NetApp Astra Control Center Overview

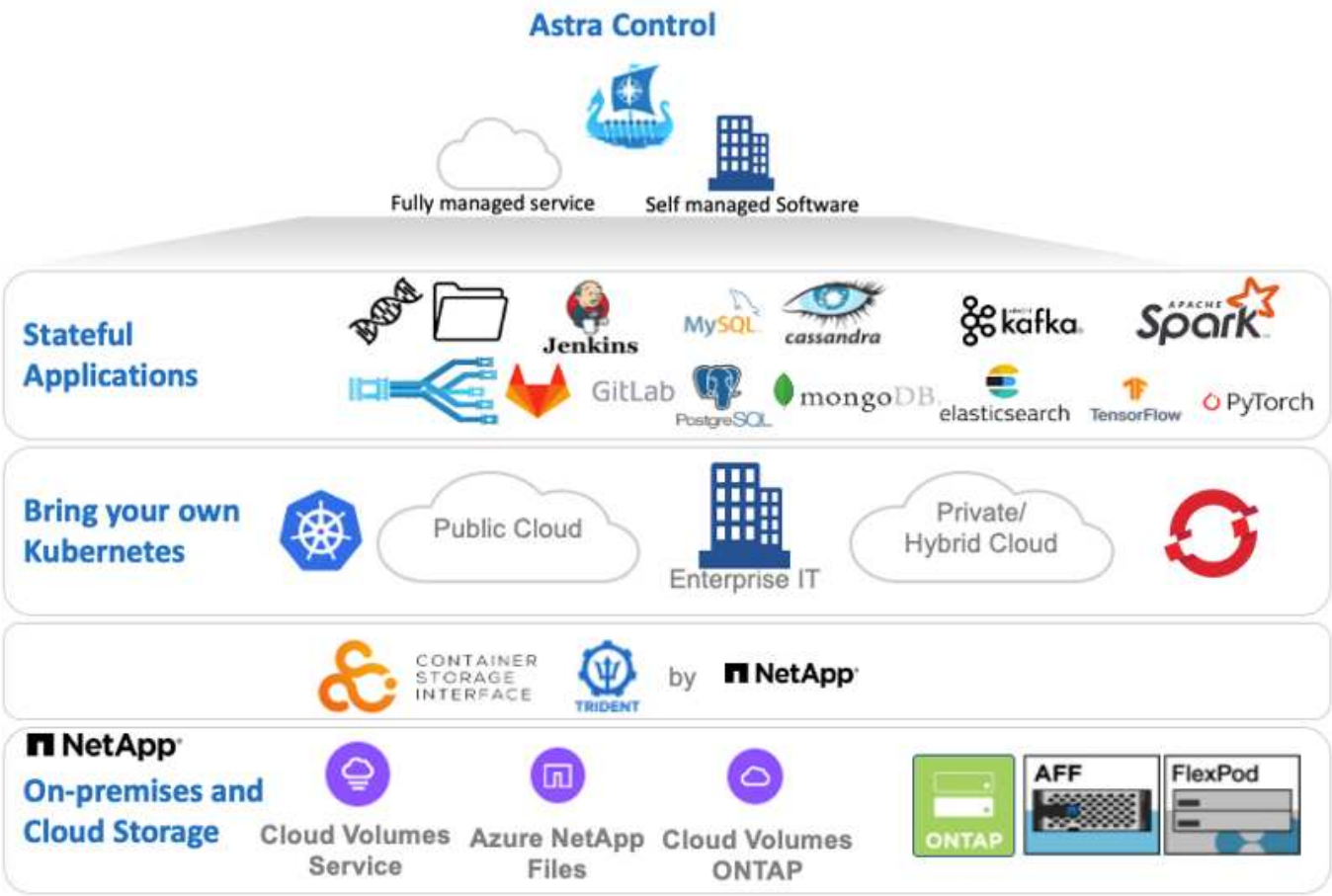
NetApp Solutions

NetApp
January 18, 2022

Table of Contents

NetApp Astra Control Center overview

NetApp Astra Control Center offers a rich set of storage and application-aware data management services for stateful Kubernetes workloads deployed in an on-premises environment and powered by NetApp data protection technology.



NetApp Astra Control Center can be installed on a Red Hat OpenShift cluster that has the Astra Trident storage orchestrator deployed and configured with storage classes and storage backends to NetApp ONTAP storage systems.

For the installation and configuration of Astra Trident to support Astra Control Center, see this document here.

In a cloud-connected environment, Astra Control Center uses Cloud Insights to provide advanced monitoring and telemetry. In the absence of a Cloud Insights connection, limited monitoring and telemetry (7-days worth of metrics) is available and exported to Kubernetes native monitoring tools (Prometheus and Grafana) through open metrics endpoints.

Astra Control Center is fully integrated into the NetApp AutoSupport and Active IQ ecosystem to provide support for users, provide assistance with troubleshooting, and display usage statistics.

In addition to the paid version of Astra Control Center, a 90-day evaluation license is available. The evaluation version is supported through the email and community (Slack channel). Customers have access to these and other knowledge-base articles and the documentation available from the in-product support dashboard.

To get started with NetApp Astra Control Center, visit the Astra website.

Astra Control Center installation prerequisites

1. One or more Red Hat OpenShift clusters. Versions 4.6 EUS and 4.7 are currently supported.
2. Astra Trident must already be installed and configured on each Red Hat OpenShift cluster.
3. One or more NetApp ONTAP storage systems running ONTAP 9.5 or greater.

   > It's best practice for each OpenShift install at a site to have a dedicated SVM for persistent storage. Multi-site deployments require additional storage systems.

4. A Trident storage backend must be configured on each OpenShift cluster with an SVM backed by an ONTAP cluster.
5. A default StorageClass configured on each OpenShift cluster with Astra Trident as the storage provisioner.
6. A load balancer must be installed and configured on each OpenShift cluster for load balancing and exposing OpenShift Services.

   > See the link here for information about load balancers that have been validated for this purpose.

7. A private image registry must be configured to host the NetApp Astra Control Center images.

   > See the link here to install and configure an OpenShift private registry for this purpose.

8. You must have Cluster Admin access to the Red Hat OpenShift cluster.
9. You must have Admin access to NetApp ONTAP clusters.
10. An admin workstation with docker or podman, tridentctl, and oc or kubectl tools installed and added to your $PATH.

    > Docker installations must have docker version greater than 20.10 and Podman installations must have podman version greater than 3.0.

Install Astra Control Center

1. Log into the NetApp Support Site and download the latest version of NetApp Astra Control Center. TO do so requires a license attached to your NetApp account. After you download the tarball, transfer it to the admin workstation.

   > To get started with a trial license for Astra Control, visit the Astra registration site.

2. Unpack the tar ball and change the working directory to the resulting folder.

   ```
   [netapp-user@rhel7 ~]$ tar -vxzf astra-control-center-21.08.65.tar.gz
   [netapp-user@rhel7 ~]$ cd astra-control-center-21.08.65
   ```

3. Before starting the installation, push the Astra Control Center images to an image registry.

You can choose to do this with either Docker or Podman; instructions for both are provided in this step.

podman

a. Export the registry FQDN with the organization/namespace/project name as a environment variable ‘registry’.

```
[netapp-user@rhel7 ~]$ export registry=astra-registry.apps.ocp-vmw.cie.netapp.com/netapp-astra
```

b. Log into the registry.

```
[netapp-user@rhel7 ~]$ podman login -u ocp-user -p password --tls-verify=false astra-registry.apps.ocp-vmw.cie.netapp.com
```

> If you are using `kubeadmin` user to log into the private registry, then use token instead of password - `podman login -u ocp-user -p token --tls-verify=false astra-registry.apps.ocp-vmw.cie.netapp.com`.

> Alternatively, you can create a service account, assign registry-editor and/or registry-viewer role (based on whether you require push/pull access) and log into the registry using service account’s token.

c. Create a shell script file and paste the following content in it.

```
[netapp-user@rhel7 ~]$ vi push-images-to-registry.sh

for astraImageFile in $(ls images/*.tar); do
   astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded image(s): //')
   podman tag $astraImage $registry/$(echo $astraImage | sed 's/^[^\/]\+\///')
   podman push $registry/$(echo $astraImage | sed 's/^[^\/]\+\///')
done
```

> If you are using untrusted certificates for your registry, edit the shell script and use `--tls-verify=false` for the podman push command `podman push $registry/$(echo $astraImage | sed 's/[\/]\+\///') --tls-verify=false`.

d. Make the file executable.

```
[netapp-user@rhel7 ~]$ chmod +x push-images-to-registry.sh
```

e. Execute the shell script.

```
[netapp-user@rhel7 ~]$ ./push-images-to-registry.sh
```

docker

a. Export the registry FQDN with the organization/namespace/project name as a environment variable 'registry'.

```
[netapp-user@rhel7 ~]$ export registry=astra-registry.apps.ocp-vmw.cie.netapp.com/netapp-astra
```

b. Log into the registry.

```
[netapp-user@rhel7 ~]$ docker login -u ocp-user -p password astra-registry.apps.ocp-vmw.cie.netapp.com
```

> If you are using `kubeadmin` user to log into the private registry, then use token instead of password - `docker login -u ocp-user -p token astra-registry.apps.ocp-vmw.cie.netapp.com`.

