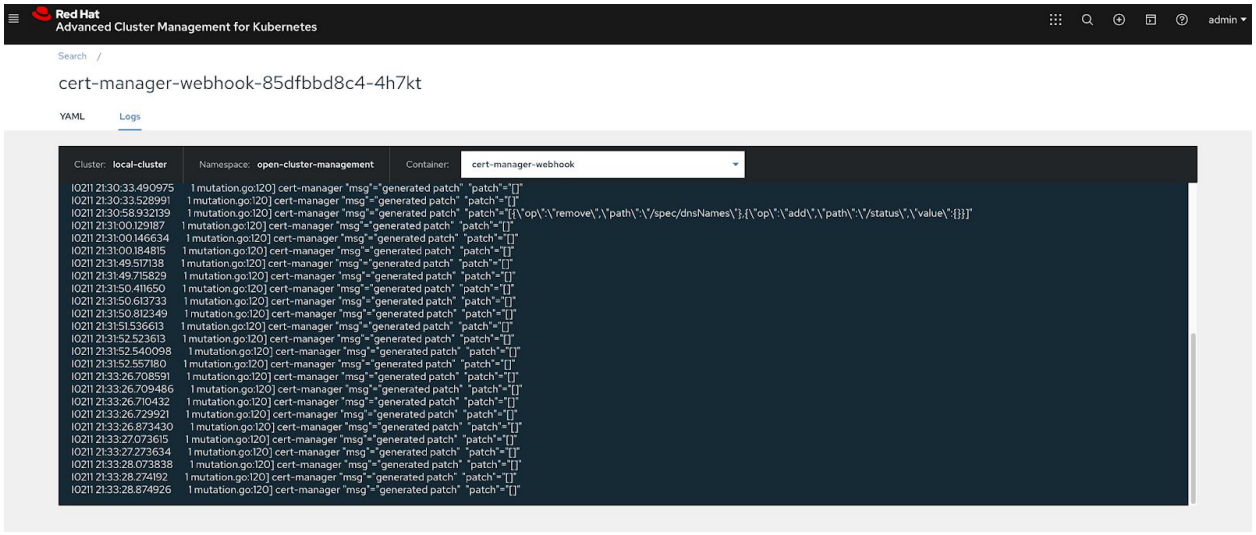
Click on the RELATED POD box, we can see all the Pods deployed on the local-cluster.



You can then drill down on any of the Pods there.

Clicking on the cert-manager-webhook-85dfbb8c4-4h7kt Pod takes me to another screen with the YAML view of the Pod and the associated Logs.


Red Hat
Advanced Cluster Management for Kubernetes
admin
Search /
cert-manager-webhook-85dfbbd8c4-4h7kt
YAML
Logs
Cluster: local-cluster
Namespace: open-cluster-management
Edit
kind: Pod
apiVersion: v1
metadata:
name: cert-manager-webhook-85dfbbd8c4-4h7kt
generateName: cert-manager-webhook-85dfbbd8c4-
namespace: open-cluster-management
selfLink: >-
/api/v1/namespaces/open-cluster-management/pods/cert-manager-webhook-85dfbbd8c4-4h7kt
uid: df5073a3-a19e-4359-b141-da8675203eb7
resourceVersion: '36316'
creationTimestamp: '2021-02-11T21:29:01Z'
labels:
app: webhook
chart: webhook-2.1.0
heritage: Helm
k8s-app: cert-manager-webhook
pod-template-hash: 85dfbbd8c4
release: cert-manager-webhook-0044e
annotations:
k8s.v1.cni.cncf.io/network-status: |-
[{
"name": "",
"interface": "eth0",
"ips": [
"10.128.2.17"
],
"default": true,

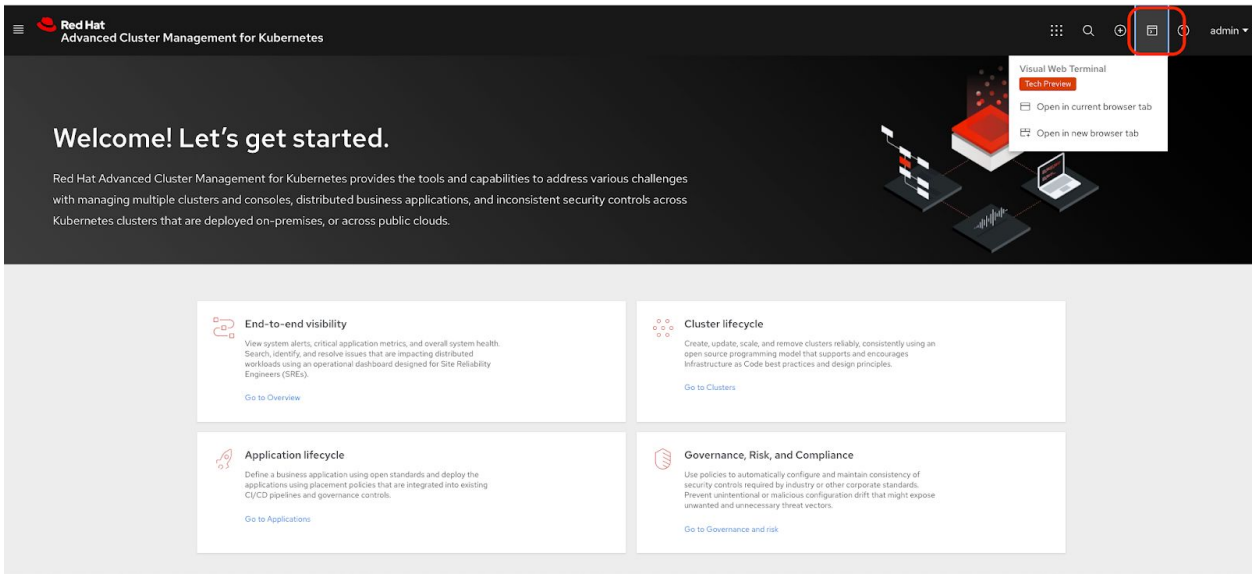

Red Hat
Advanced Cluster Management for Kubernetes
admin
Search /
cert-manager-webhook-85dfbbd8c4-4h7kt
YAML
Logs
Cluster: local-cluster
Namespace: open-cluster-management
Container: cert-manager-webhook
I0211 21:30:33.490975 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:30:33.528991 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:30:58.932139 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[{\"op\":\"remove\",\"path\":\"/spec/dnsNames\"},{\"op\":\"add\",\"path\":\"/status\",\"value\":{}}]"
I0211 21:31:00.129187 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:31:00.146634 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:31:00.184815 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:31:49.517138 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:31:49.715829 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:31:50.411650 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:31:50.613733 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:31:50.812349 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:31:51.536613 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:31:52.523613 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:31:52.540098 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:31:52.557180 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:33:26.708591 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:33:26.709486 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:33:26.710432 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:33:26.729921 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:33:26.873430 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:33:27.073615 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:33:27.273634 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:33:28.073838 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:33:28.274192 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"
I0211 21:33:28.874926 1 mutation.go:120] cert-manager "msg"="generated patch" "patch"="[]"

Extra activity 4 - Use the Visual Web Terminal to run commands from a central location

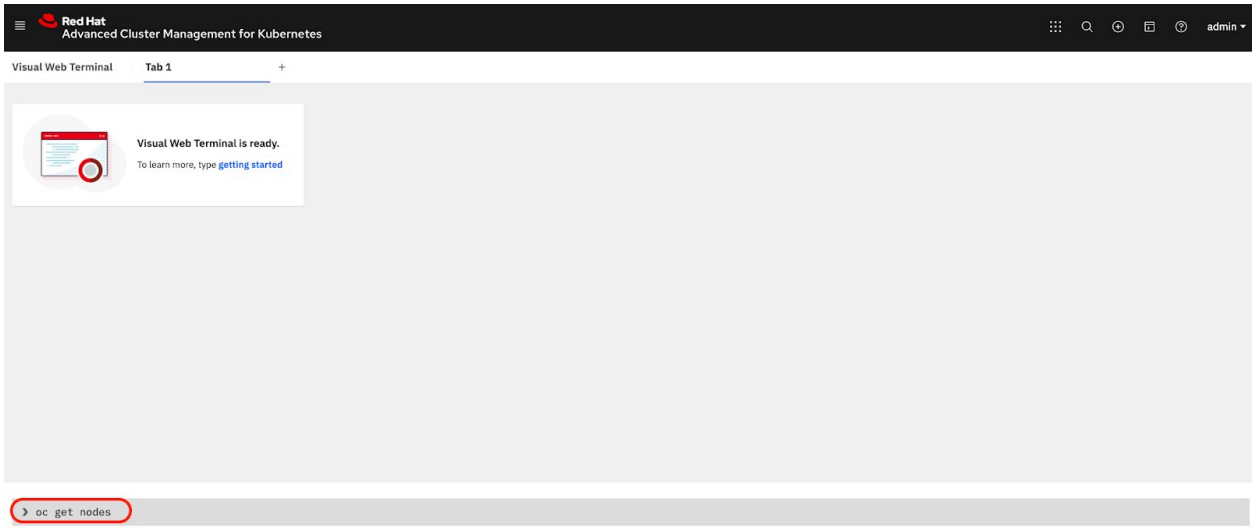
The Visual Web Terminal is a tech preview feature. You can use the Visual Web Terminal to run many commands across your cluster(s).

Run commands and get outputs without leaving the entire console. You can run multiple commands, similar to what you can do in a standard terminal.

With Visual Web Terminal, commands that return cluster resource data are displayed visually in interactive tabular format, rather than plain text that is displayed in a standard terminal. You can see the data, click on a row, and then see more details about that resource.

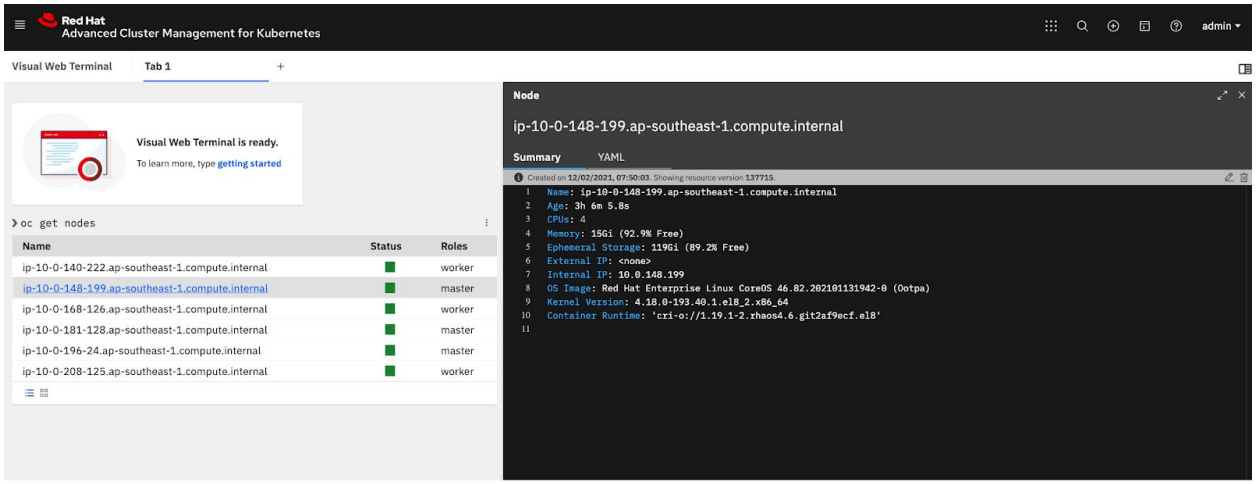


You will then be able to run commands on the various clusters. For example, type oc get nodes (as per the screenshot).



Results will be displayed and you can then interact directly with them by clicking on them.

