Concepts

Astra Control Center

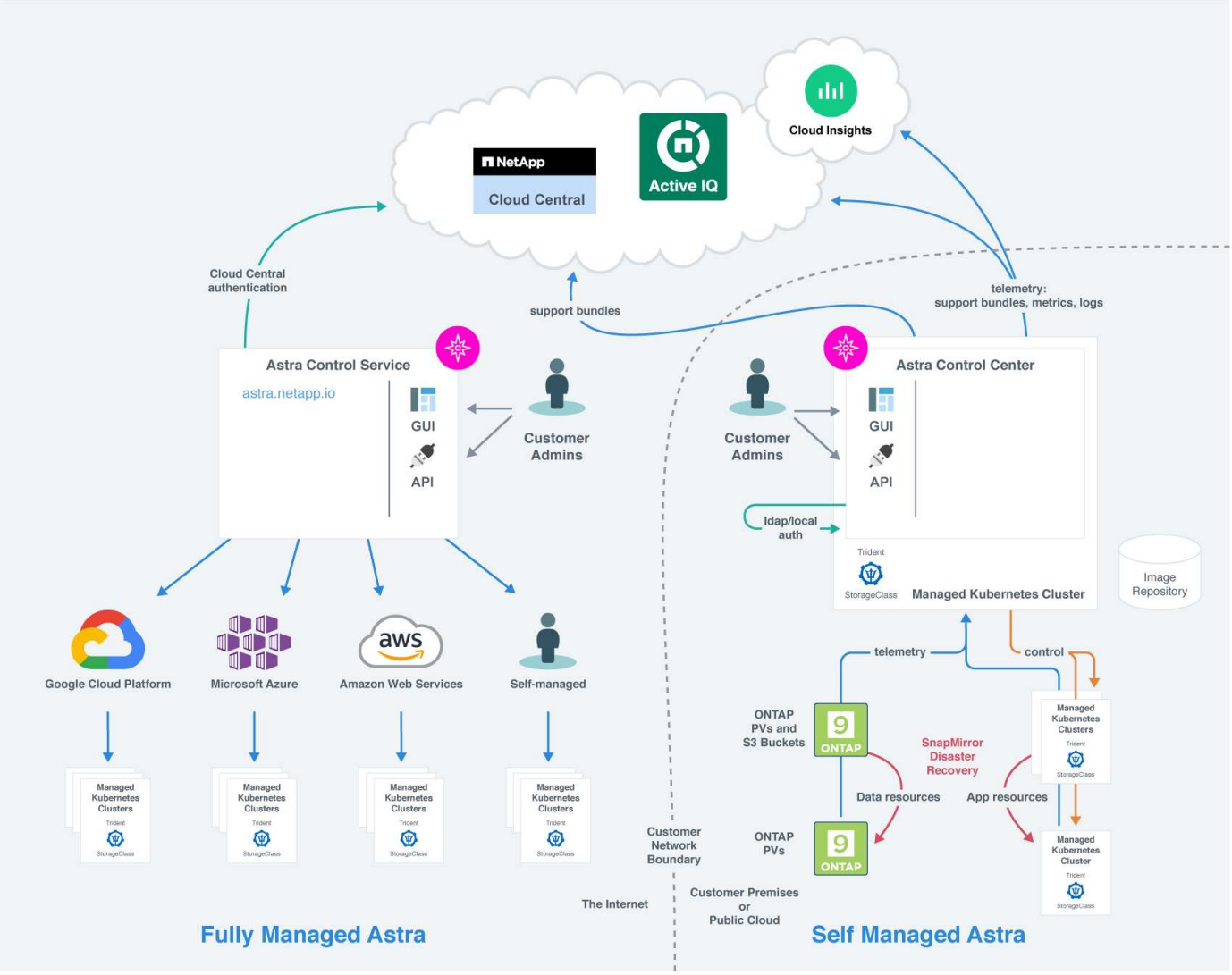
NetApp
June 11, 2024

Table of Contents

Concepts

Architecture and components

Here is an overview of the various components of the Astra Control environment.