> Alternatively, you can create a service account, assign registry-editor and/or registry-viewer role (based on whether you require push/pull access) and log into the registry using service account's token.

c. Create a shell script file and paste the following content in it.

```
[netapp-user@rhel7 ~]$ vi push-images-to-registry.sh

for astraImageFile in $(ls images/*.tar); do
   astraImage=$(docker load --input ${astraImageFile} | sed
's/Loaded image: //')
   docker tag $astraImage $registry/$(echo $astraImage | sed
's/^[^\/]\+\///')
   docker push $registry/$(echo $astraImage | sed 's/^[^\/]\+\///')
done
```

d. Make the file executable.

```
[netapp-user@rhel7 ~]$ chmod +x push-images-to-registry.sh
```

e. Execute the shell script.

```
[netapp-user@rhel7 ~]$ ./push-images-to-registry.sh
```

4. Next, upload the image registry TLS certificates to the OpenShift nodes. To do so, create a configmap in the openshift-config namespace using the TLS certificates and patch it to the cluster image config to make the certificate trusted.

```
[netapp-user@rhel7 ~]$ oc create configmap default-ingress-ca -n
openshift-config --from-file=astra-registry.apps.ocp
-vmw.cie.netapp.com=tls.crt

[netapp-user@rhel7 ~]$ oc patch image.config.openshift.io/cluster
--patch '{"spec":{"additionalTrustedCA":{"name":"default-ingress-ca"}}}'
--type=merge
```

> If you are using an OpenShift internal registry with default TLS certificates from the ingress operator with a route, you still need to follow the previous step to patch the certificates to the route hostname. To extract the certificates from ingress operator, you can use the command `oc extract secret/router-ca --keys=tls.crt -n openshift-ingress-operator`.

5. Create a namespace `netapp-acc-operator` for installing the Astra Control Center Operator.

```
[netapp-user@rhel7 ~]$ oc create ns netapp-acc-operator
```

6. Create a secret with credentials to log into the image registry in `netapp-acc-operator` namespace.

```
[netapp-user@rhel7 ~]$ oc create secret docker-registry astra-registry-
cred --docker-server=astra-registry.apps.ocp-vmw.cie.netapp.com --docker
-username=ocp-user --docker-password=password -n netapp-acc-operator
secret/astra-registry-cred created
```

7. Edit the Astra Control Center Operator CR `astra_control_center_operator_deploy.yaml`, which is a set of all resources Astra Control Center deploys. In the operator CR, find the deployment definition for `acc-operator-controller-manager` and enter the FQDN for your registry along with the organization name as it was given while pushing the images to registry (in this example, `astra-registry.apps.ocp-vmw.cie.netapp.com/netapp-astra`) by replacing the text `ASTRA_IMAGE_REGISTRY` and provide the name of the secret we just created in `imagePullSecrets` section. Verify other details of the operator, save, and close.

```
[netapp-user@rhel7 ~]$ vim astra_control_center_operator_deploy.yaml

apiVersion: apps/v1
```

```
kind: Deployment
metadata:
  labels:
    control-plane: controller-manager
  name: acc-operator-controller-manager
  namespace: netapp-acc-operator
spec:
  replicas: 1
  selector:
    matchLabels:
      control-plane: controller-manager
  template:
    metadata:
      labels:
        control-plane: controller-manager
    spec:
      containers:
      - args:
        - --secure-listen-address=0.0.0.0:8443
        - --upstream=http://127.0.0.1:8080/
        - --logtostderr=true
        - --v=10
        image: ASTRA_IMAGE_REGISTRY/kube-rbac-proxy:v0.5.0
        name: kube-rbac-proxy
        ports:
        - containerPort: 8443
          name: https
      - args:
        - --health-probe-bind-address=:8081
        - --metrics-bind-address=127.0.0.1:8080
        - --leader-elect
        command:
        - /manager
        env:
        - name: ACCOP_LOG_LEVEL
          value: "2"
        image: astra-registry.apps.ocp-vmw.cie.netapp.com/netapp-
astra/acc-operator:21.08.7
        imagePullPolicy: IfNotPresent
        livenessProbe:
          httpGet:
            path: /healthz
            port: 8081
          initialDelaySeconds: 15
          periodSeconds: 20
        name: manager
```

```
        readinessProbe:
          httpGet:
            path: /readyz
            port: 8081
          initialDelaySeconds: 5
          periodSeconds: 10
        resources:
          limits:
            cpu: 300m
            memory: 750Mi
          requests:
            cpu: 100m
            memory: 75Mi
        securityContext:
          allowPrivilegeEscalation: false
      imagePullSecrets: [name: astra-registry-cred]
      securityContext:
        runAsUser: 65532
      terminationGracePeriodSeconds: 10
```

8. Create the operator by running the following command.

```
[netapp-user@rhel7 ~]$ oc create -f
astra_control_center_operator_deploy.yaml
```

9. Create a dedicated namespace for installing all the Astra Control Center resources.

```
[netapp-user@rhel7 ~]$ oc create ns netapp-astra-cc
namespace/netapp-astra-cc created
```

10. Create the secret for accessing the image registry in that namespace.

```
[netapp-user@rhel7 ~]$ oc create secret docker-registry astra-registry-
cred --docker-server=astra-registry.apps.ocp-vmw.cie.netapp.com --docker
-username=ocp-user --docker-password=password -n netapp-astra-cc

secret/astra-registry-cred created
```

11. Edit the Astra Control Center CRD file `astra_control_center_min.yaml` and enter the FQDN, image registry details, administrator email address, and other details.

```
[netapp-user@rhel7 ~]$ vim astra_control_center_min.yaml

apiVersion: astra.netapp.io/v1
kind: AstraControlCenter
metadata:
  name: astra
spec:
  accountName: "NetApp HCG Solutions"
  astraVersion: "21.08.65"
  astraAddress: "astra-control-center.cie.netapp.com"
  autoSupport:
    enrolled: true
  email: "solutions_tme@netapp.com"
  firstName: "NetApp HCG"
  lastName: "Admin"
  imageRegistry:
    name: "astra-registry.apps.ocp-vmw.cie.netapp.com/netapp-astra"
# use your registry
    secret: "astra-registry-cred"             # comment out if not
needed
```

12. Create the Astra Control Center CRD in the namespace created for it.

```
[netapp-user@rhel7 ~]$ oc apply -f astra_control_center_min.yaml -n
netapp-astra-cc
astracontrolcenter.astra.netapp.io/astra created
```

> The previous file `astra_control_center_min.yaml` is the minimum version of the Astra Control Center CRD. If you want to create the CRD with more control, such as defining a storageclass other than the default for creating PVCs or providing SMTP details for mail notifications, you can edit the file `astra_control_center.yaml`, enter then needed details, and use it to create the CRD.

Installation verification

1. It might take several minutes for the installation to complete. Verify that all the pods and services in the `netapp-astra-cc` namespace are up and running.

```
[netapp-user@rhel7 ~]$ oc get all -n netapp-astra-cc
```

2. Check the `acc-operator-controller-manager` logs to ensure that the installation is completed.

```
[netapp-user@rhel7 ~]$ oc logs deploy/acc-operator-controller-manager -n
netapp-acc-operator -c manager -f
```

The following message indicates the successful installation of Astra Control Center.

```
{"level":"info","ts":1624054318.029971,"logger":"controllers.AstraContro
lCenter","msg":"Successfully Reconciled AstraControlCenter in
[seconds]s","AstraControlCenter":"netapp-astra-
cc/astra","ae.Version":"[21.08.65]"}
```

3. The username for logging into Astra Control Center is the email address of the administrator provided in the CRD file and the password is a string `ACC-` appended to the Astra Control Center UUID. Run the following command:

```
[netapp-user@rhel7 ~]$ oc get astracontrolcenters -n netapp-astra-cc
NAME    UUID
astra   345c55a5-bf2e-21f0-84b8-b6f2bce5e95f
```

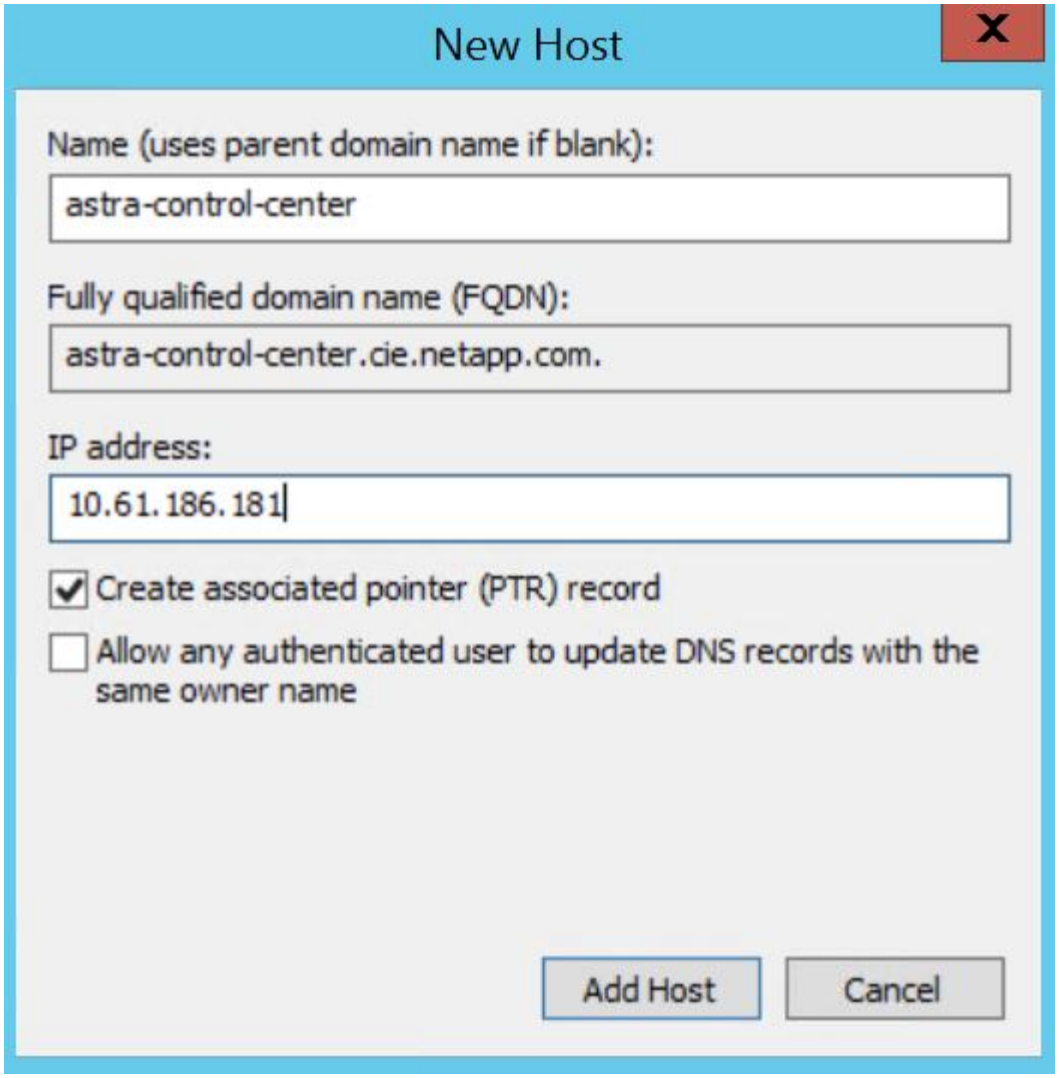
In this example, the password is `ACC-345c55a5-bf2e-21f0-84b8-b6f2bce5e95f`.