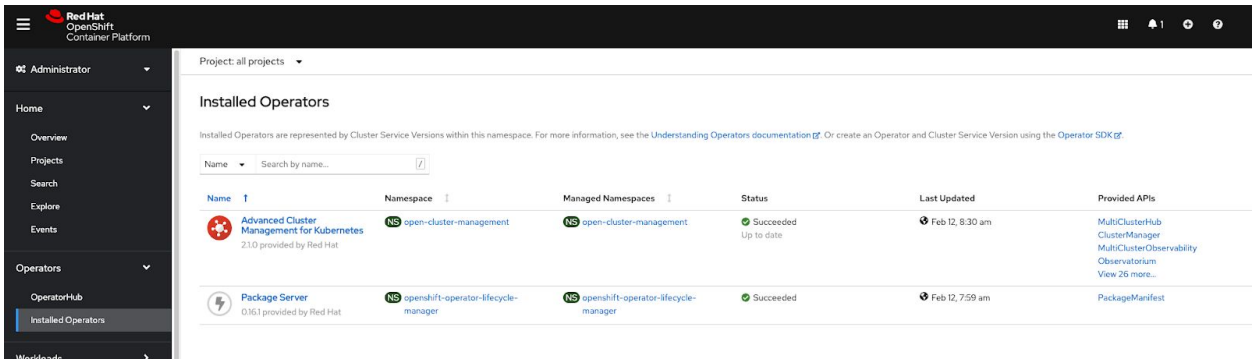
For example by clicking on one of the nodes, you get more information about this node. In this screenshot, I have clicked on ip-10-0-148-199.ap-southeast-1-compute.internal node. And on the right side, I can see all the information related to it.



Extra activity 5 - Check ACM deployment on the Hub cluster (local-cluster)

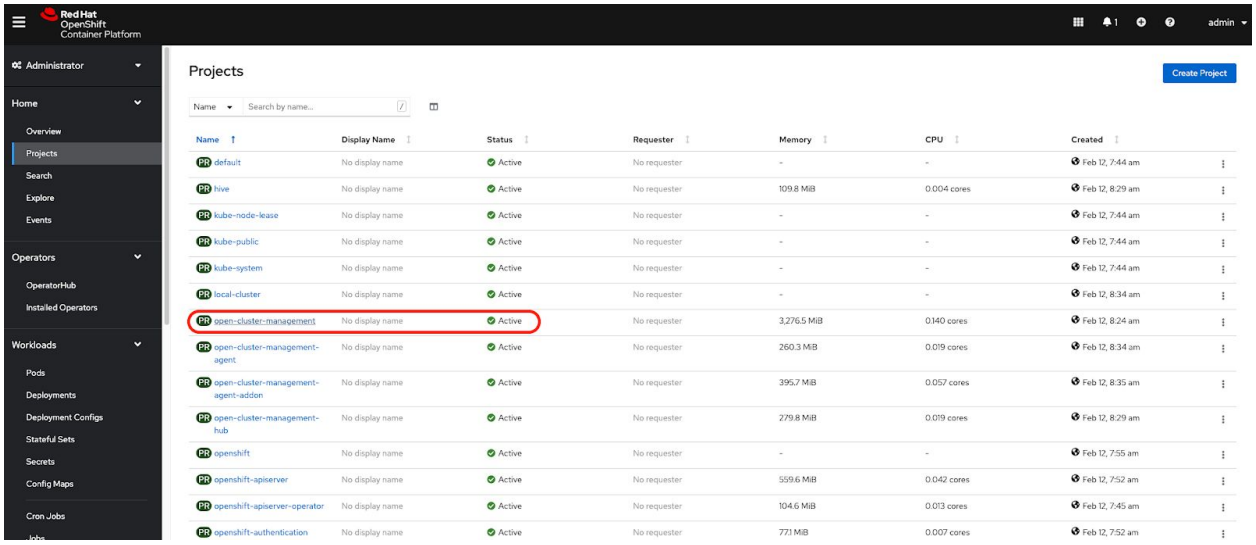
You can also check how ACM has been deployed on the OpenShift Hub cluster. For this, go back to the email from RHPDS and login to the OpenShift Console (you may have to reenter the username and password).

Once you are on the OpenShift console, go to Operators-> Installed Operators.

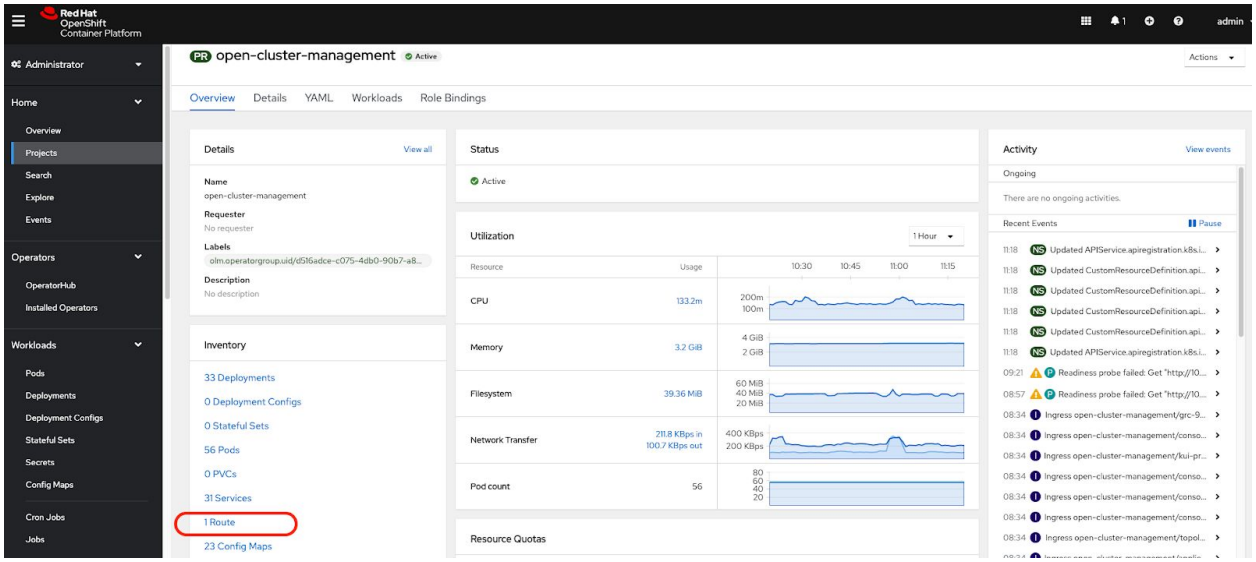


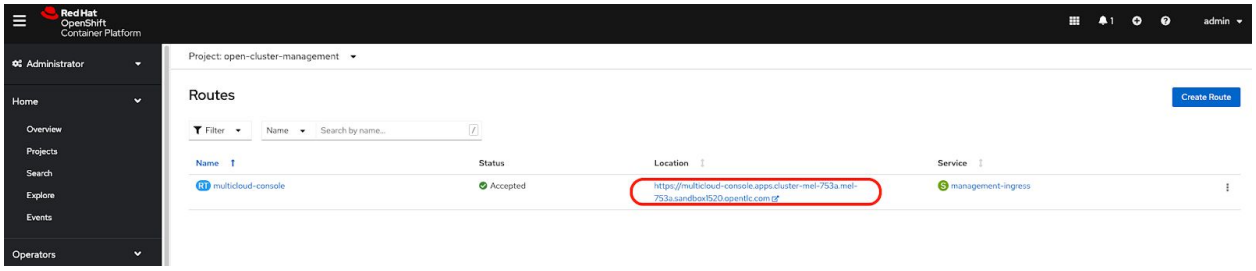
You can see that ACM is deployed in the open-cluster-management Namespace.

Click on the Projects (Home -> Projects Tab) and click on the open-cluster-management Project



This is where ACM has been deployed. Click on the 1 Route item and you will recognise the ACM Web console URL that was sent by email.

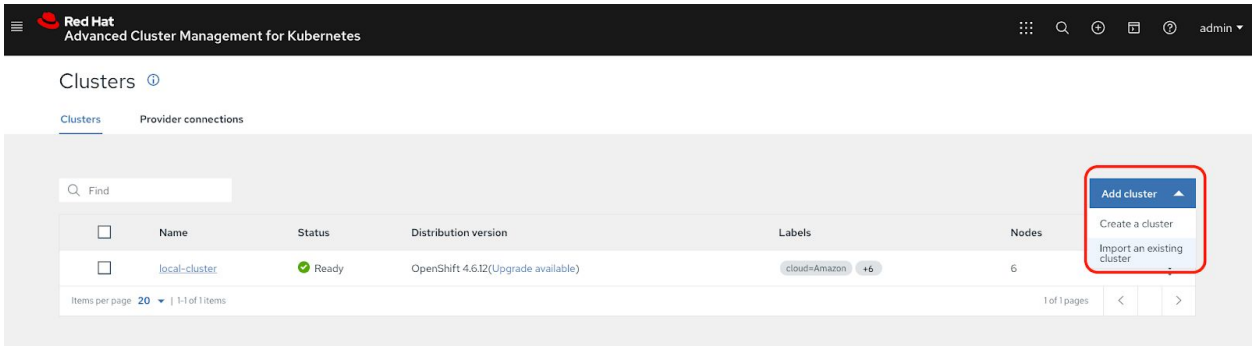




This is the end of the extra activities for the first exercise.
Feel free to go through the various components of the ACM Web console. More information is available under [ACM Web Console]

Second Exercise - Import a Cluster

Back to the Home Page in ACM.
Click on Cluster lifecycle (Go to Cluster).
Click on **Add Cluster --> Import an existing cluster**

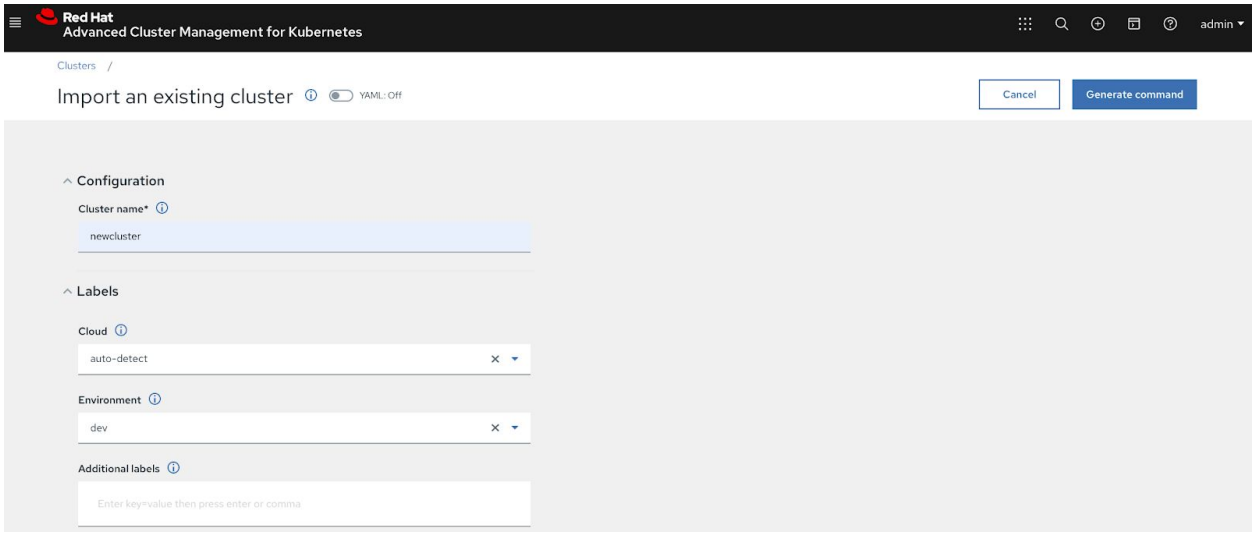


This will take you to the Import an existing cluster page.

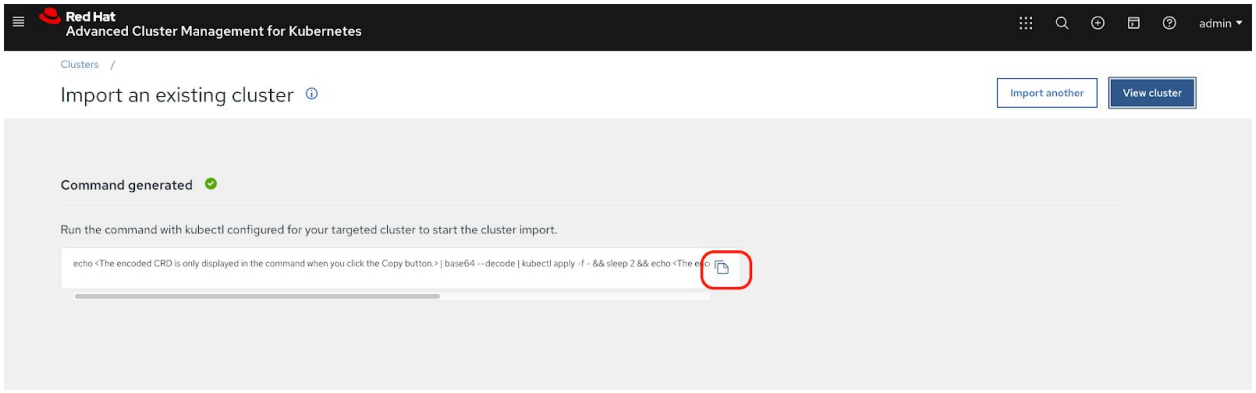
Fill the various fields on the page:

- Configuration -> Cluster name: Just specify any lower-case name. In the rest of this tutorial, we have used **newcluster** as the name (please note that the name you use for the cluster is not relevant).
- Labels:
    - Cloud: select auto-detect
    - Environment: select the **dev** label.
    - Additional labels: leave this field empty

We will be using those labels later on to provision applications.
Your screen should look like this



Once finished click **Generate command** and copy the kubectl command.