Astra Control components

- **Kubernetes clusters**: Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation. Astra provides management services for applications hosted in a Kubernetes cluster.
- **Astra Trident**: As a fully supported open source storage provisioner and orchestrator maintained by NetApp, Astra Trident enables you to create storage volumes for containerized applications managed by Docker and Kubernetes. When deployed with Astra Control Center, Astra Trident includes a configured ONTAP storage backend.
- **Storage backend**:
  - Astra Control Service uses the following storage backends:
    - NetApp Cloud Volumes Service for Google Cloud or Google Persistent Disk as the storage

backend for GKE clusters
        - Azure NetApp Files or Azure Managed Disks as the storage backend for AKS clusters.
        - Amazon Elastic Block Store (EBS) or Amazon FSx for NetApp ONTAP as backend storage options for EKS clusters.
    - Astra Control Center uses the following storage backends:
        - ONTAP AFF, FAS, and ASA. As a storage software and hardware platform, ONTAP provides core storage services, support for multiple storage access protocols, and storage management functionality, such as snapshots and mirroring.
        - Cloud Volumes ONTAP
- **Cloud Insights**: A NetApp cloud infrastructure monitoring tool, Cloud Insights enables you to monitor performance and utilization for your Kubernetes clusters managed by Astra Control Center. Cloud Insights correlates storage usage to workloads. When you enable the Cloud Insights connection in Astra Control Center, telemetry information shows in Astra Control Center UI pages.

Astra Control interfaces

You can complete tasks using different interfaces:

- **Web user interface (UI)**: Both Astra Control Service and Astra Control Center use the same web-based UI where you can manage, migrate and protect apps. Use the UI also to manage user accounts and configuration settings.
- **API**: Both Astra Control Service and Astra Control Center use the same Astra Control API. Using the API, you can perform the same tasks that you would using the UI.

Astra Control Center also enables you to manage, migrate, and protect Kubernetes clusters running within VM environments.

For more information

- Astra Control Service documentation
- Astra Control Center documentation
- Astra Trident documentation
- Use the Astra Control API
- Cloud Insights documentation
- ONTAP documentation

Data protection

Learn about the available types of data protection in Astra Control Center, and how best to use them to protect your apps.

Snapshots, backups, and protection policies

Both snapshots and backups protect the following types of data:

- The application itself
- Any persistent data volumes associated with the application

- Any resource artifacts belonging to the application

A *snapshot* is a point-in-time copy of an app that's stored on the same provisioned volume as the app. They are usually fast. You can use local snapshots to restore the application to an earlier point in time. Snapshots are useful for fast clones; snapshots include all of the Kubernetes objects for the app, including configuration files. Snapshots are useful for cloning or restoring an app within the same cluster.

A *backup* is based on a snapshot. It is stored in the external object store, and because of this, can be slower to take compared to local snapshots. You can restore an app backup to the same cluster, or you can migrate an app by restoring its backup to a different cluster. You can also choose a longer retention period for backups. Because they are stored in the external object store, backups generally offer you better protection than snapshots in cases of server failure or data loss.

A *protection policy* is a way to protect an app by automatically creating snapshots, backups, or both according to a schedule that you define for that app. A protection policy also enables you to choose how many snapshots and backups to retain in the schedule, and set different schedule granularity levels. Automating your backups and snapshots with a protection policy is the best way to ensure each app is protected according to the needs of your organization and service level agreement (SLA) requirements.

You can't be fully protected until you have a recent backup. This is important because backups are stored in an object store away from the persistent volumes. If a failure or accident wipes out the cluster and its associated persistent storage, then you need a backup to recover. A snapshot would not enable you to recover.

Clones

A *clone* is an exact duplicate of an app, its configuration, and its persistent data volumes. You can manually create a clone on either the same Kubernetes cluster or on another cluster. Cloning an app can be useful if you need to move applications and storage from one Kubernetes cluster to another.

Replication between storage backends

Using Astra Control, you can build business continuity for your applications with a low-RPO (Recovery Point Objective) and low-RTO (Recovery Time Objective) using asynchronous replication capabilities of NetApp SnapMirror technology. Once configured, this enables your applications to replicate data and application changes from one storage backend to another, on the same cluster or between different clusters.

You can replicate between two ONTAP SVMs on the same ONTAP cluster or on different ONTAP clusters.

Astra Control asynchronously replicates app snapshot copies to a destination cluster. The replication process includes data in the persistent volumes replicated by SnapMirror and the app metadata protected by Astra Control.