4. Get the traefik service load balancer IP.

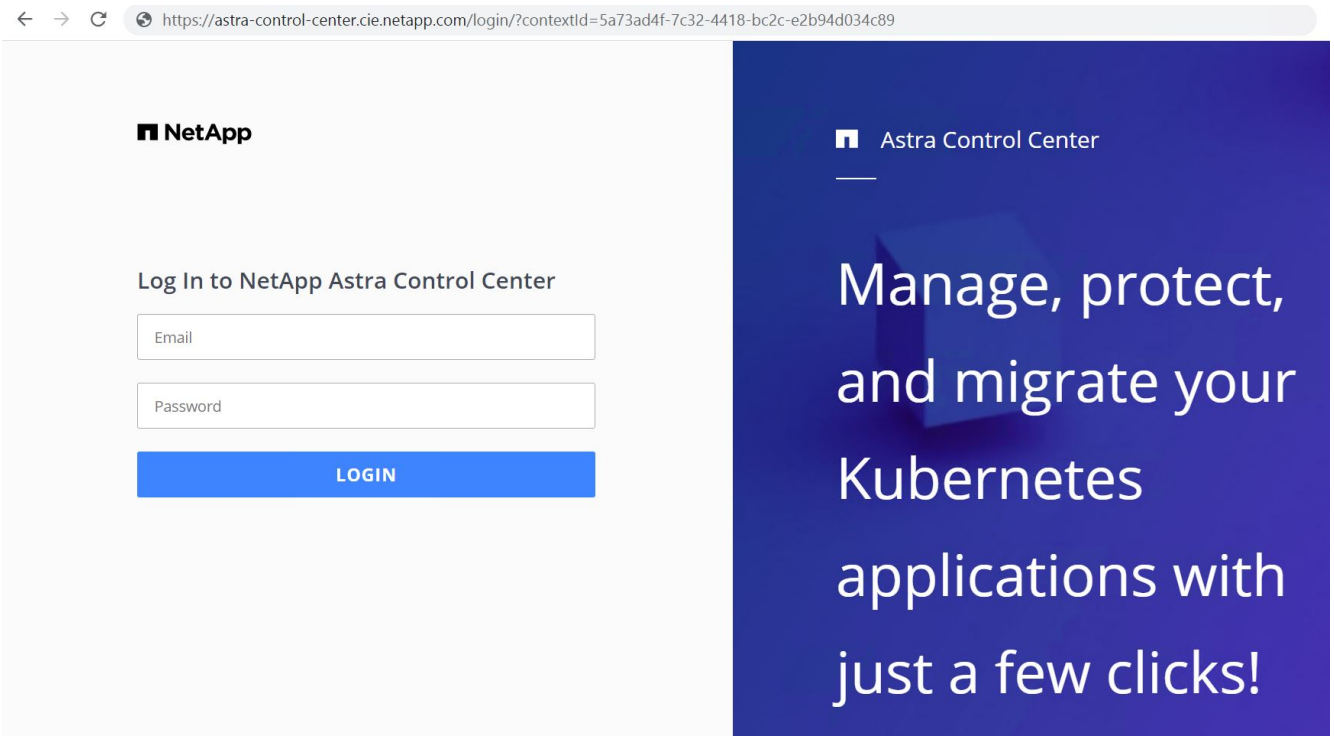
```
[netapp-user@rhel7 ~]$ oc get svc -n netapp-astra-cc | egrep
'EXTERNAL|traefik'

NAME                                        TYPE           CLUSTER-IP
EXTERNAL-IP     PORT(S)
AGE
traefik                                     LoadBalancer   172.30.99.142
10.61.186.181   80:30343/TCP,443:30060/TCP
16m
```

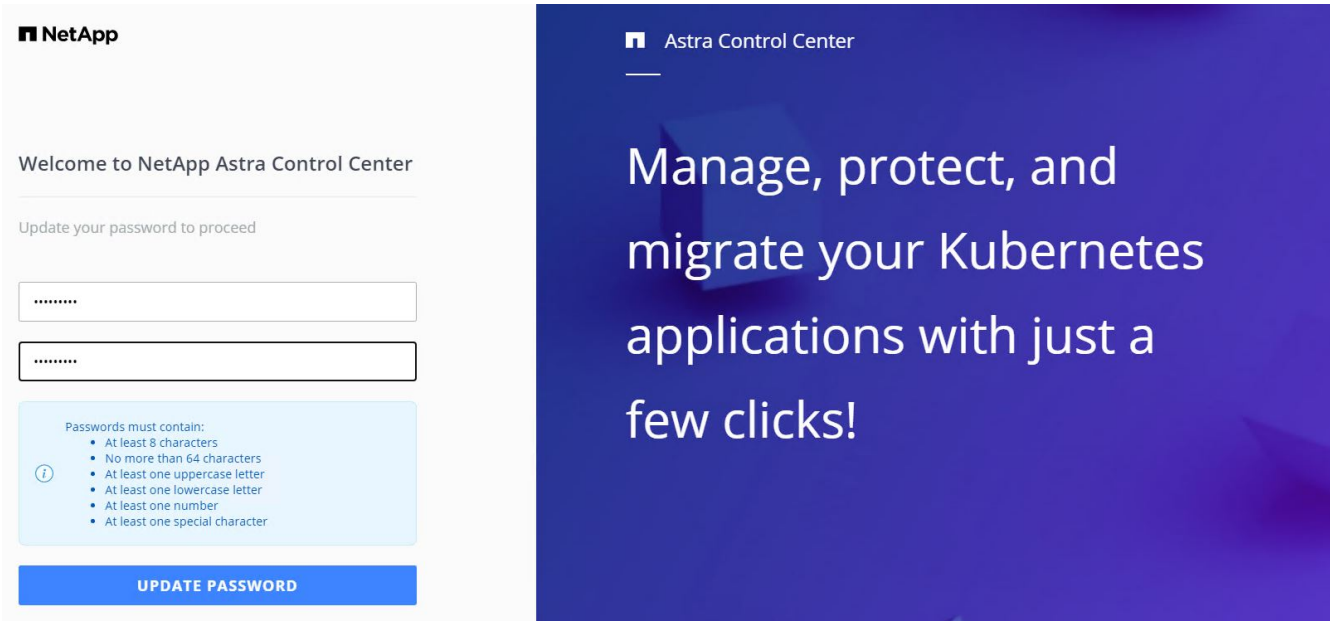
5. Add an entry in the DNS server pointing the FQDN provided in the Astra Control Center CRD file to the `EXTERNAL-IP` of the traefik service.



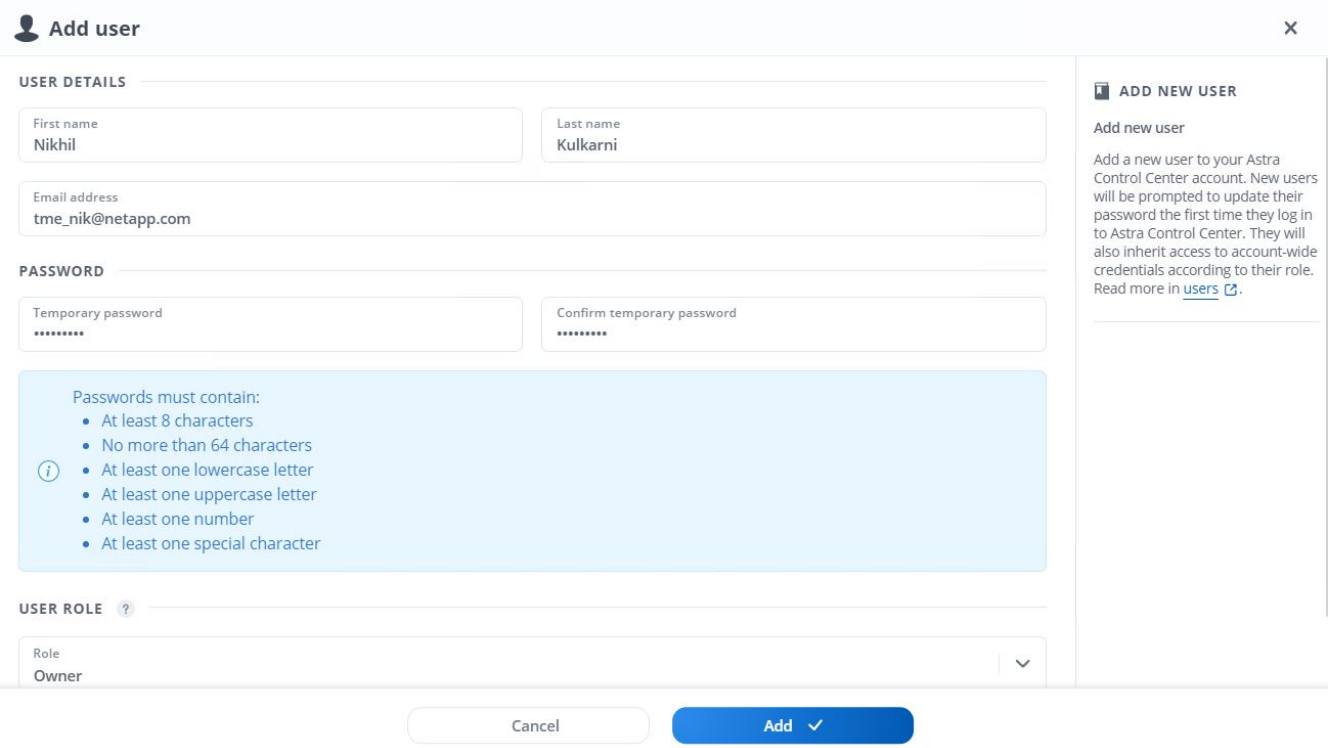
6. Log into the Astra Control Center GUI by browsing its FQDN.