Using the credentials for the managed cluster, SSH to the Bastion host **on the managed cluster**. Make sure you use the Bastion host from the RHPDS OCP4 ACM Managed email (as per the screenshot below).

Thank you for ordering OCP4 ACM Managed
Your environment for RHPDS-RH-sdelord-redhat.com-PROD_OCP4_ACM_MANAGED_RHPDS-mel-2bda_COMPLETED has been provisioned.
Access to the environment will be granted as soon as the environment deployment is complete.

Please refer to the instructions for next steps and how to access your environment.

Troubleshooting and Access issues:

Always refer to the instructions to understand how to properly access and navigate your environment. If you need help using the SSH client on your computer you can consult http://www.opentlc.com/ssh.html.

NOTICE: Your environment will expire and be deleted in **7 day(s) at 2021-02-18 00:00:00 -0500.**

In order to conserve resources, we cannot archive or restore any data in this environment. All data will be lost upon expiration.

Here is some important information about your environment:

Openshift Master Console: https://console-openshift-console.apps.cluster-mel-2bda.mel-2bda.sandbox944.opentlc.com
Openshift API for command line 'oc' client: https://api.cluster-mel-2bda.mel-2bda.sandbox944.opentlc.com:6443
Download oc client from http://d3s3zqyaz8cp2d.cloudfront.net/pub/openshift-v4/clients/ocp/4.5.16/openshift-client-linux-4.5.16.tar.gz

HTPasswd Authentication is enabled on this cluster.
Users user1 .. user5 are created with password `openshift`
User `admin` with password `w9yllW0fwSuPi0WM` is cluster admin.
You can access your bastion via SSH:
ssh sdelord-redhat.com@bastion.mel-2bda.sandbox944.opentlc.com

Make sure you use the username 'sdelord-redhat.com' and the password 'nV3p8ASa9Pmb' when prompted.

Once you are logged in, double check that you are connected to the right cluster by running an oc whoami --show-server command and check that the name of the cluster matches the one in the email.

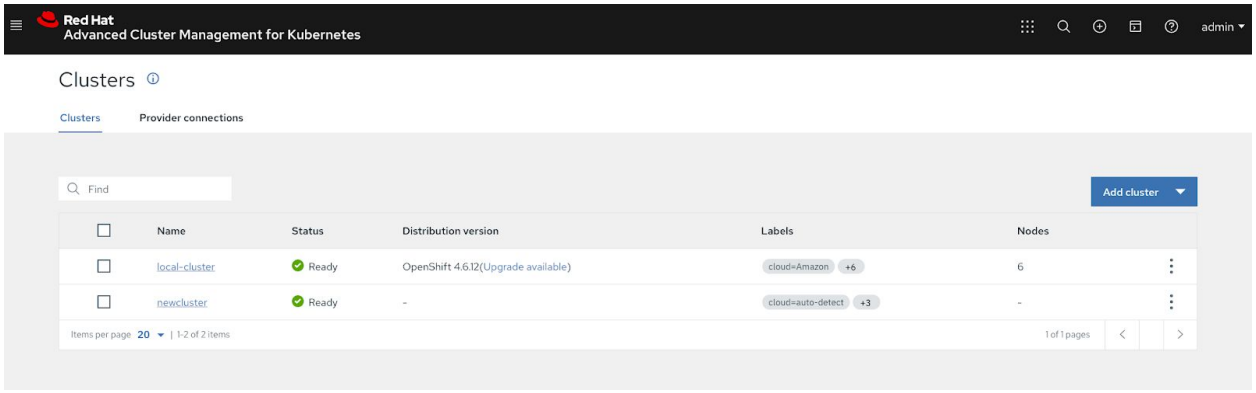
```
[sdelord-redhat.com@bastion ~]$ oc whoami --show-server
https://api.cluster-mel-2bda.mel-2bda.sandbox944.opentlc.com:6443
[sdelord-redhat.com@bastion ~]$
```

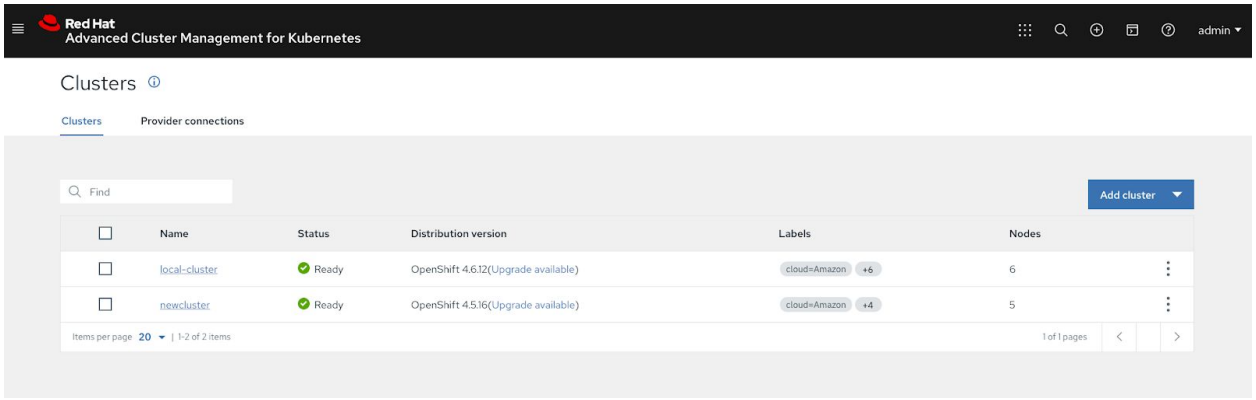
Paste the **Kubectl** command that you copied from ACM earlier in the bastion host, if all worked well you should see this on your terminal window

```
GVsYXlfZWNvbmRzOiAyCiAgICAgIHNlcnZpY2VBY2NvdW50YmFtZToga2x1c3Rlcmxldi0KLS0tCmFwaVZlcnNpb246IGV9wZXJhdG9yLm9wZW4tY2VtZW50
0LmlvL3YxCmtpbmQ6IEtsdXN0ZXJsZXQKbWV0YWRhdGE6CiAgbmFtZToga2x1c3Rlcmxldi0gzcGVjOgogIGNsdXN0ZXJOYW1lOiBuZXdjbHVzdGVyCiAgbmFtZXNwYWNlOiBvc
GVuLWNsdXN0ZXItbWFuYWdlbWVudC1hZ2VudAogIHJlZ2lzdHJhdGlvbkltYWdlUHVsbFNwZWM6IHJlZ2lzdHJ5LnJlZGhhdC5pby9yaGFjbTIvcmVnaXN0cmF0aW9uLXJoZWw
4QHNoYTI1NjpkZDNhNDBjYjFlNzI0NTE0YzhmOTA0NGE3NjQ4YTQzYTVmYTY4MWJlYmVjZGNkMWUyZDBlZWRhY2ZiZDZhYTYxCiAgd29ya0ltYWdlUHVsbFNwZWM6IHJlZ2lzd
HJ5LnJlZGhhdC5pby9yaGFjbTIvd29yay1yaGVsOEBzaGEyNTY6ZDg2Y2YzMjFiZjgwYjk1MmNmNDU2ZGFjMGY5ZWU0NDEwNDczMzc4ZjE3NTAzNjVjNzY1OWMwNjMxNDQ1ZTA
ZNAo= | base64 --decode | kubectl apply -f -
customresourcedefinition.apiextensions.k8s.io/klusterlets.operator.open-cluster-management.io created
namespace/open-cluster-management-agent created
serviceaccount/klusterlet created
secret/bootstrap-hub-kubeconfig created
clusterrole.rbac.authorization.k8s.io/klusterlet created
clusterrole.rbac.authorization.k8s.io/open-cluster-management:klusterlet-admin-aggregate-clusterrole created
clusterrolebinding.rbac.authorization.k8s.io/klusterlet created
deployment.apps/klusterlet created
klusterlet.operator.open-cluster-management.io/klusterlet created
[sdelord-redhat.com@bastion ~]$
```

On the ACM console, navigate back to the Clusters view. And wait for the cluster to become available (should be no more than 5 min).

You will first see the cluster newcluster becoming Ready as a Status and then after a short wait, all the rest of the information will be displayed (Distribution version, Nodes).

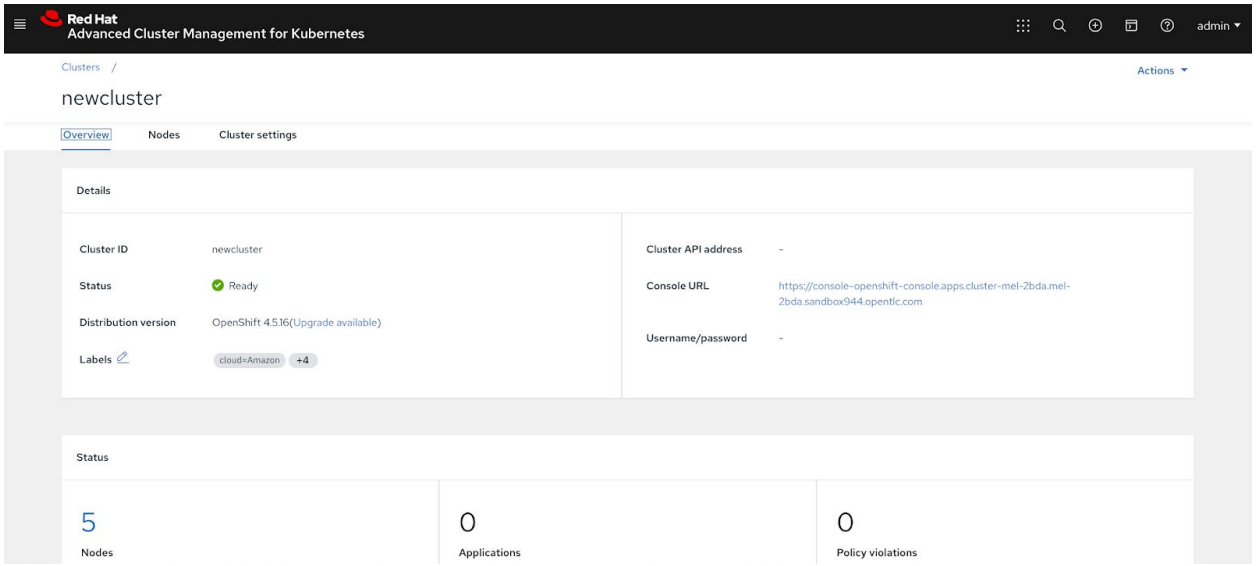




Extra activities:

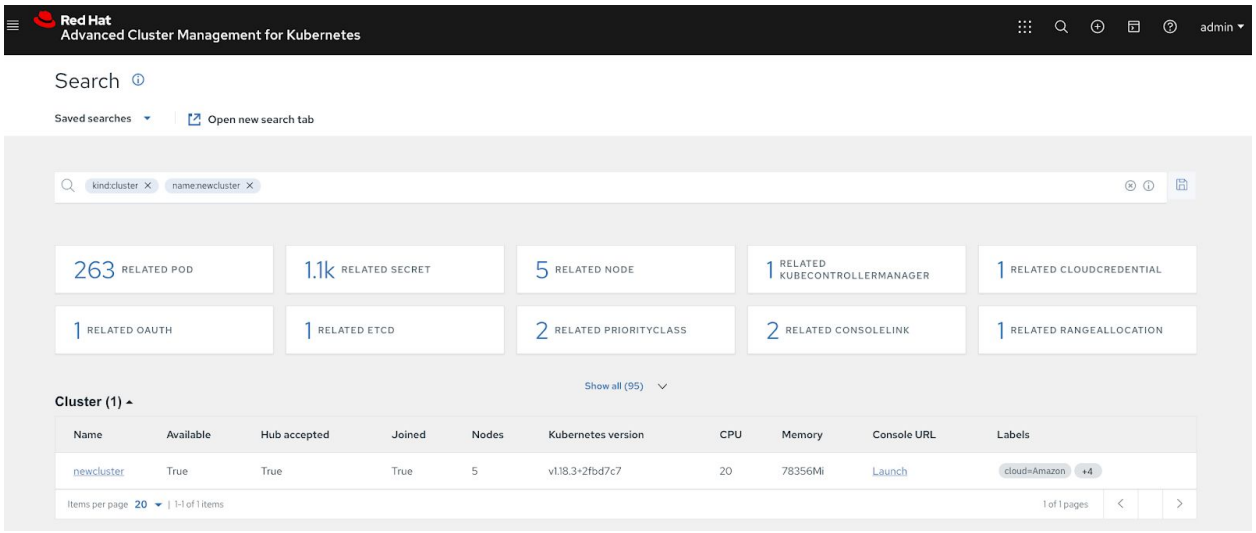
Extra activity 1- Discover the local-cluster

Similarly as Extra activity 1 in Exercise 1, explore the various tabs under newcluster (Overview, Nodes, Cluster settings).