App replication is different from app backup and restore in the following ways:

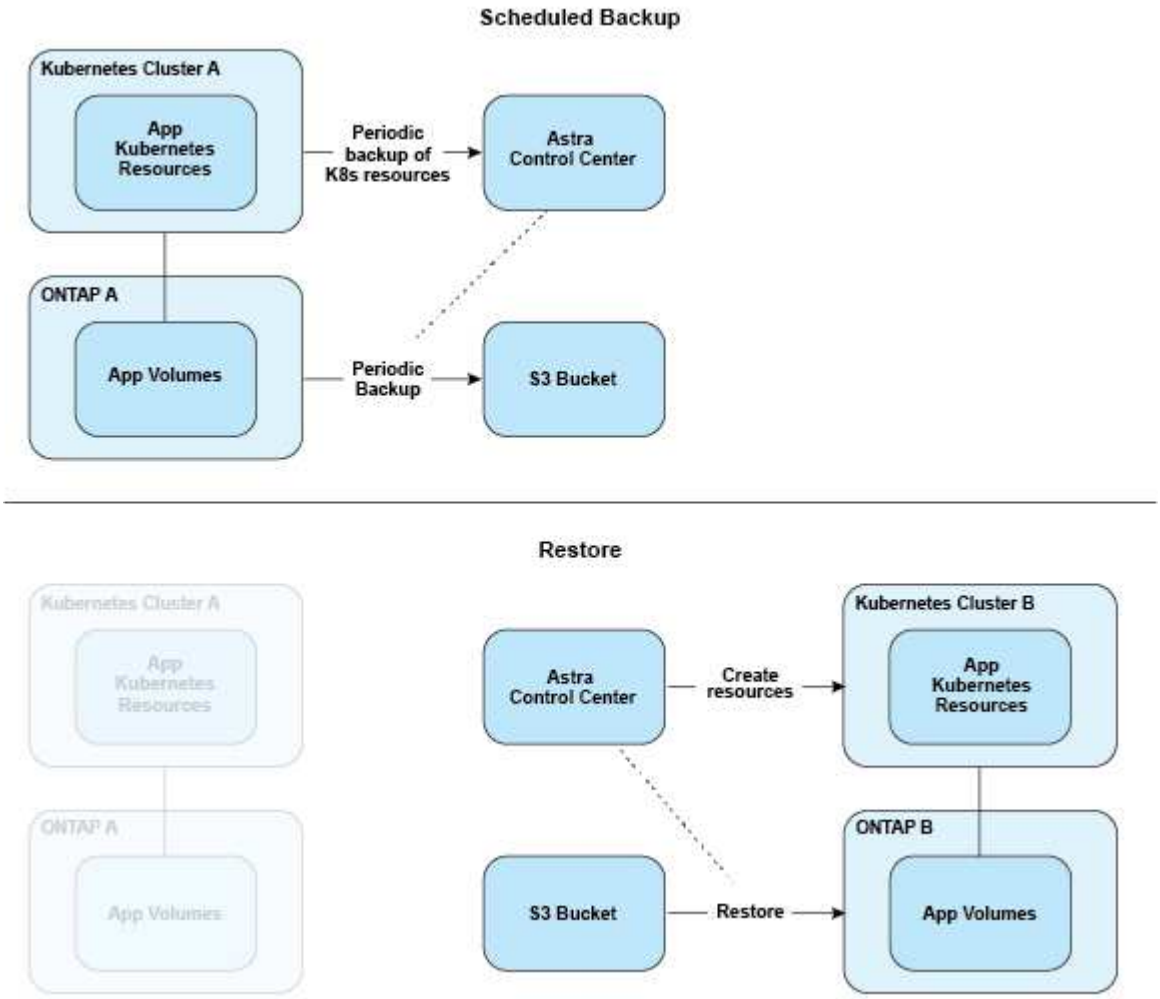
- **App replication**: Astra Control requires the source and destination Kubernetes clusters (which can be the same cluster) to be available and managed with their respective ONTAP storage backends configured to enable NetApp SnapMirror. Astra Control takes the policy-driven application snapshot and replicates it to the destination storage backend. NetApp SnapMirror technology is used to replicate the persistent volume data. To fail over, Astra Control can bring the replicated app online by recreating the app objects on the destination Kubernetes cluster with the replicated volumes on the destination ONTAP cluster. Because the persistent volume data is already present on the destination ONTAP cluster, Astra Control can offer quick recovery times for failover.

- **App backup and restore**: When backing up applications, Astra Control creates a snapshot of the app data and stores it in an object storage bucket. When a restore is needed, the data in the bucket must be copied to a persistent volume on the ONTAP cluster. The backup/restore operation does not require the secondary Kubernetes/ONTAP cluster to be available and managed, but the additional data copy can result in longer restore times.