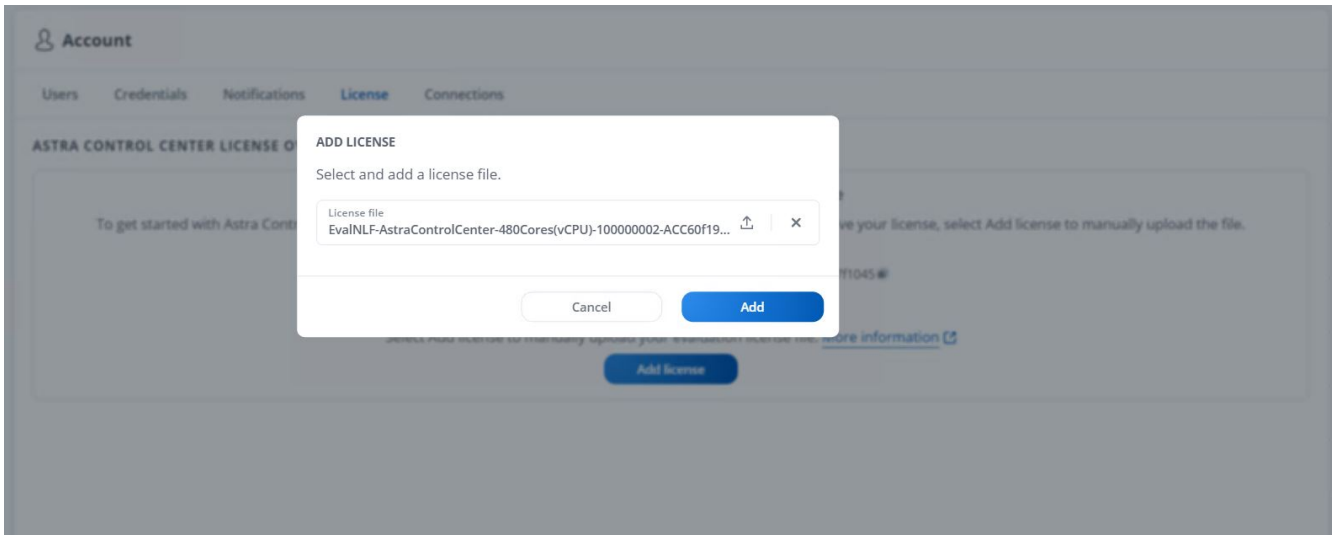
7. When you log into Astra Control Center GUI for the first time using the admin email address provided in CRD, you need to change the password.



8. If you wish to add a user to Astra Control Center, navigate to Account > Users, click Add, enter the details of the user, and click Add.



9. Astra Control Center requires a license for all of it's functionalities to work. To add a license, navigate to Account > License, click Add License, and upload the license file.



If you encounter issues with the install or configuration of NetApp Astra Control Center, the knowledge base of known issues is available here.

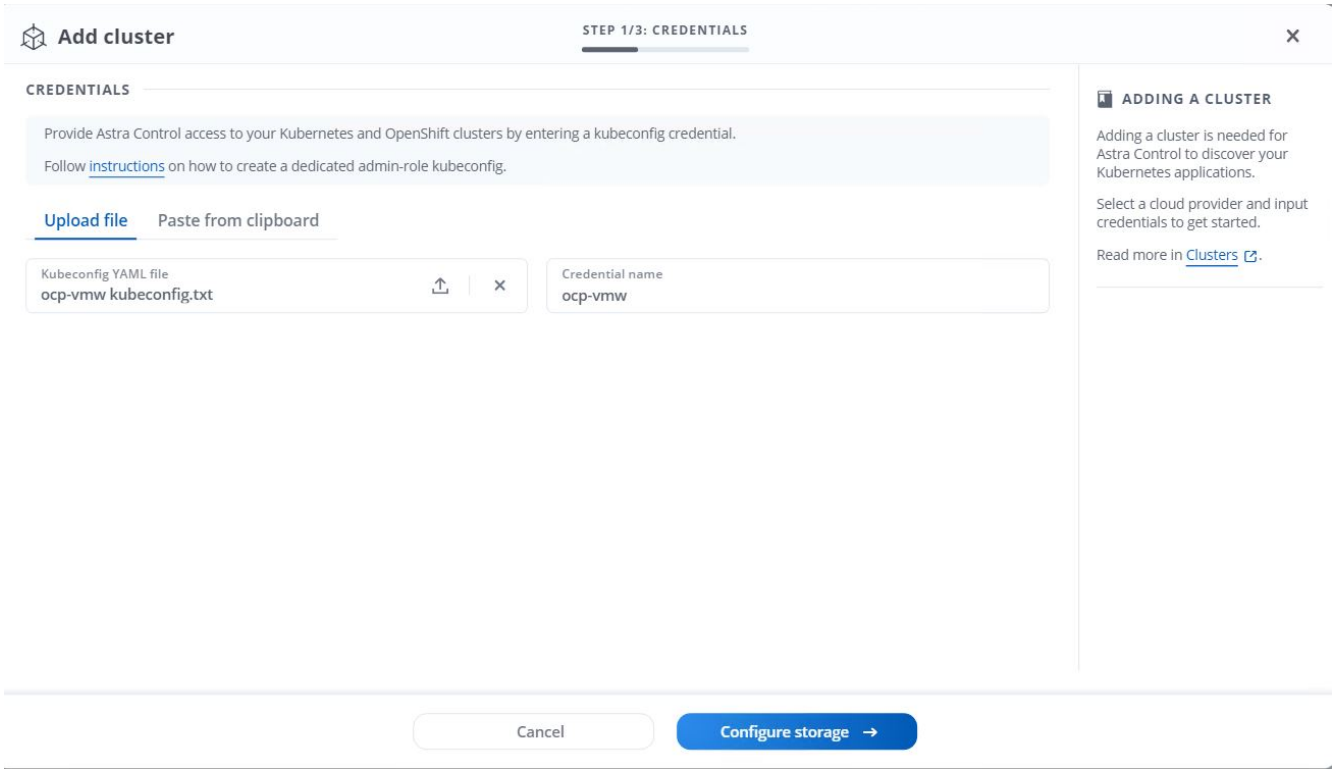
Next: Register your Red Hat OpenShift Clusters: Red Hat OpenShift with NetApp.

Register your Red Hat OpenShift Clusters with the Astra Control Center

To enable the Astra Control Center to manage your workloads, you must first register your Red Hat OpenShift cluster.

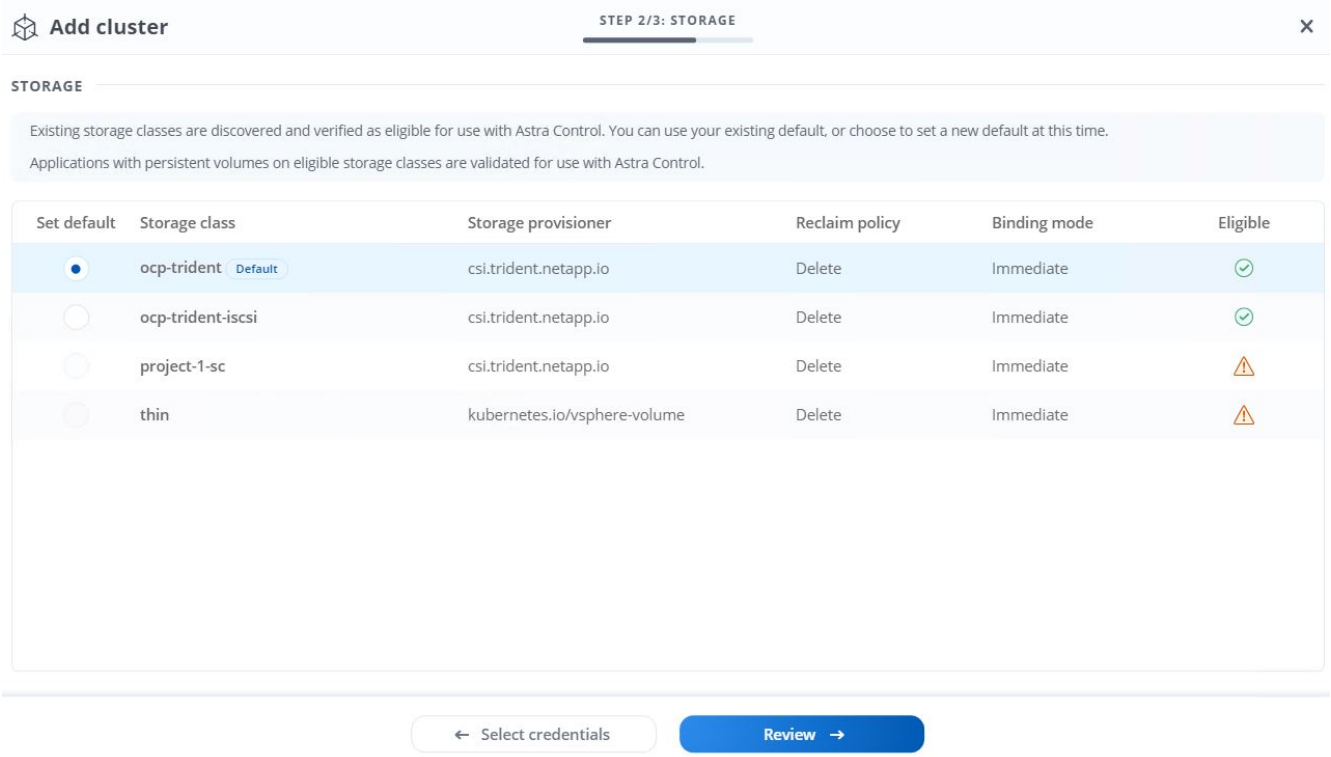
Register Red Hat OpenShift clusters

1. The first step is to add the OpenShift clusters to the Astra Control Center and manage them. Go to Clusters and click Add a Cluster, upload the kubeconfig file for the OpenShift cluster, and click Select Storage.