Extra activity 2 - Perform a search

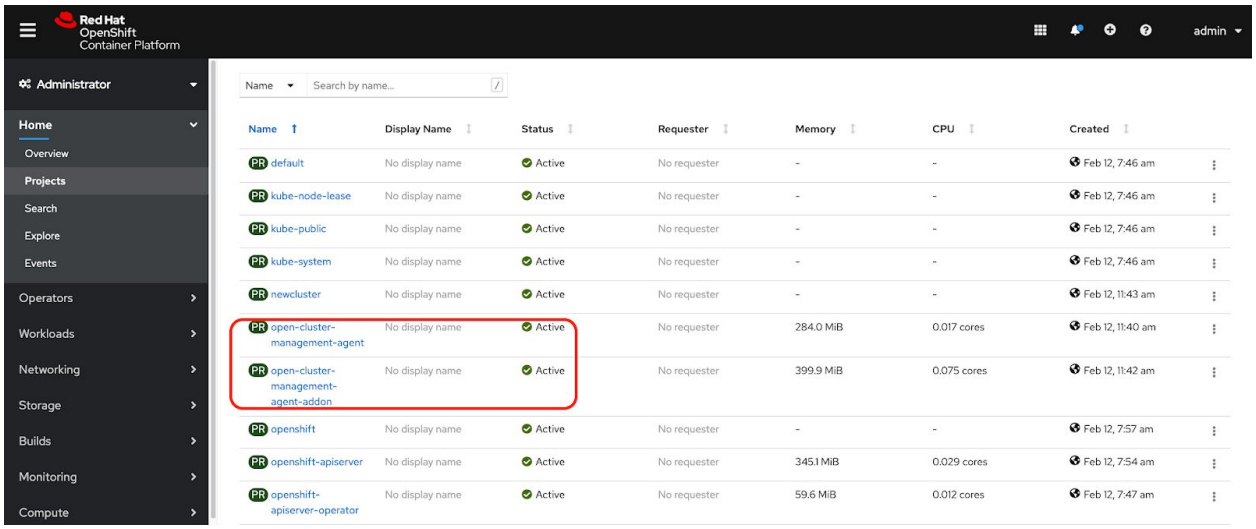
Go to the Search Icon on the top right of ACM and issue a search for kind:cluster & name:newcluster.



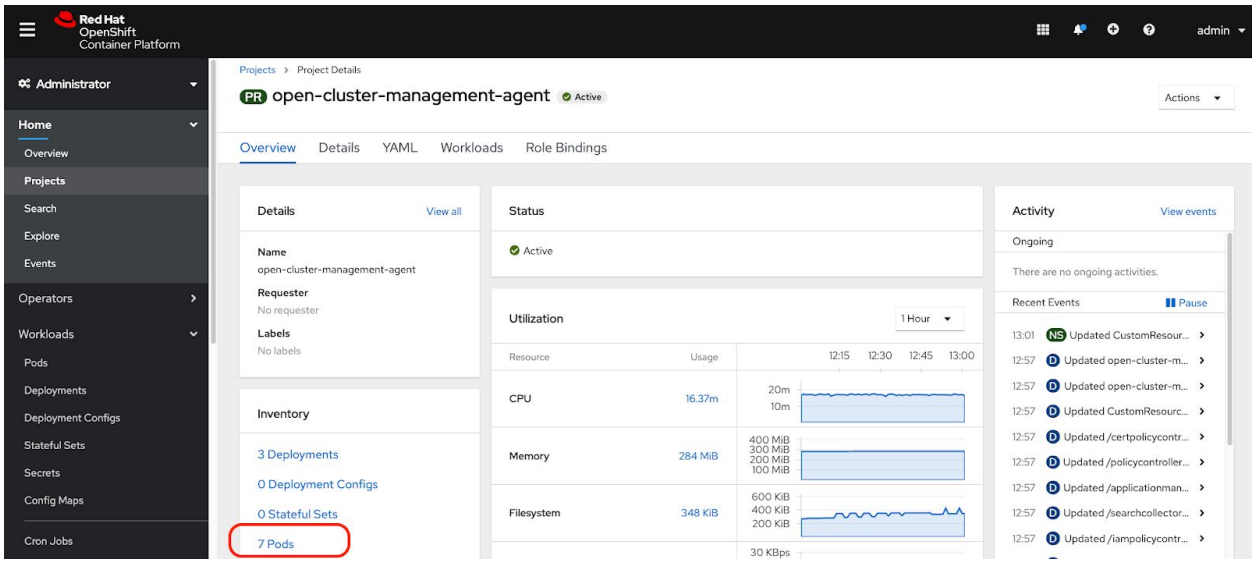
Extra activity 3 - Check the Klusterlet Agent on the managed cluster (newcluster)

You can also check how the various parts for the Klusterlet and associated functions have been deployed on the managed cluster. For this, go back to the email from RHPDS for the OCP4 ACM Managed request and login to the OpenShift Console.

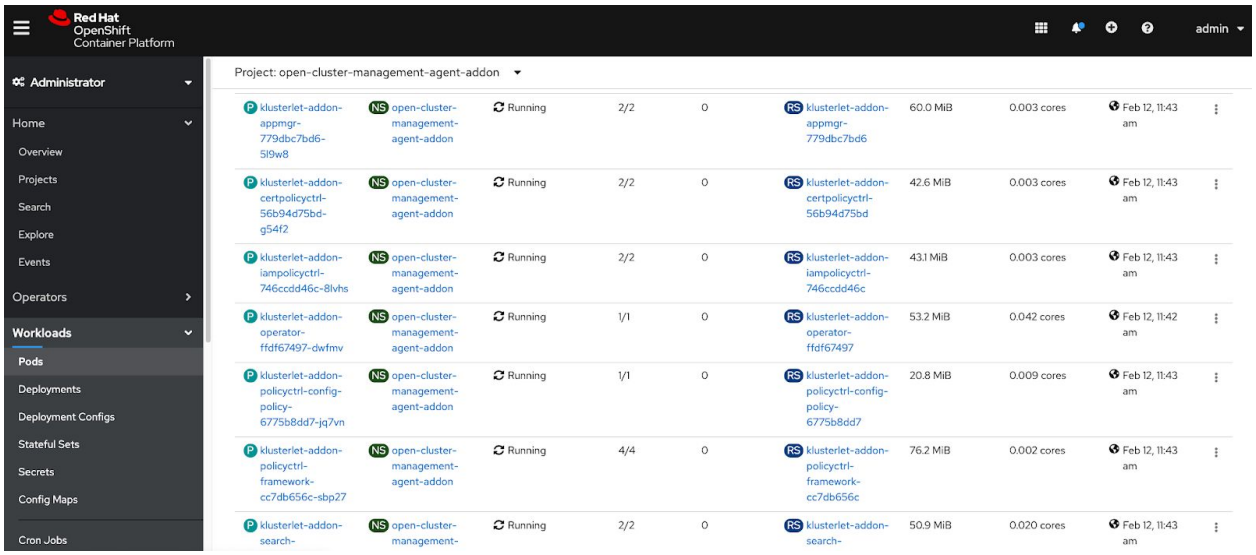
Click on the Projects (Home -> Projects Tab) and click on the open-cluster-management-agent Project



The open-cluster-management-agent project is where the klusterlet agent gets deployed to connect back to the Hub cluster. Click on the Pods and see the various Pods deployed in there.



Check the open-cluster-management-addon project and see the various Pods used for applications, adds-on for policy, applications)

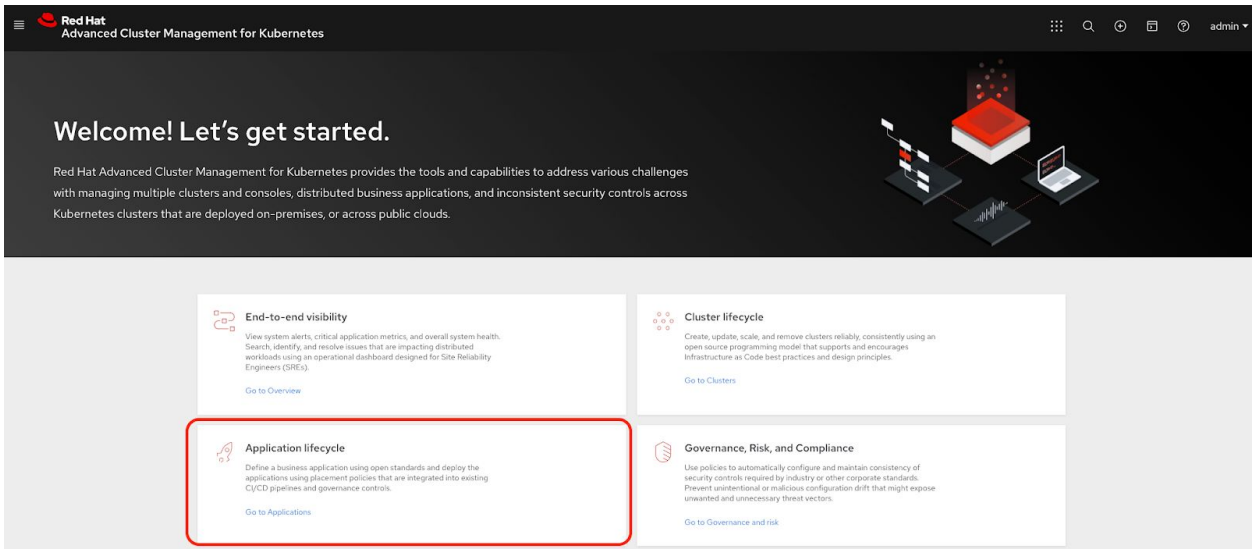


Exercise 2: Application Lifecycle

In the previous exercise, you explored the Cluster Lifecycle functionality in RHACM. This allowed you to import an OpenShift® cluster to RHACM, which you can now use to deploy applications.

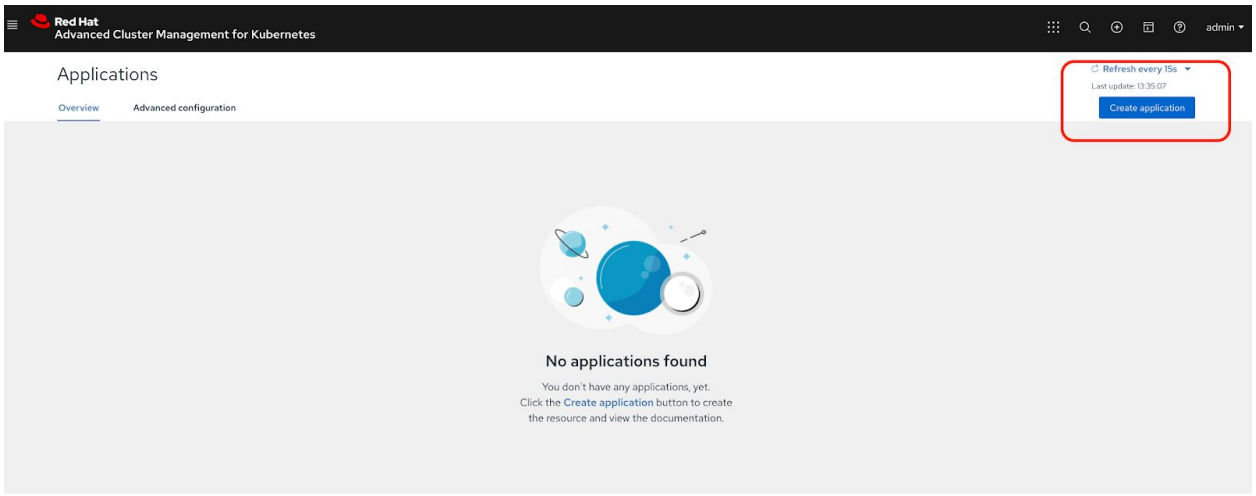
Application Lifecycle functionality in RHACM provides the processes that are used to manage application resources on your managed clusters.