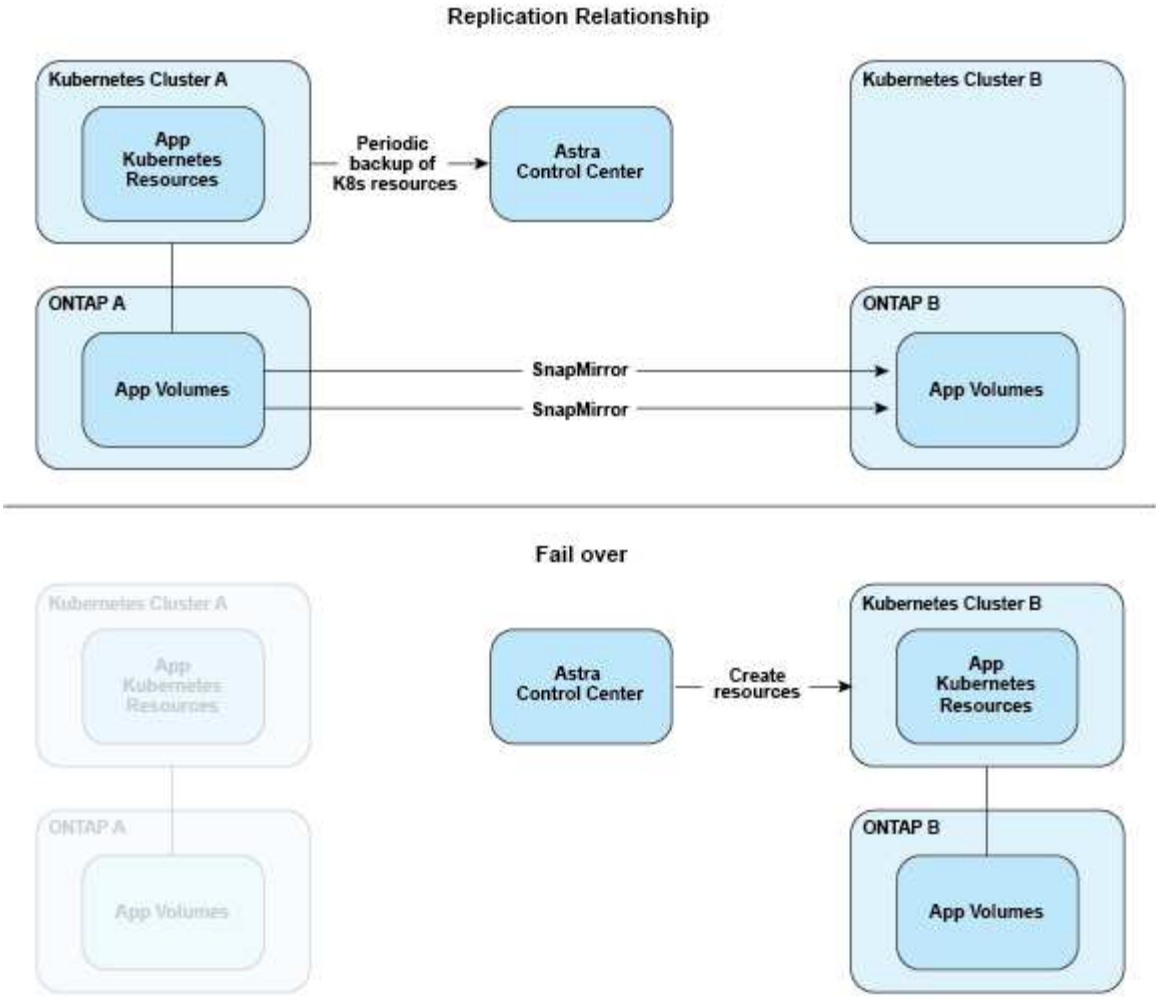
To learn how to replicate apps, refer to Replicate apps to a remote system using SnapMirror technology.

The following images show the scheduled backup and restore process compared to the replication process.

The backup process copies data to S3 buckets and restores from S3 buckets:



On the other hand, replication is done by replicating to ONTAP and then a failover creates the Kubernetes resources:



Backups, snapshots, and clones with an expired license

If your license expires, you can add a new application or perform application protection operations (such as snapshots, backups, clones, and restore operations) only if the application you are adding or protecting is another Astra Control Center instance.