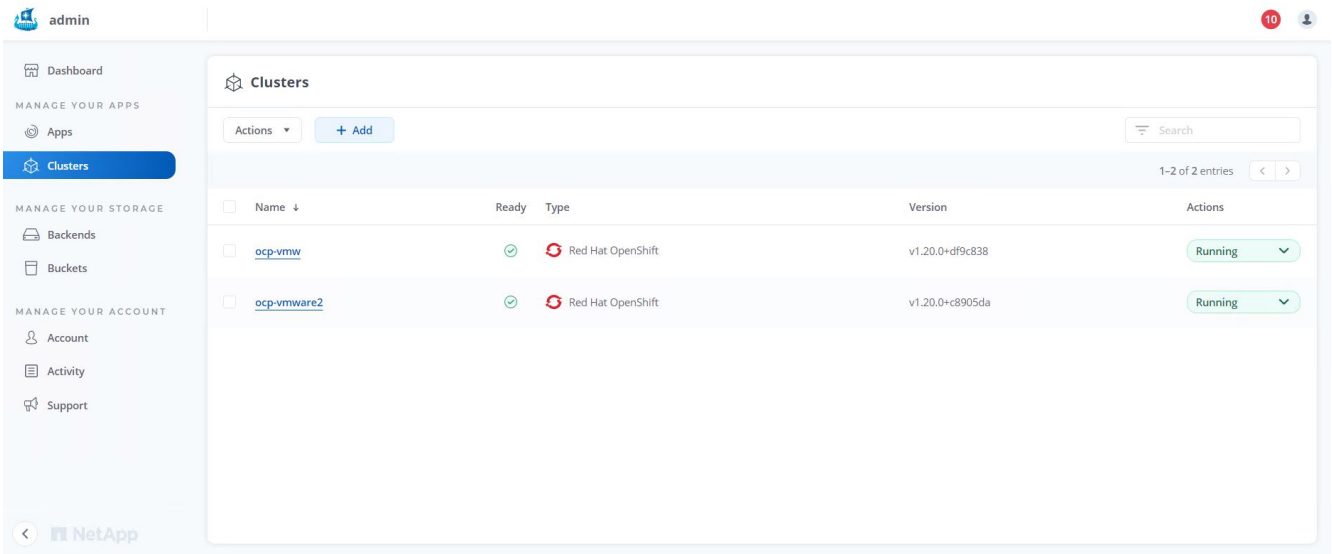


> The kubeconfig file can be generated to authenticate with a username and password or a token. Tokens expire after a limited amount of time and might leave the registered cluster unreachable. NetApp recommends using a kubeconfig file with a username and password to register your OpenShift clusters to Astra Control Center.

2. Astra Control Center detects the eligible storage classes. Now select the way that storageclass provisions volumes using Trident backed by an SVM on NetApp ONTAP and click Review. In the next pane, verify the details and click Add Cluster.

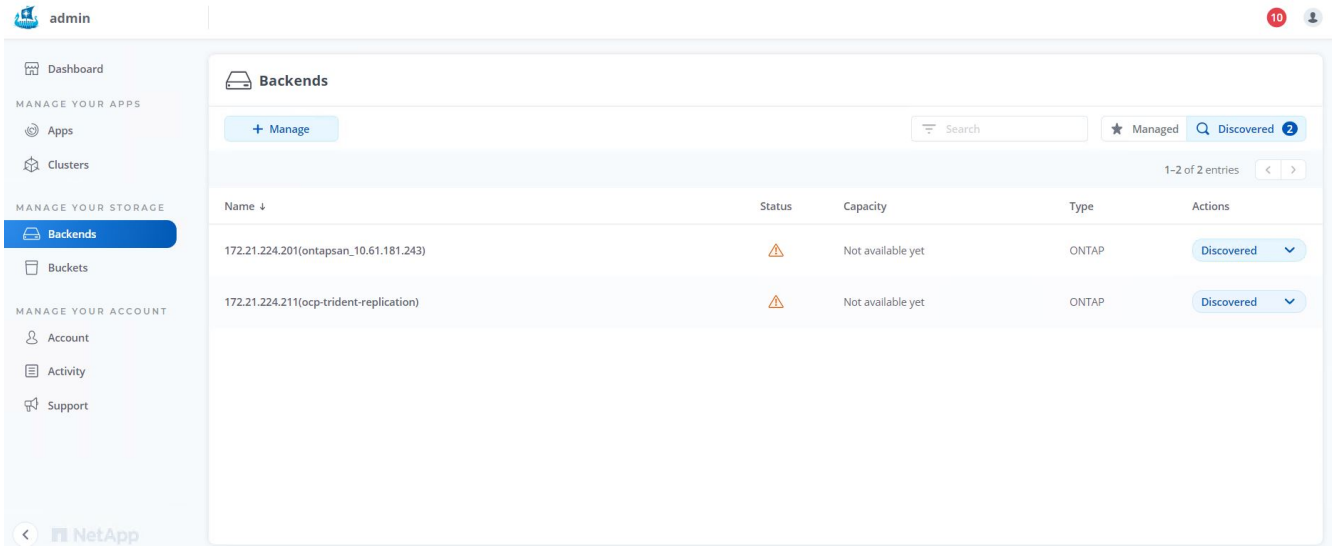


3. Register both OpenShift clusters as described in step 1. When added, the clusters move to the Discovering status while Astra Control Center inspects them and installs the necessary agents. Cluster status changes to Running after they are successfully registered.

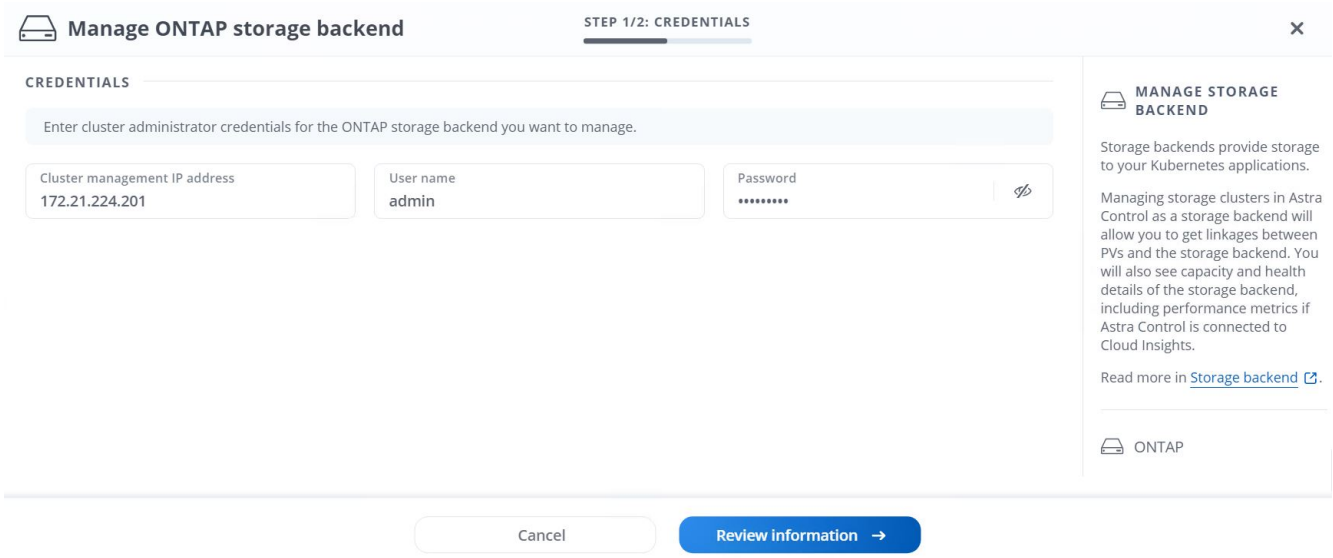


> All Red Hat OpenShift clusters to be managed by Astra Control Center should have access to the image registry that was used for its installation as the agents installed on the managed clusters pull the images from that registry.

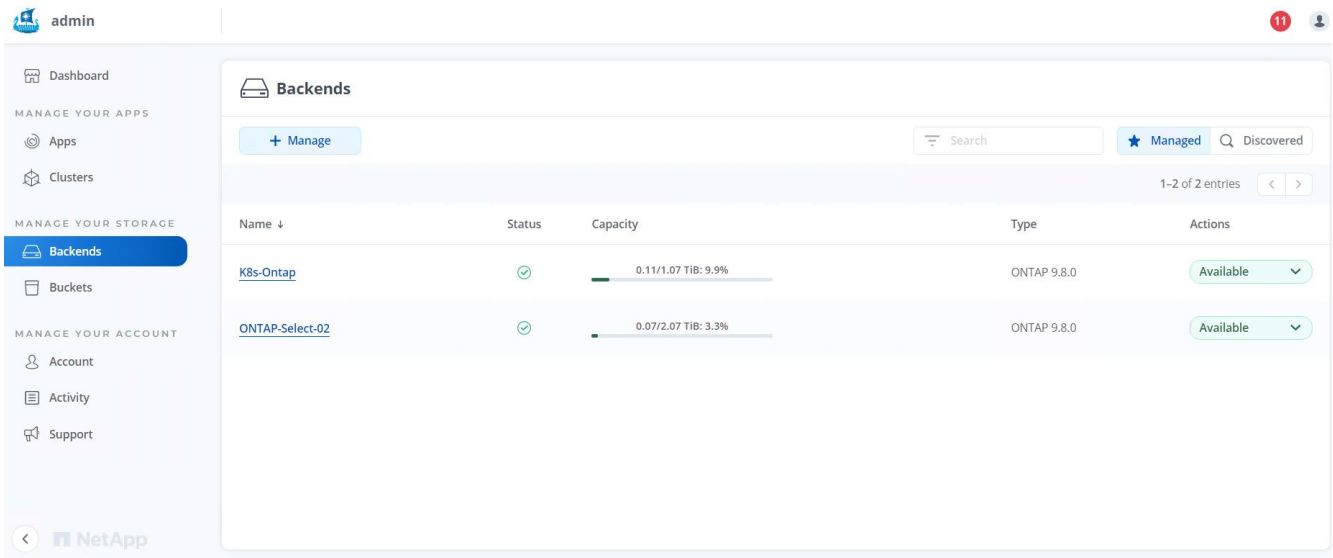
4. Import ONTAP clusters as storage resources to be managed as backends by Astra Control Center. When OpenShift clusters are added to Astra and a storageclass is configured, it automatically discovers and inspects the ONTAP cluster backing the storageclass but does not import it into the Astra Control Center to be managed.