Go back to the Home screen of ACM and click on the Application Lifecycle (Go to Applications) box.



Creating an Application

Within the Applications window click on **Create application**.

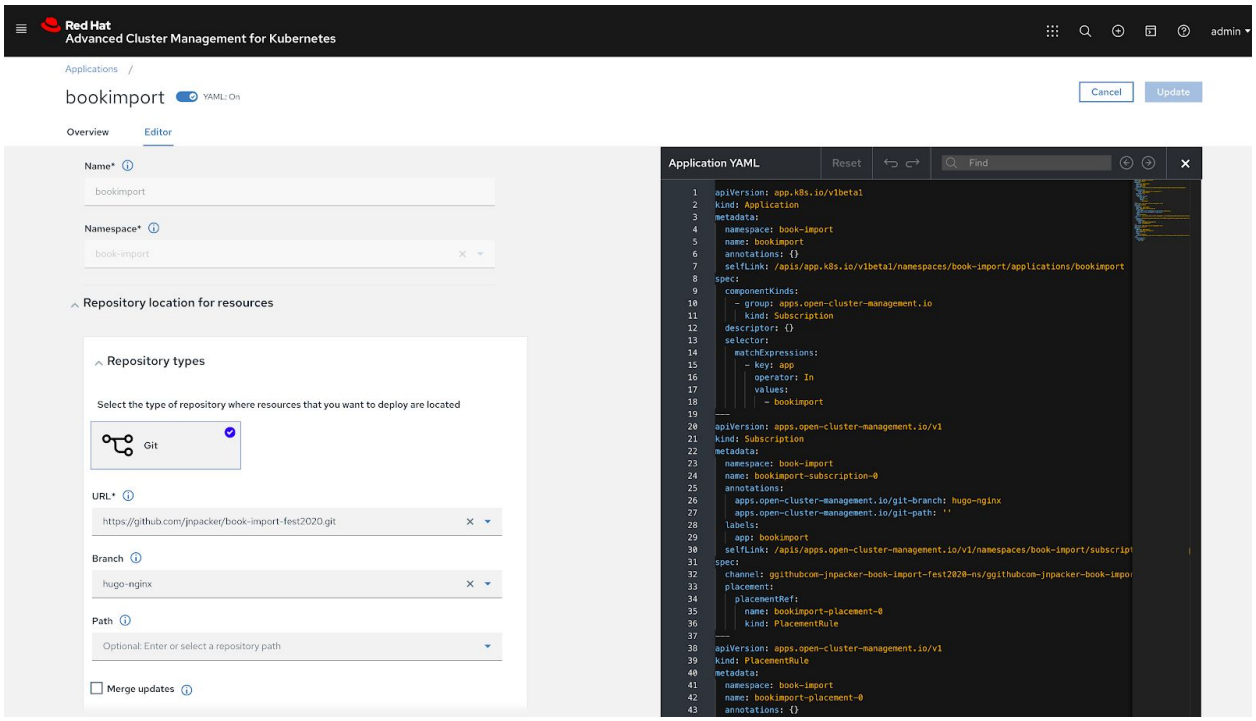


Enter the following information:

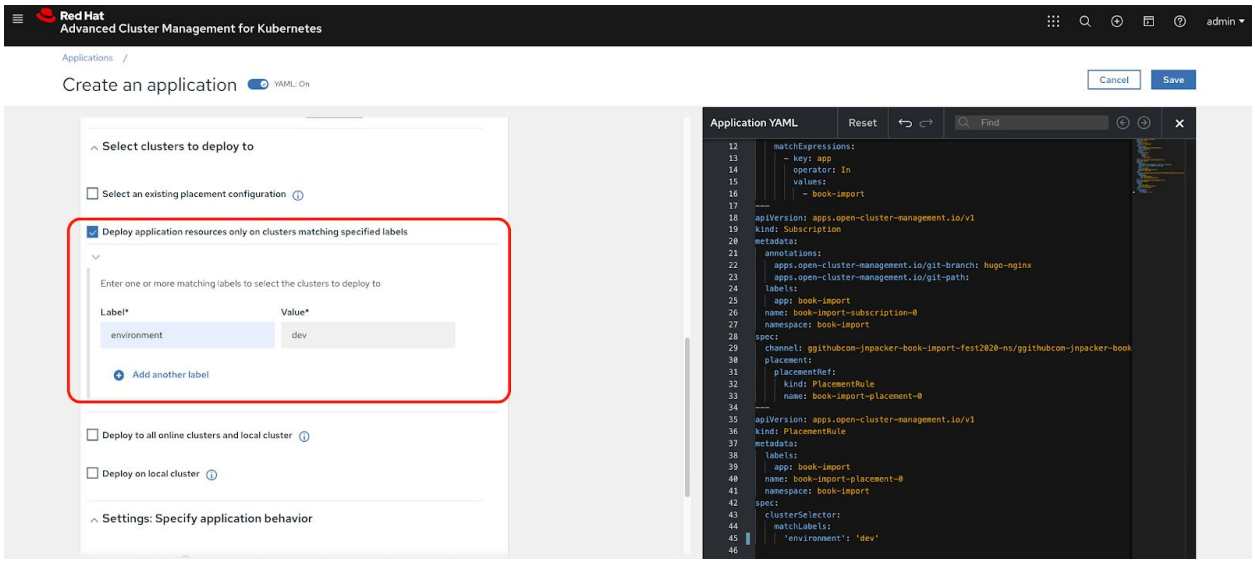
Name: book-import
Namespace: book-import

Under repository types, select the GIT repository

URL: https://github.com/jnpacker/book-import-fest2020.git
Branch: hugo-nginx

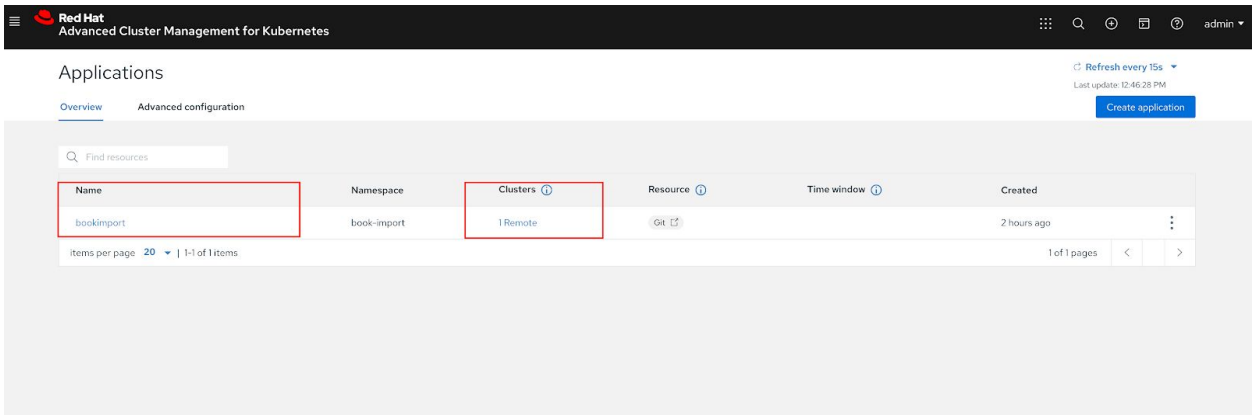


Select Deploy application resources only on clusters matching specified labels
Label: environment
Value: dev



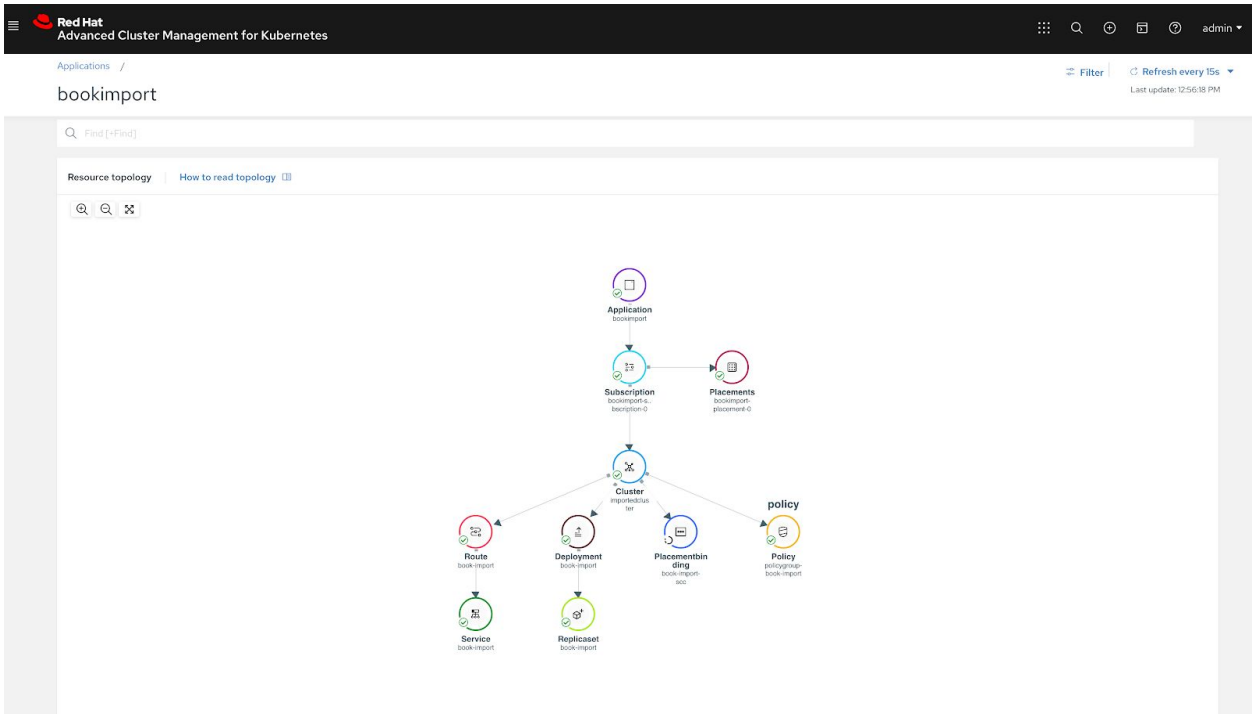
Click **Save**

After a few minutes you will see the application available in ACM Applications Window.