Licensing

When you deploy Astra Control Center, it is installed with an embedded 90-day evaluation license for 4,800 CPU units. If you need more capacity or a longer evaluation period, or want to upgrade to a full license, you can obtain a different evaluation license or full license from NetApp.

You obtain a license in one of the following ways:

- If you are evaluating Astra Control Center and need different evaluation terms than what is included in the embedded evaluation license, contact NetApp to request a different evaluation license file.
- If you already purchased Astra Control Center, generate your NetApp license file (NLF) by logging in to the NetApp Support Site and navigating to your software licenses under the Systems menu.

For details about licenses needed for ONTAP storage backends, refer to supported storage backends.

Make sure that your license enables at least as many CPU units as you need. If the number of CPU units that Astra Control Center is currently managing exceeds the available CPU units in the new license being applied, you will not be able to apply the new license.

Evaluation licenses and full licenses

An embedded evaluation license is provided with a new Astra Control Center installation. An evaluation license enables the same capabilities and features as a full license for a limited (90 day) period. After the evaluation period, a full license is required to continue with full functionality.

License expiration

If the active Astra Control Center license expires, UI and API functionality for the following features are unavailable:

- Manual local snapshots and backups
- Scheduled local snapshots and backups
- Restoring from a snapshot or backup
- Cloning from a snapshot or current state
- Managing new applications
- Configuring replication policies

How license consumption is calculated

When you add a new cluster to Astra Control Center, it doesn't count toward consumed licenses until at least one application running on the cluster is managed by Astra Control Center.

When you start managing an app on a cluster, all of that cluster's CPU units are included in the Astra Control Center license consumption, except Red Hat OpenShift cluster node CPU units that are reported by a using the label `node-role.kubernetes.io/infra: ""`.

Red Hat OpenShift infrastructure nodes do not consume licenses in Astra Control Center. To mark a node as an infrastructure node, apply the label `node-role.kubernetes.io/infra: ""` to the node.

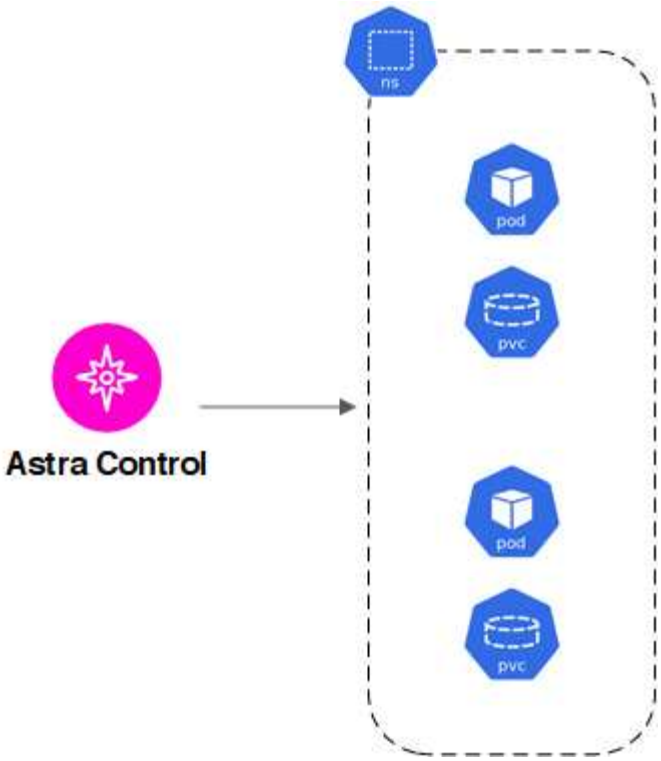
Find more information

- Add a license when you first set up Astra Control Center
- Update an existing license