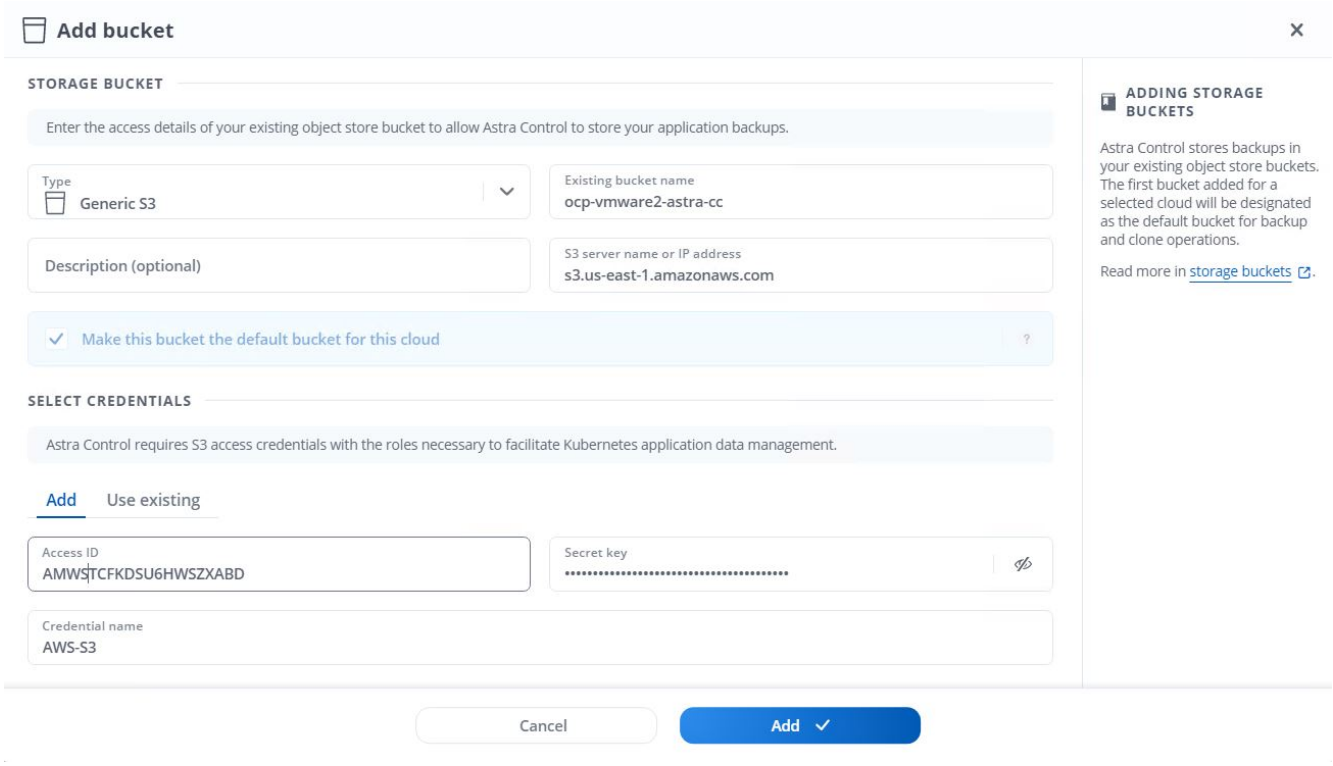
5. To import the ONTAP clusters, go to Backends, click the dropdown, and select Manage next to the ONTAP cluster to be managed. Enter the ONTAP cluster credentials, click Review Information, and then click Import Storage Backend.



6. After the backends are added, the status changes to Available. These backends now have the information about the persistent volumes in the OpenShift cluster and the corresponding volumes on the ONTAP system.



7. For backup and restore across OpenShift clusters using Astra Control Center, you must provision an object storage bucket that supports the S3 protocol. Currently supported options are ONTAP S3, StorageGRID, and AWS S3. For the purpose of this installation, we are going to configure an AWS S3 bucket. Go to Buckets, click Add bucket, and select Generic S3. Enter the details about the S3 bucket and credentials to access it, click the checkbox "Make this bucket the default bucket for the cloud," and then click Add.



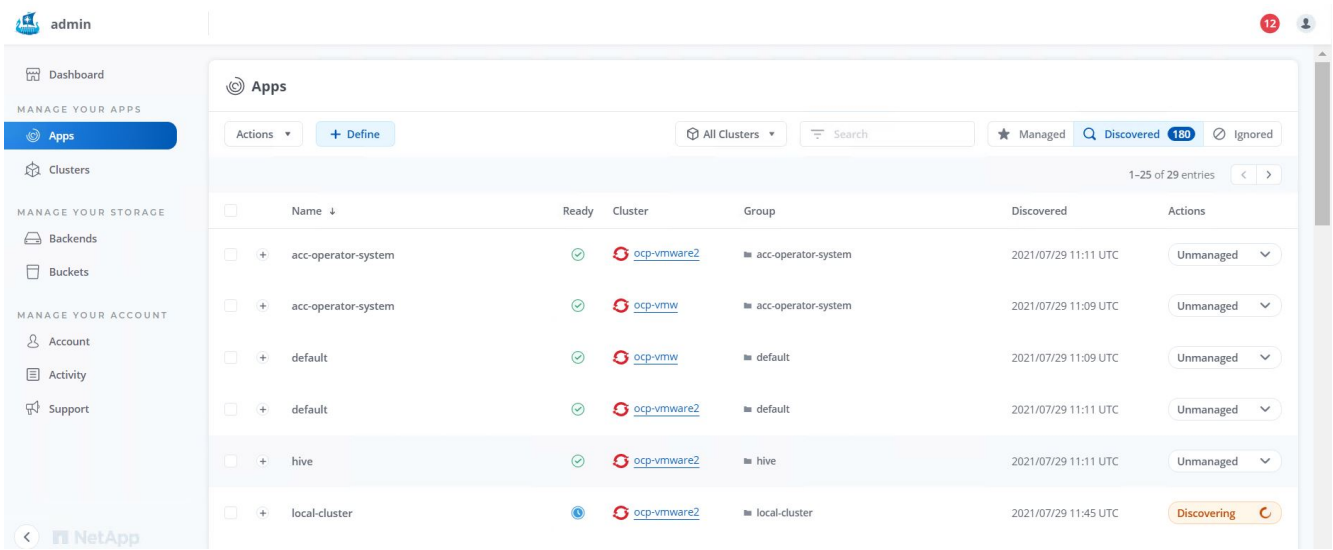
Next: Choose the Applications To Protect.

Choose the applications to protect

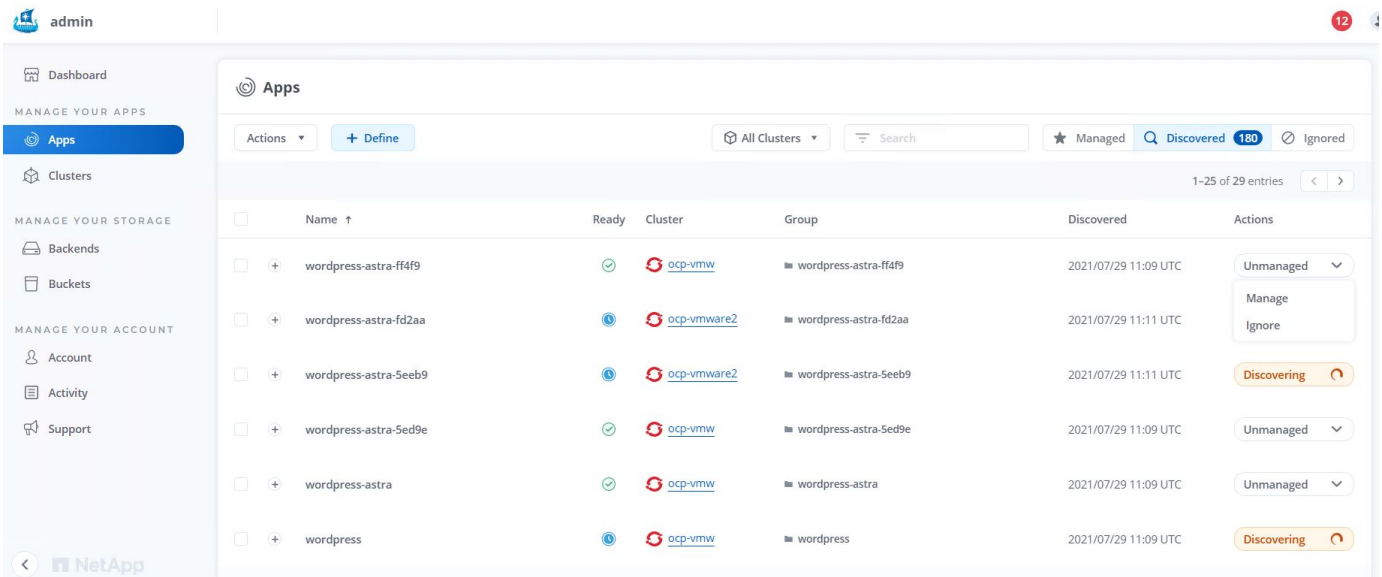
After you have registered your Red Hat OpenShift clusters, you can discover the applications that are deployed and manage them via the Astra Control Center.