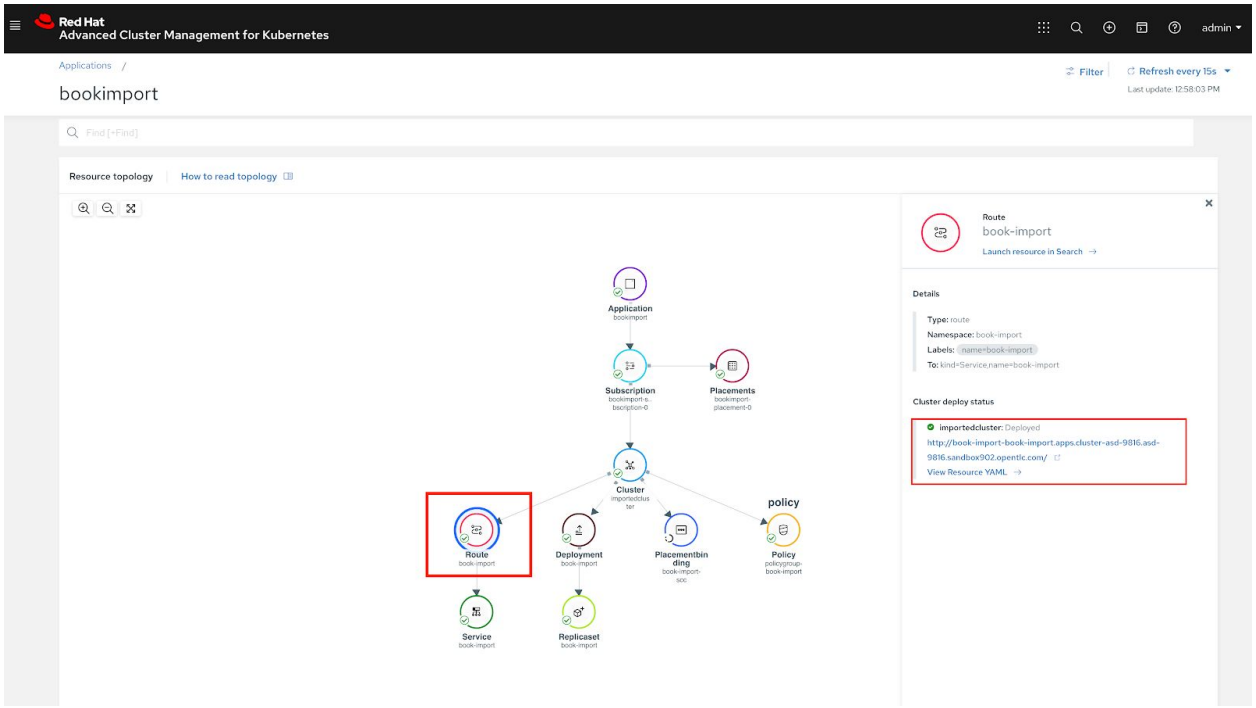


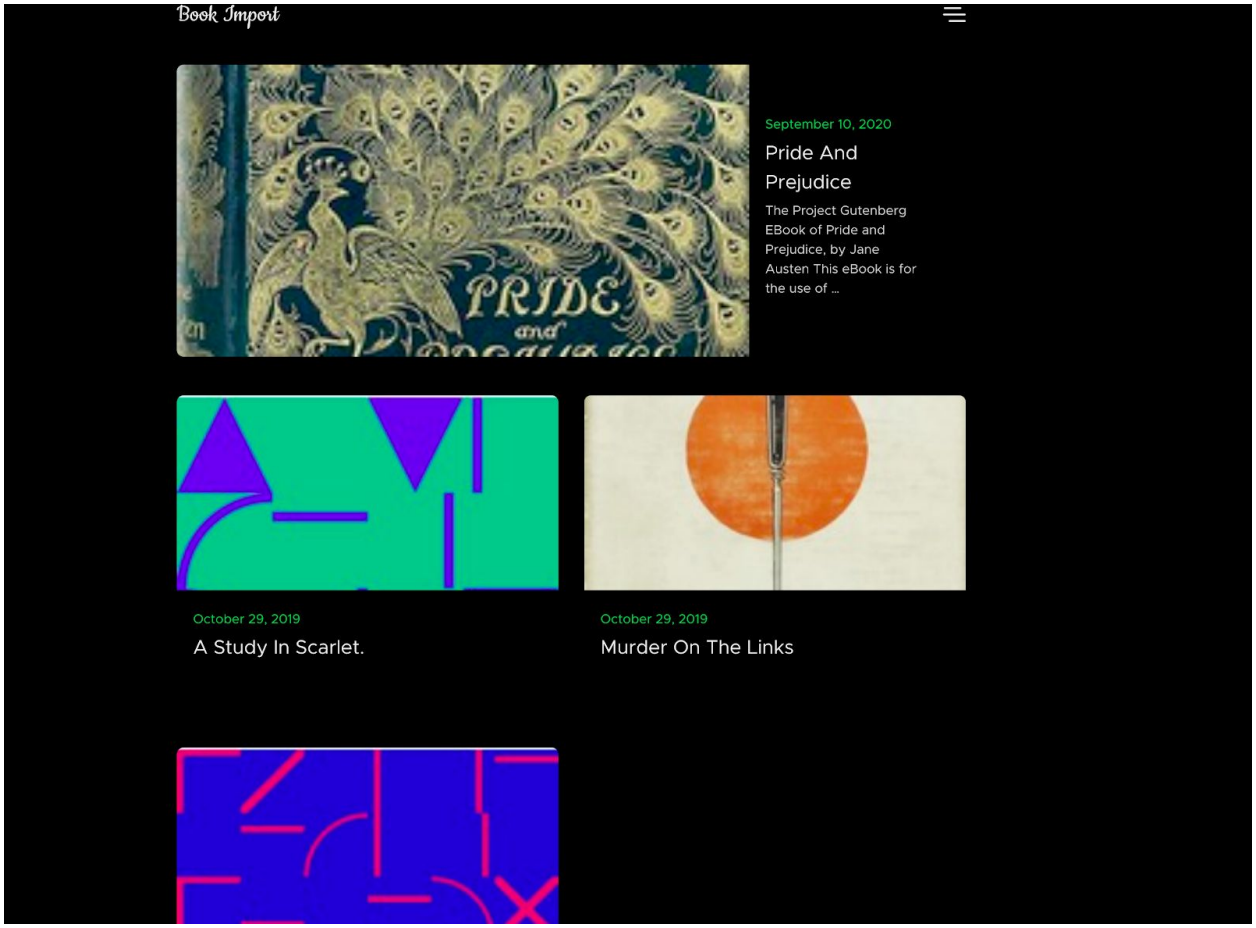
If everything was done correctly you should be able to see the application deployed to 1 Remote cluster.

Click the **book-import** application and have a look at the topology view



Select the Route and click on the URL provided, you should see the Book Import application





You have now completed the overview of the **Application Lifecycle functionality in RHACM.**

You successfully deployed an application to a target cluster using RHACM.

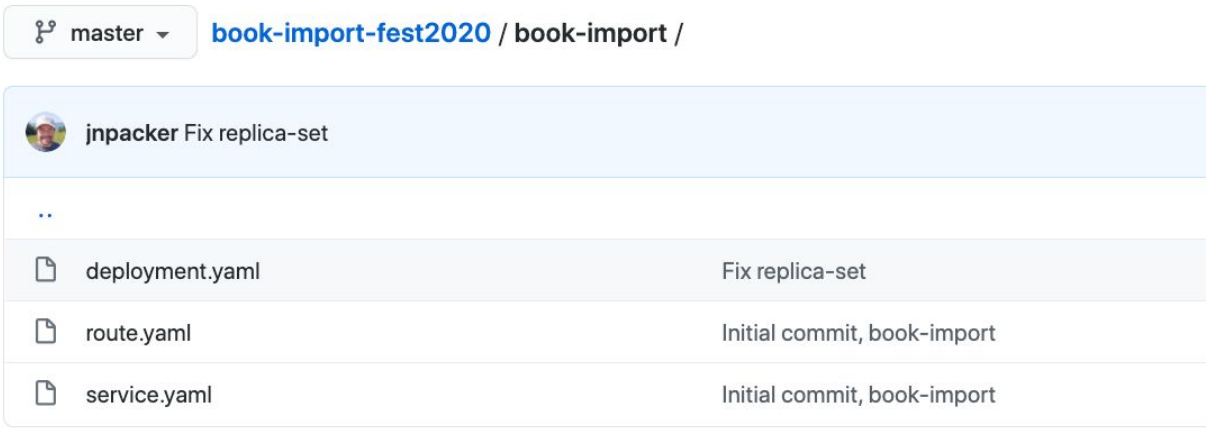
This approach leveraged a Git repository which housed all of the manifests that defined your application. RHACM was able to take those manifests and use them as deployables, which were then deployed to the target cluster.

You also leverage the power of labels and deploy the application to your imported cluster.

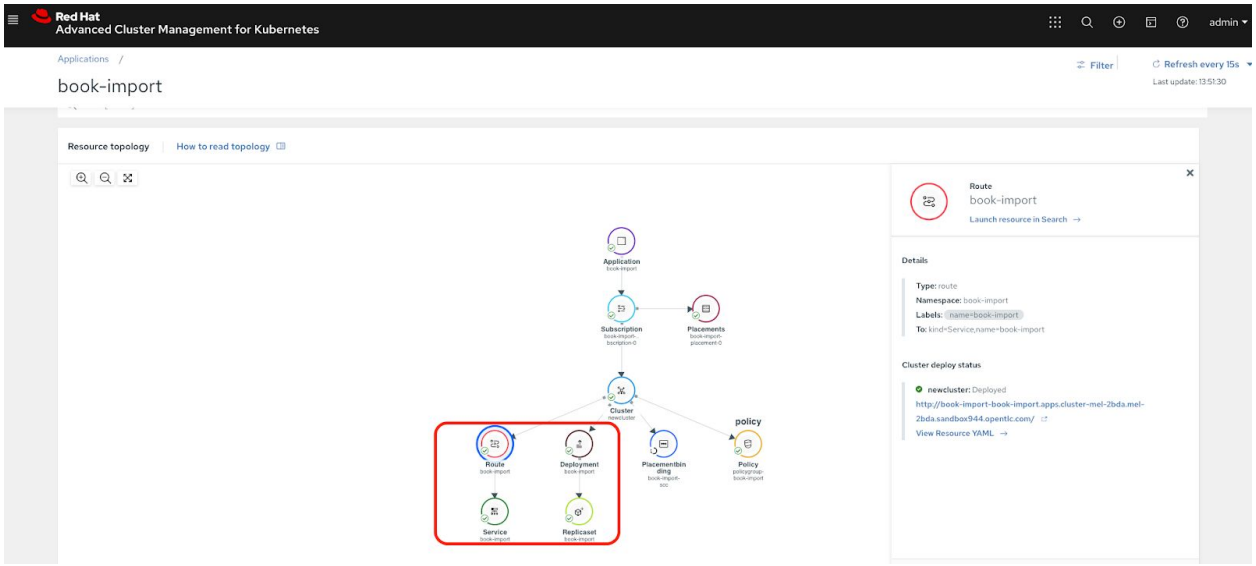
Extra activities:

Extra activity 1 - Compare the Git Repository and the ACM Display

Open a new tab in your browser and go to
https://github.com/jnpacker/book-import-fest2020/tree/master/book-import

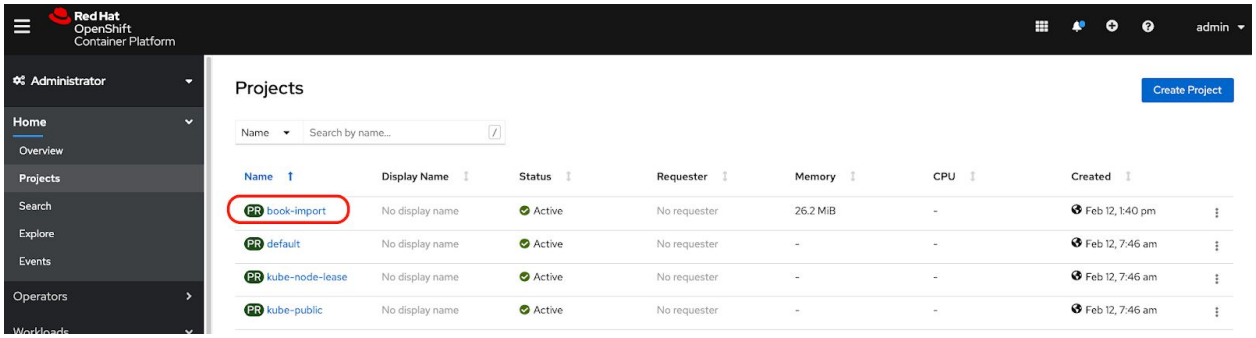


You should be able to see that the Git Repo resources have been represented as part of the book-info deployed App (the Replicaset being part of the deployment.yaml).