Manage applications

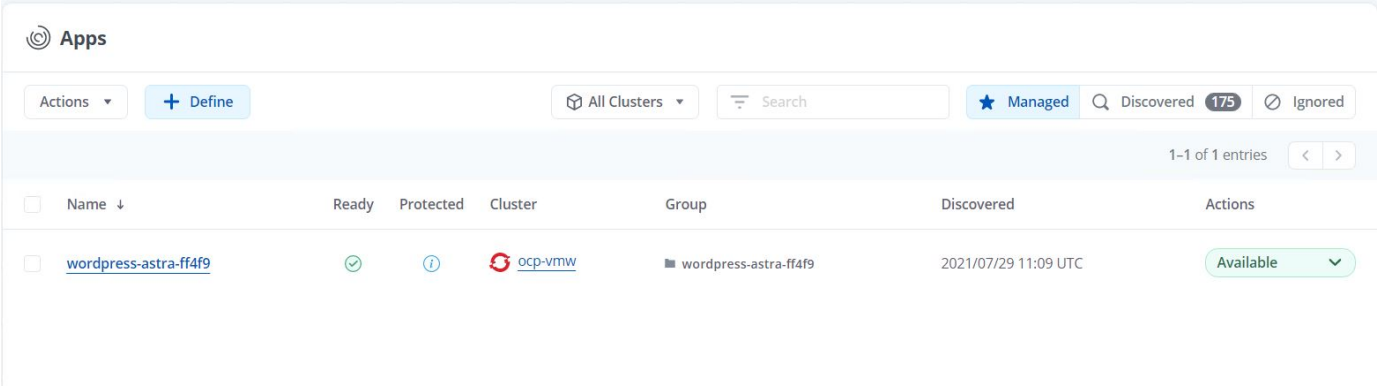
1. After the OpenShift clusters and ONTAP backends are registered with the Astra Control Center, the control center automatically starts discovering the applications in all the namespaces that are using the storageclass configured with the specified ONTAP backend.



2. Navigate to Apps > Discovered and click the dropdown menu next to the application you would like to manage using Astra. Then click Manage.



1. The application enters the Available state and can be viewed under the Managed tab in the Apps section.



Next: Protect Your applications.

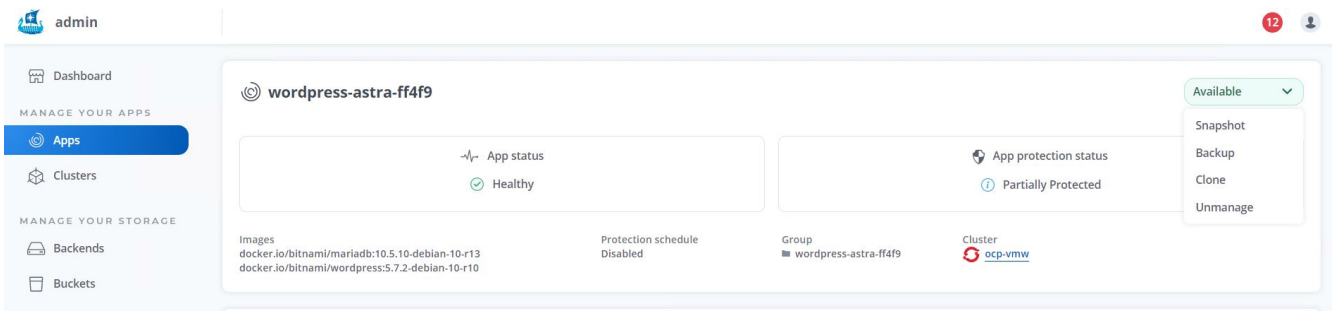
Protect your applications

After application workloads are managed by Astra Control Center, you can configure the protection settings for those workloads.

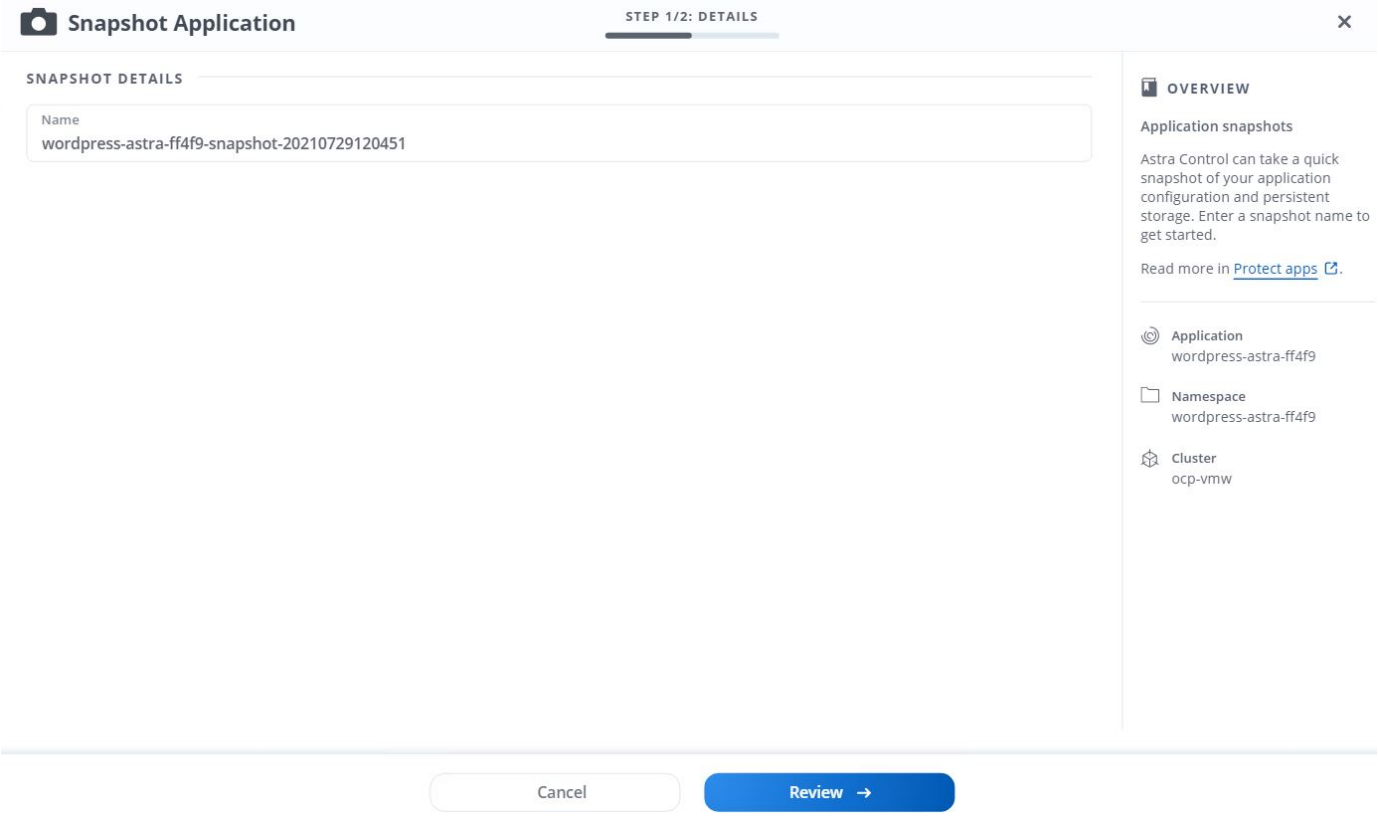
Creating an application snapshot

A snapshot of an application creates an ONTAP Snapshot copy that can be used to restore or clone the application to a specific point in time based on that Snapshot copy.

1. To take a snapshot of the application, navigate to the Apps > Managed tab and click the application you would like to make a Snapshot copy of. Click the dropdown menu next to the application name and click Snapshot.



2. Enter the snapshot details, click Review, and then click Snapshot. It takes about a minute to create the snapshot, and the status becomes Available after the snapshot is successfully created.



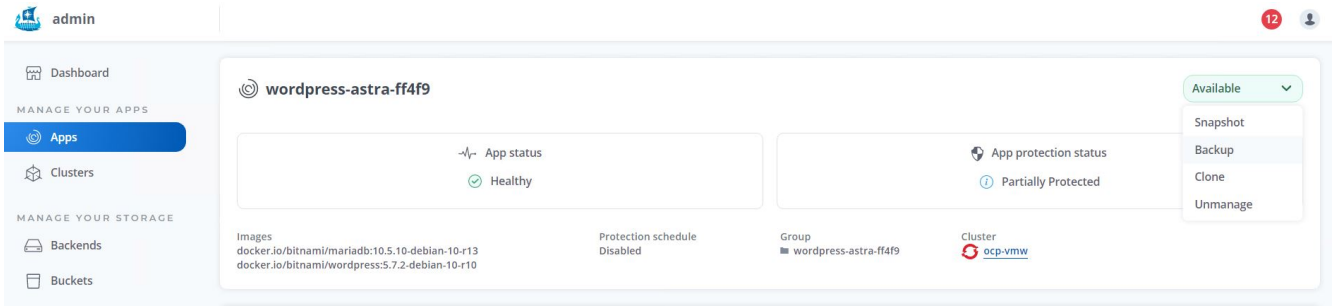
Creating an application backup

A backup of an application captures the active state of the application and the configuration of it's resources, coverts them into files, and stores them in a remote object storage bucket.

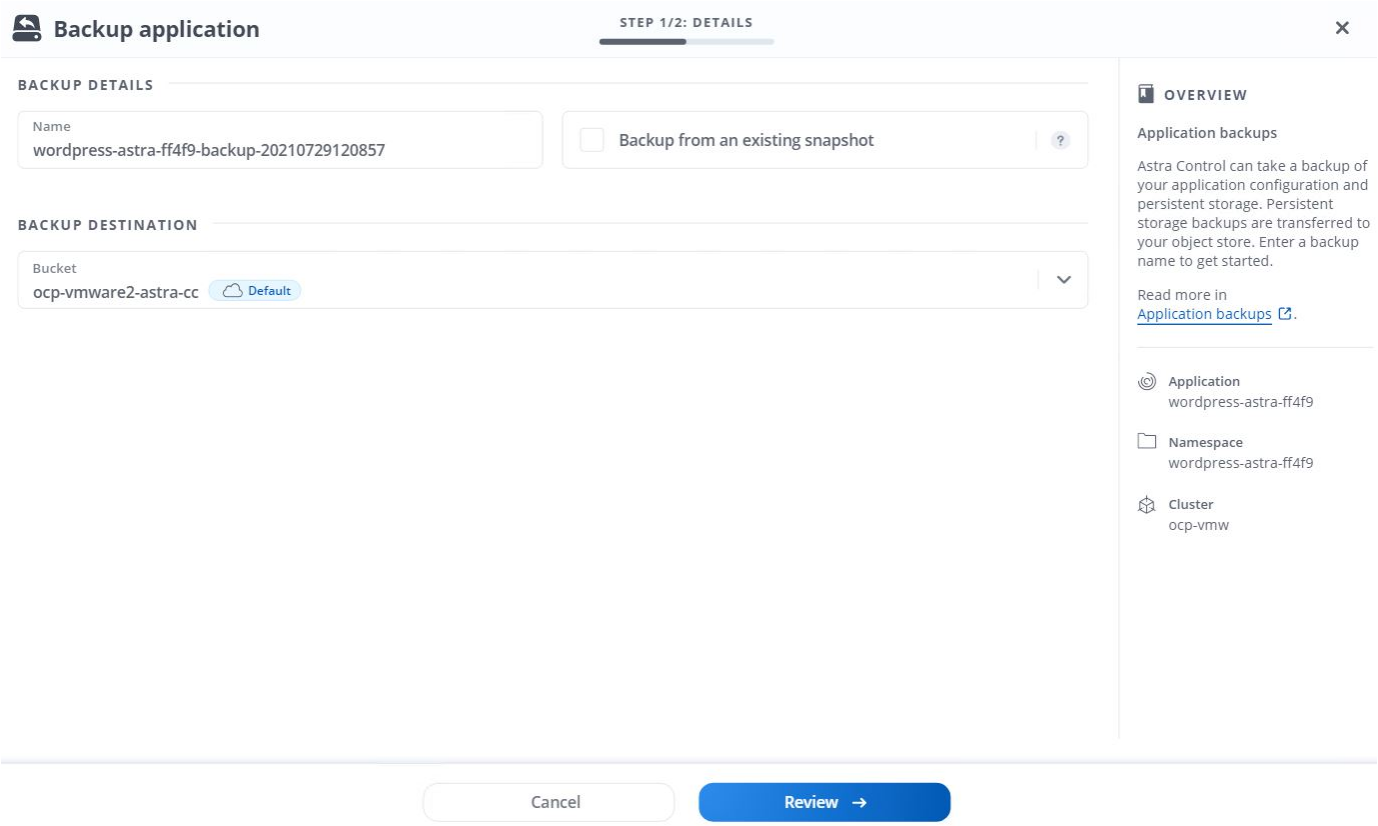
For the backup and restore of managed applications in the Astra Control Center, you must configure superuser settings for the backing ONTAP systems as a prerequisite. To do so, enter the following commands.

```
ONTAP::> export-policy rule modify -vserver ocp-trident -policyname
default -ruleindex 1 -superuser sys
ONTAP::> export-policy rule modify -policyname default -ruleindex 1 -anon
65534 -vserver ocp-trident
```

1. To create a backup of the managed application in the Astra Control Center, navigate to the Apps > Managed tab and click the application that you want to take a backup of. Click the dropdown menu next to the application name and click Backup.



2. Enter the backup details, select the object storage bucket to hold the backup files, click Review, and, after reviewing the details, click Backup. Depending on the size of the application and data, the backup can take several minutes, and the status of the backup becomes Available after the backup is completed successfully.

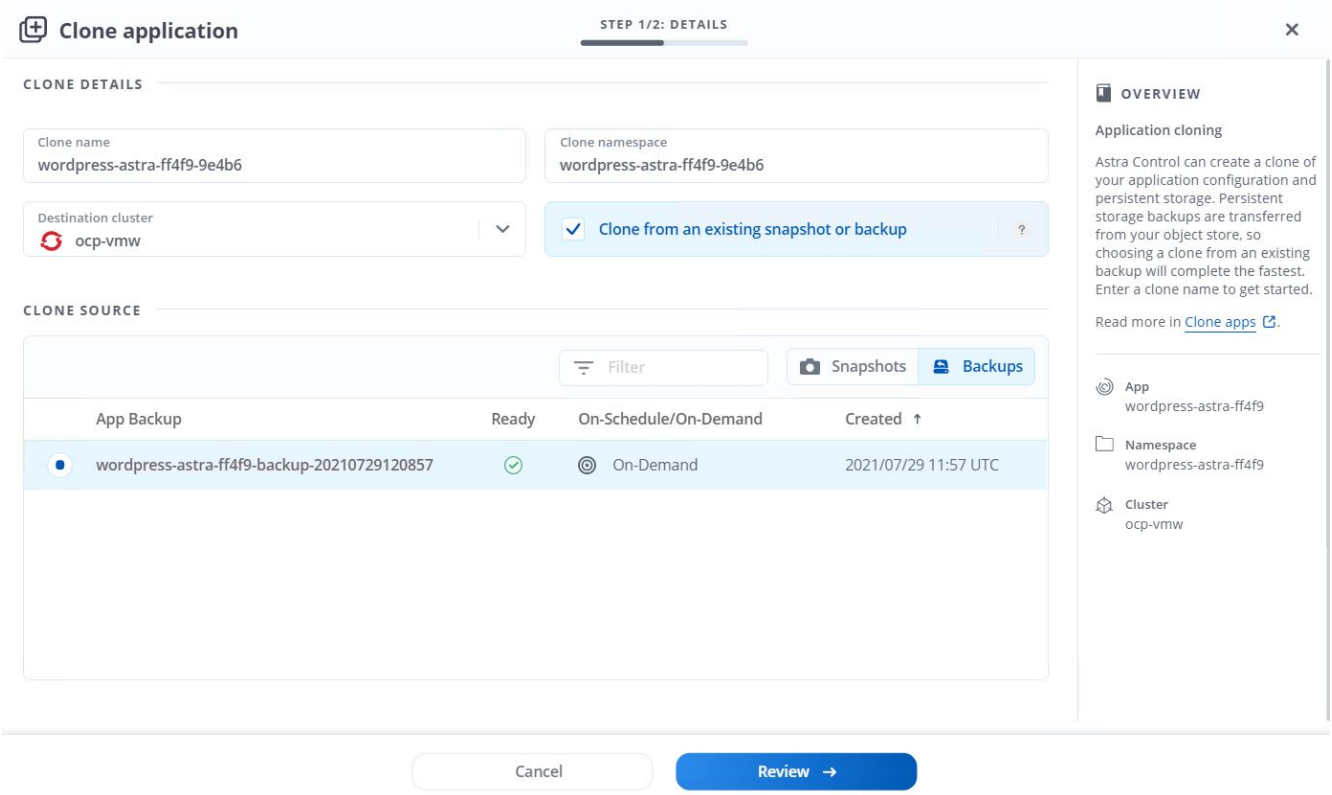


Restoring or cloning an application

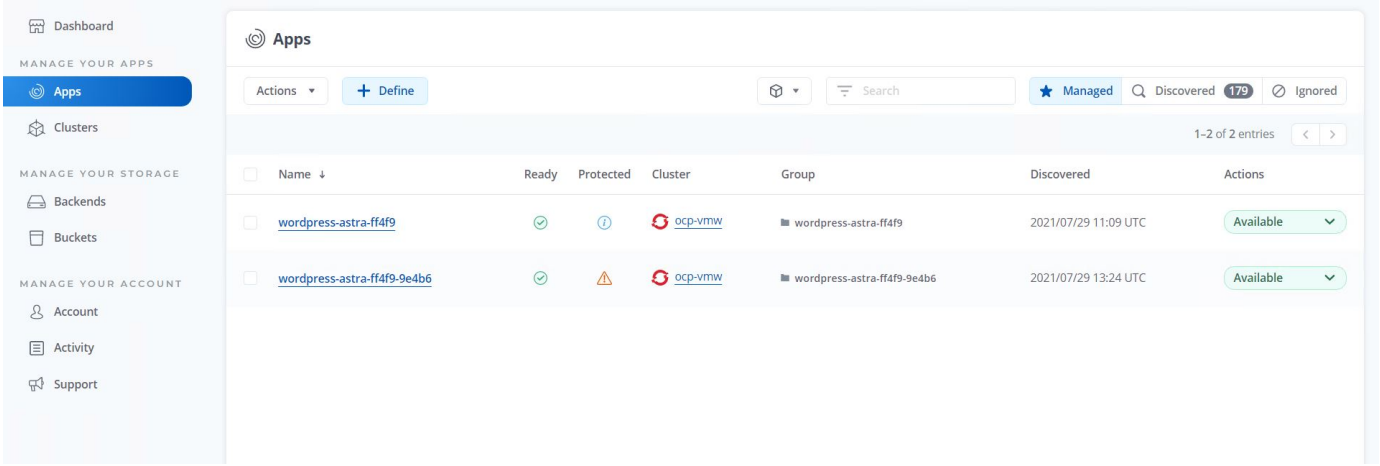
At the push of a button, you can restore an application to the originating cluster or clone it to a remote cluster for dev/test or application protection and disaster recovery purposes.

1. To restore or clone an application, navigate to the Apps > Managed tab and click the app in question. Click the dropdown menu next to the application name and click Clone.

2. Enter the details of the new namespace, select the cluster you want to restore or clone it to, and choose if you want to restore or clone it from an existing snapshot or from a backup of the current state of the application. Then click Review and click Clone after you have reviewed the details.



3. The new application goes to the Discovering state while Astra Control Center creates the application on the selected cluster. After all the resources of the application are installed and detected by Astra, the application goes to the Available state.



Next: Solution Validation/Use Cases.