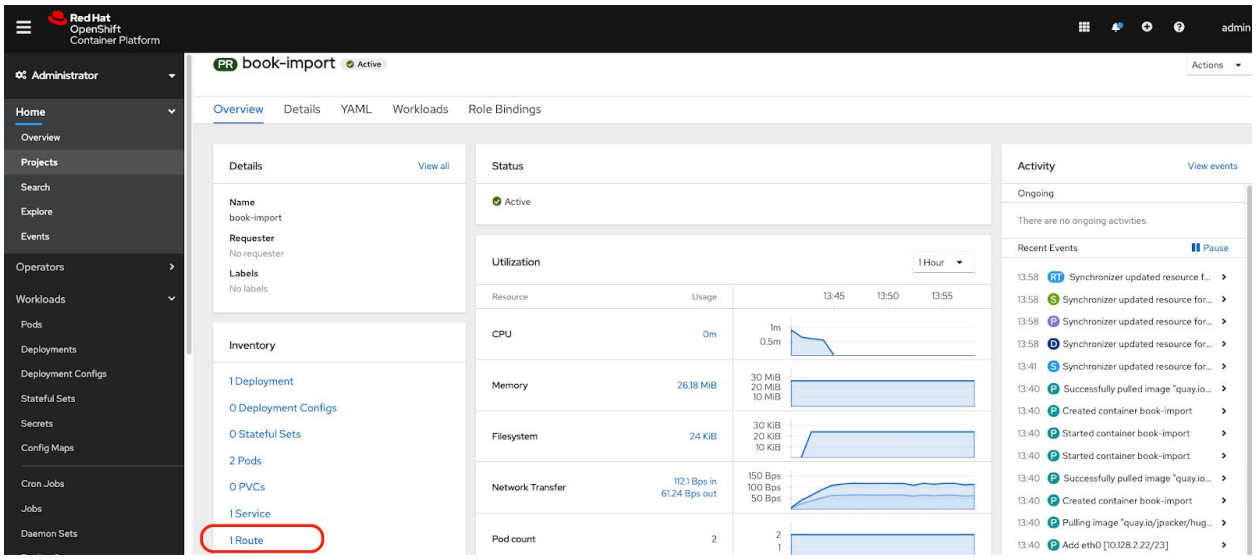


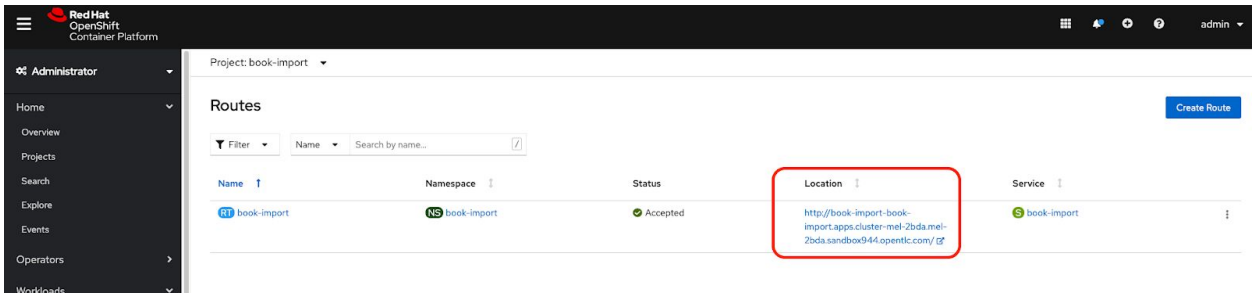
Extra activity 2 - Compare the OpenShift newcluster and the ACM Display

Log into the OCP managed cluster (newcluster) using the details from the RHPDS email. Under the Home->Projects tab check the book-import project.



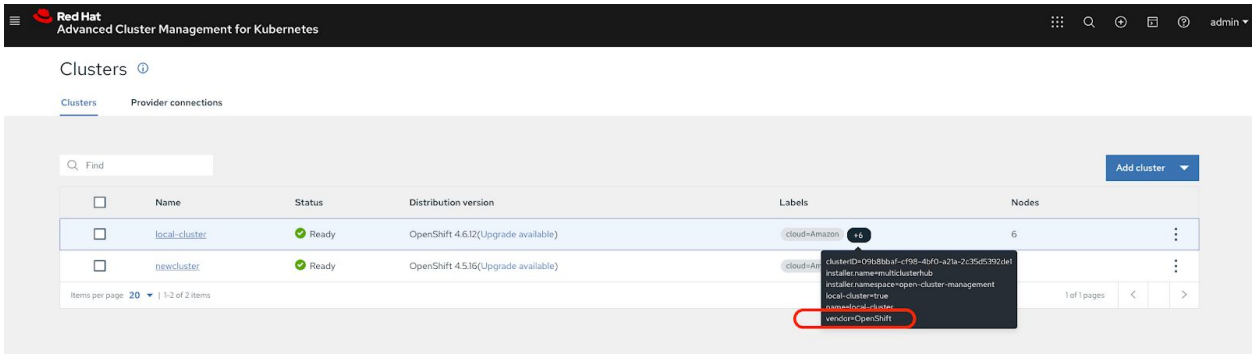
Check the 1 Route and see if it matches the ACM Route display in the Applications book-import window.



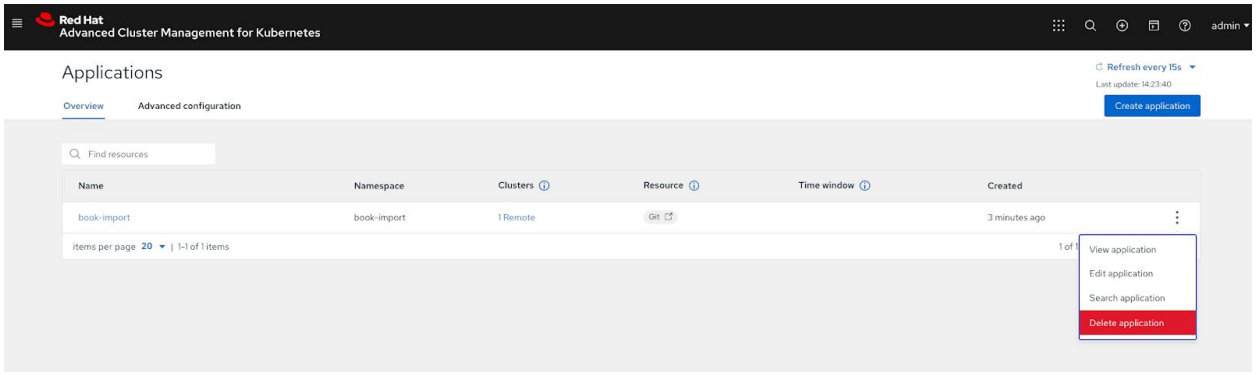


Extra activity 3 - Deploy the book-import application to the local-cluster

Go back to the Clusters view. By browsing over the Labels column, check that both the local-cluster and newcluster have the following label **vendor=OpenShift**



We will now go back to the Applications view. On the book-import line, click on the 3 grey dots and select Delete application.

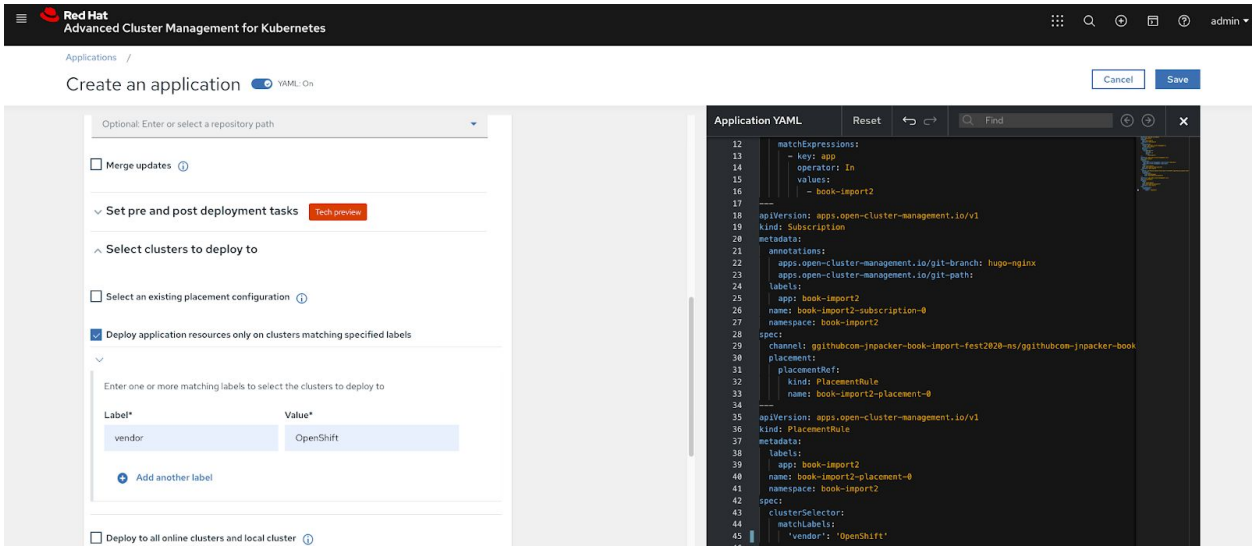


We will now Create a new application (reusing the same Git Repo) with a different label.

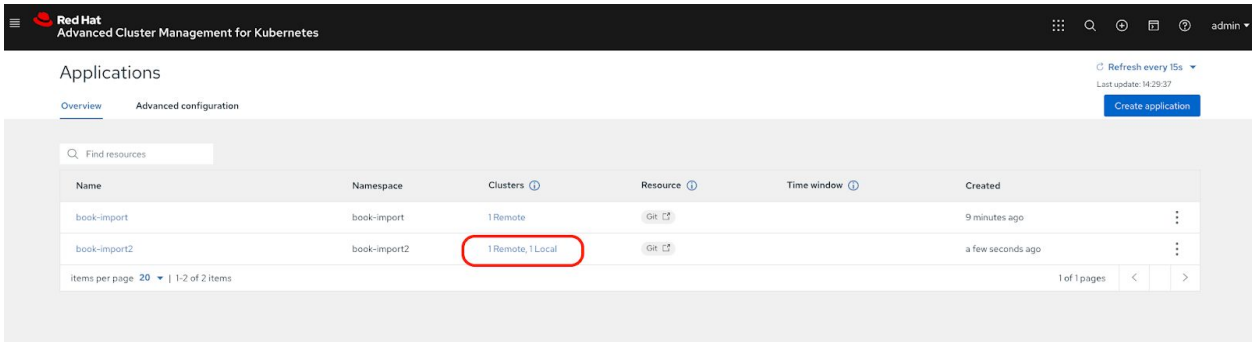
Click on the Create application button.
Use the following values for the different fields:

- Name: book-import2
- Namespace: book-import2
- Repository Type: Git
    - URL: https://github.com/jnpacker/book-import-fest2020.git
    - Branch: hugo-nginx
- Label: vendor
- Value: OpenShift (be careful this is case sensitive)

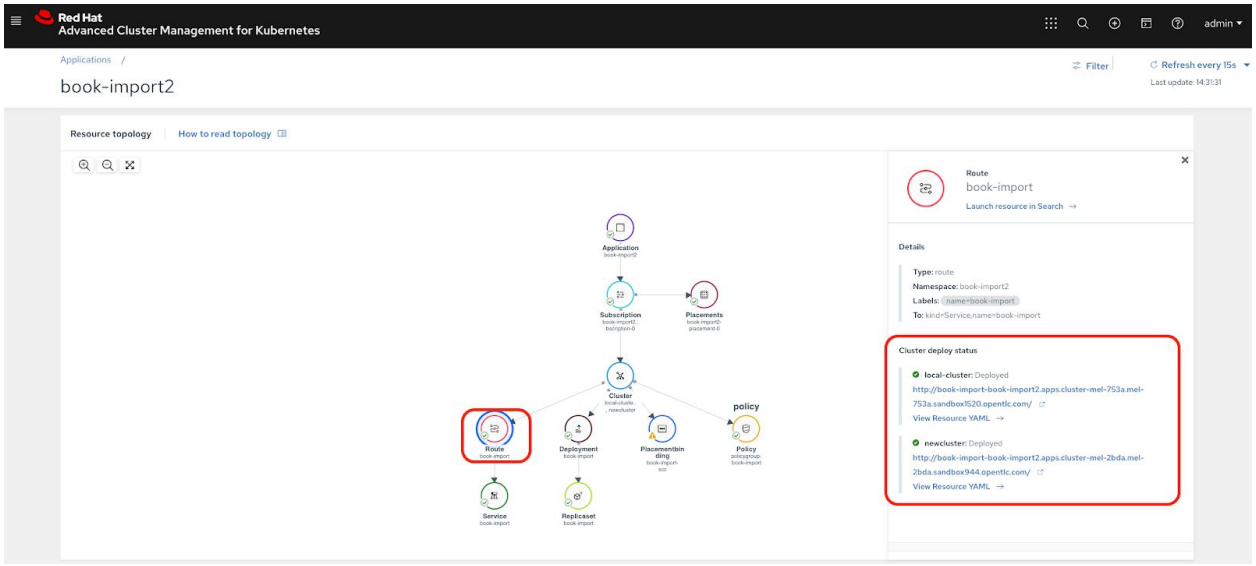
Then click Save.



Going back to the Applications window, you will see after a bit of time that this new application (book-import2) is deployed on both the local-cluster and the newcluster.



Click on book-import2, and you will see that there are now 2 clusters where this application is deployed.
Click the Route icon, and you will see that there 2 entries to which you can now navigate to access the Application.

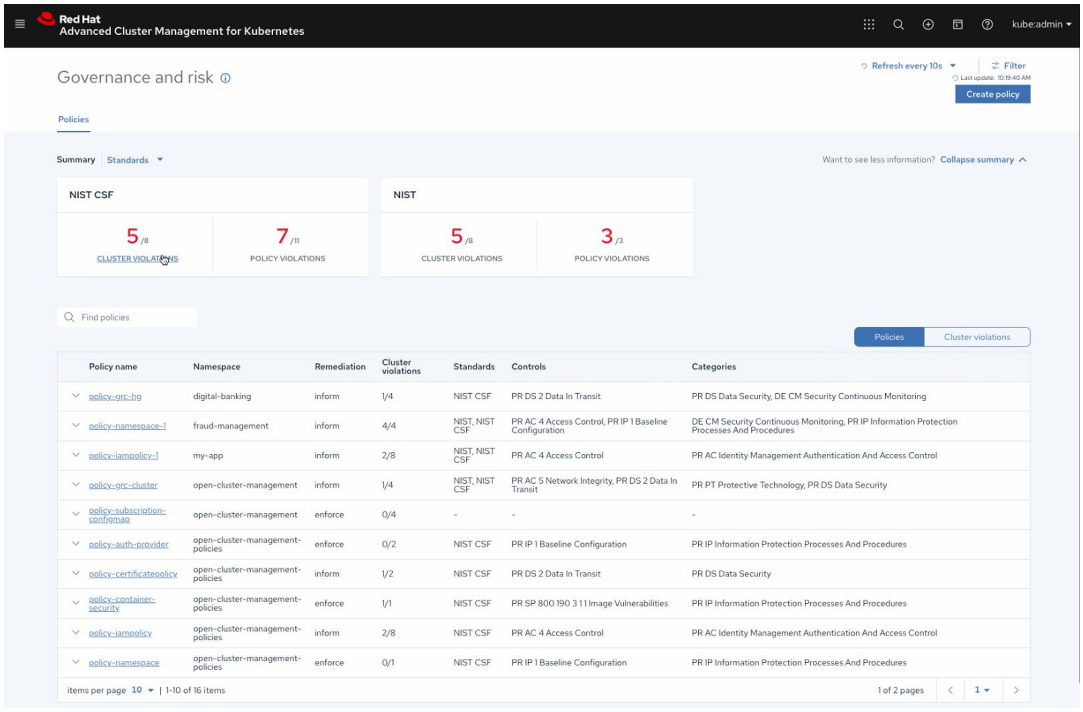

Red Hat
Advanced Cluster Management for Kubernetes
admin
Applications /
book-import2
Filter
Refresh every 15s
Last update: 14:31:31
Resource topology
How to read topology
Application
Subscription
Placements
Cluster
Route
Deployment
Placementbin
ding
policy
Policy
Service
Replicaset
Route
book-import
Launch resource in Search
Details
Type: route
Namespace: book-import2
Labels: name=book-import
To: kind=Service,name=book-import
Cluster deploy status
local-cluster: Deployed
http://book-import-book-import2.apps.cluster-mel-753a.mel-753a.sandbox1520.opentlc.com/
View Resource YAML
newcluster: Deployed
http://book-import-book-import2.apps.cluster-mel-2bda.mel-2bda.sandbox944.opentlc.com/
View Resource YAML

Exercise 3: Creating Policies in ACM

At this point, you have completed the overview labs for Cluster Lifecycle and Application Lifecycle capabilities in RHACM.

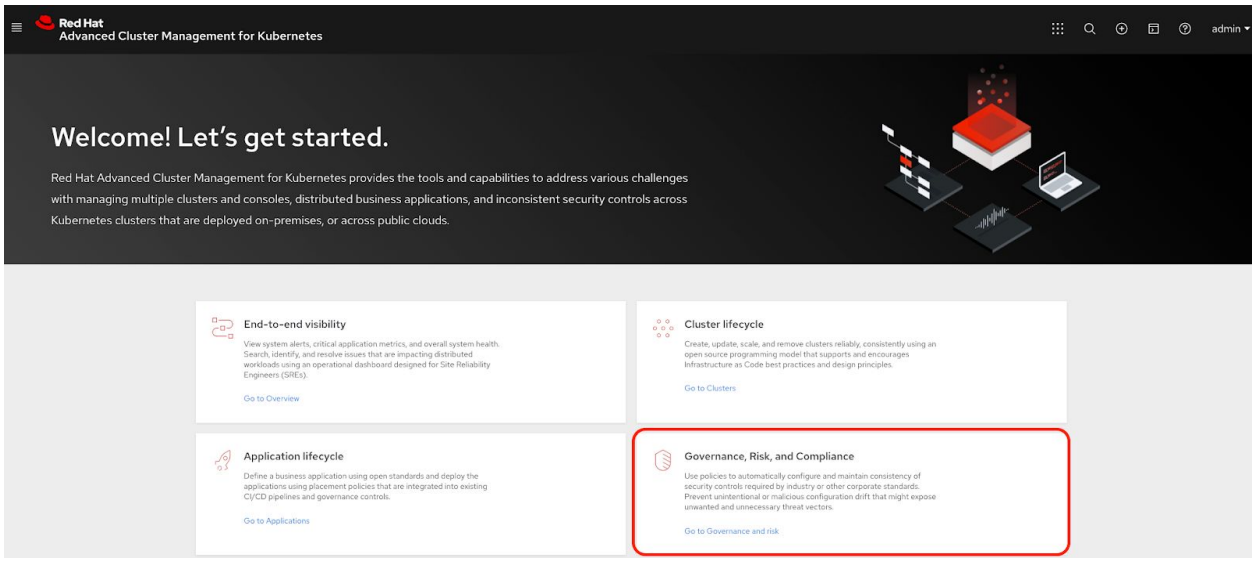
Now that you have a cluster and a deployed application, you need to make sure that they do not drift from their original configurations. This kind of drift is a serious problem, because it can happen from benign and benevolent fixes and changes, as well as malicious activities that you might not notice but can cause significant problems.
The solution that RHACM provides for this is the Governance, Risk, and Compliance, or GRC, functionality.
Review GRC Functionality

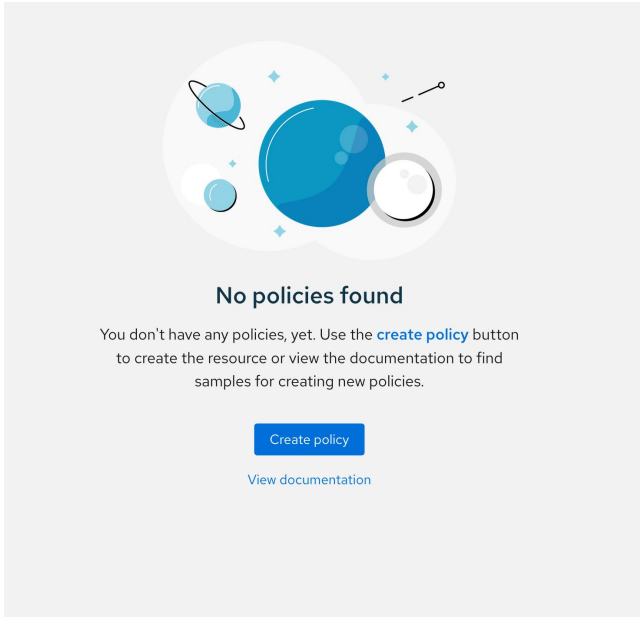
The RHACM console provides an easy way to start creating basic policies, as shown below. In this example, you can see that when you change values for elements in the RHACM console, the YAML content is updated.



This is a complex topic, and this course is only providing an overview. Please consult the GRC product documentation for more details on any of these policy controllers.

From the Home page, navigate to the **Governance, Risk and Compliance** screen by clicking on Go to Governance and risk.

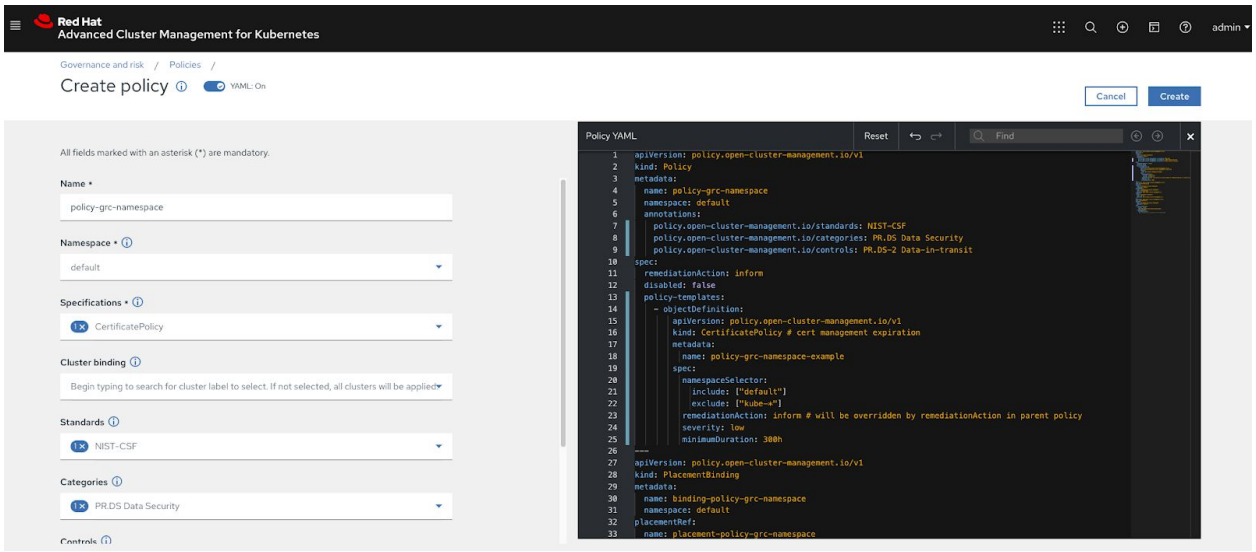


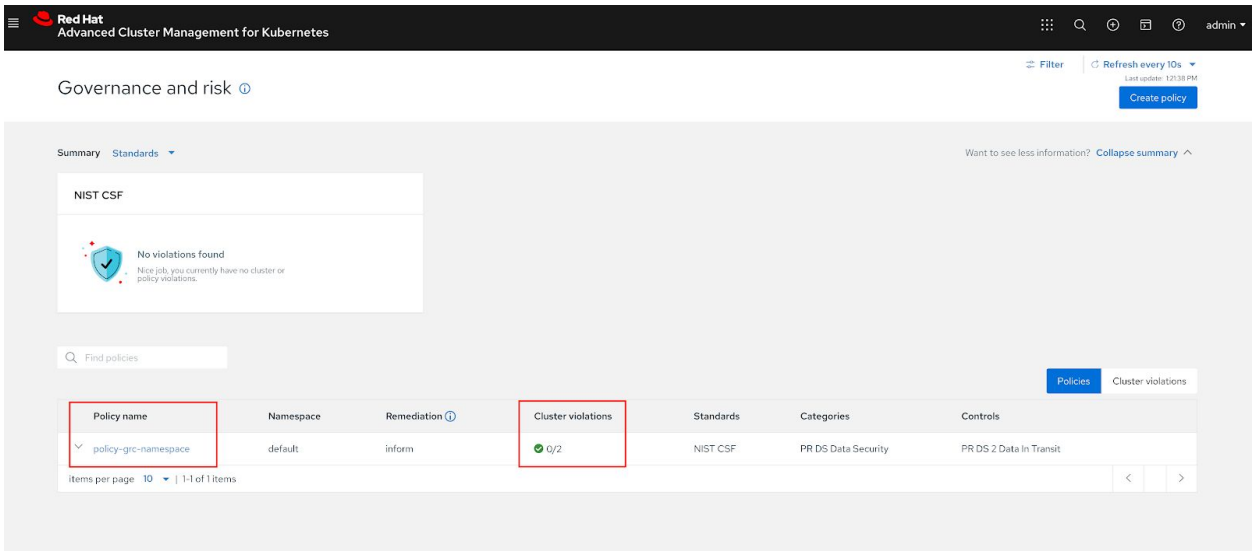


Build a policy by clicking on Create policy and provide the following information:

- **Name**: policy-grc-namespace
- **Namespace**: default
- **Specifications**: CertificatePolicy -cert management expiration

Leave everything else as default and click **Create**. Please note that initially it will complain that there is an issue with the policy but shortly after should go green and get a checkmark





You can also play around by creating more policies if needed, we have some examples in the [GitPolicy repo](#) .

There is also a smaller subject of these policies I normally use for demos all located [here](#)