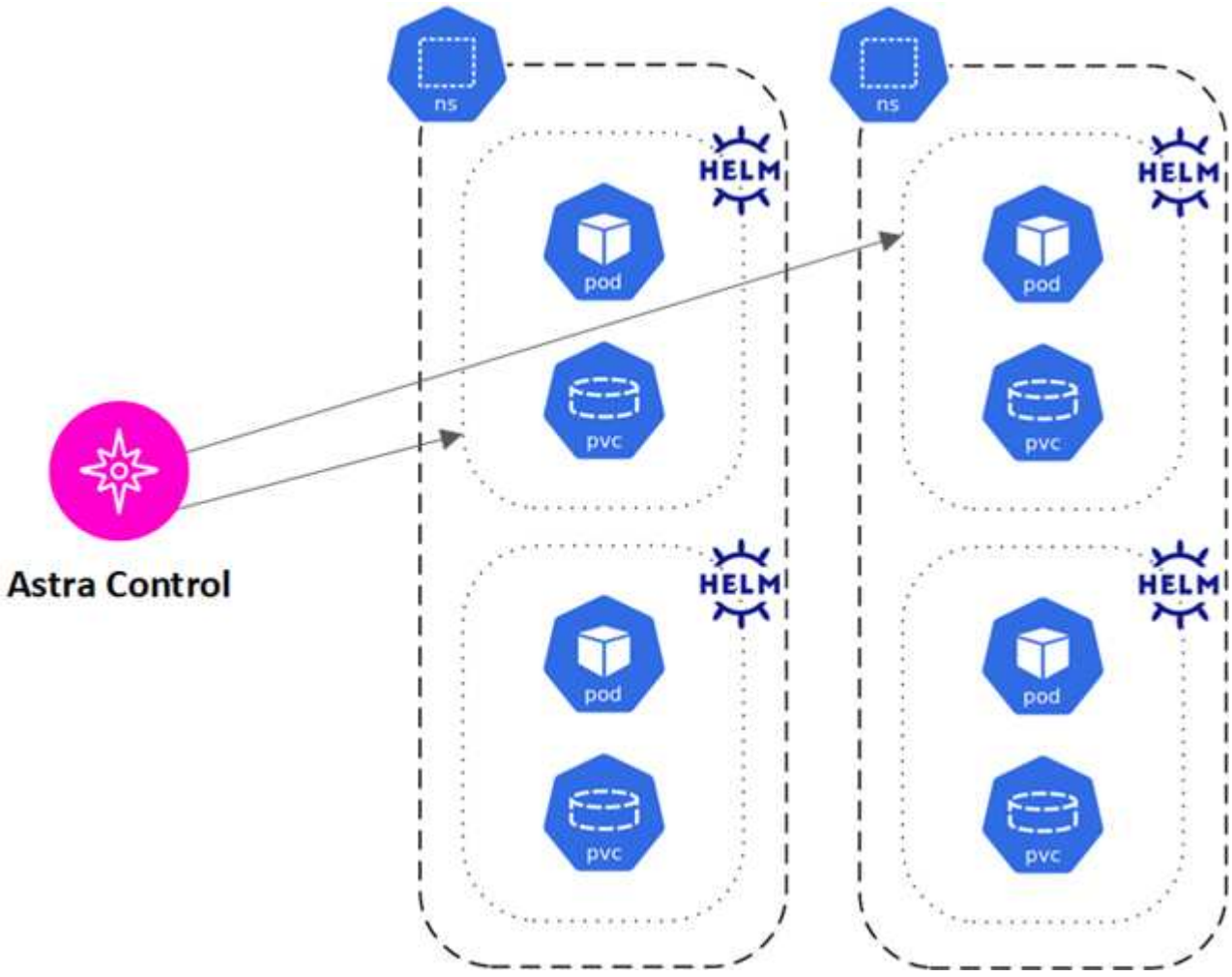
App management

When Astra Control discovers your clusters, the apps on those clusters are unmanaged until you choose how you want to manage them. A managed application in Astra Control can be any of the following:

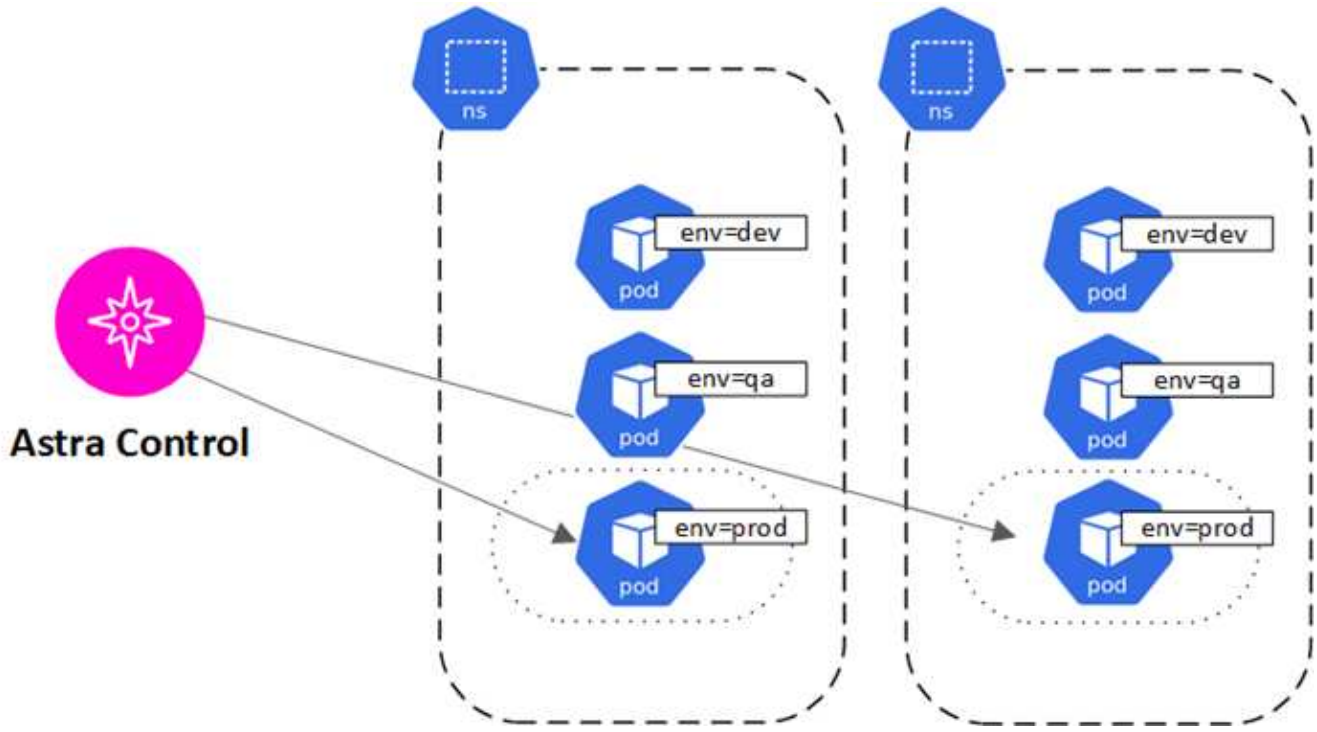
- A namespace, including all resources in that namespace



- An individual application deployed within one or more namespaces (helm3 is used in this example)



- A group of resources that are identified by a Kubernetes label within one or more namespaces



Storage classes and persistent volume size

Astra Control Center supports NetApp ONTAP and Longhorn as storage backends.

Overview

Astra Control Center supports the following:

- **Astra Trident storage classes backed by ONTAP storage**: If you are using an ONTAP backend, Astra Control Center offers the ability to import the ONTAP backend to report various monitoring information.
- **CSI-based storage classes backed by Longhorn**: You can use Longhorn with the Longhorn Container Storage Interface (CSI) driver.

Astra Trident storage classes should be preconfigured outside of Astra Control Center.

Storage classes

When you add a cluster to Astra Control Center, you're prompted to select one previously configured storage class on that cluster as the default storage class. This storage class will be used when no storage class is specified in a persistent volume claim (PVC). The default storage class can be changed at any time within Astra Control Center and any storage class can be used at any time by specifying the name of the storage class within the PVC or Helm chart. Ensure that you have only a single default storage class defined for your Kubernetes cluster.

For more information

- Astra Trident documentation

User roles and namespaces

Learn about user roles and namespaces in Astra Control, and how you can use them to control access to resources in your organization.

User roles

You can use roles to control the access users have to resources or capabilities of Astra Control. The following are the user roles in Astra Control:

- A **Viewer** can view resources.
- A **Member** has Viewer role permissions and can manage apps and clusters, unmanage apps, and delete snapshots and backups.
- An **Admin** has Member role permissions and can add and remove any other users except the Owner.
- An **Owner** has Admin role permissions and can add and remove any user accounts.

You can add constraints to a Member or Viewer user to restrict the user to one or more Namespaces.

Namespaces

A namespace is a scope you can assign to specific resources within a cluster that is managed by Astra Control. Astra Control discovers a cluster's namespaces when you add the cluster to Astra Control. Once discovered, the namespaces are available to assign as constraints to users. Only members that have access to that namespace are able to use that resource. You can use namespaces to control access to resources using a paradigm that makes sense for your organization; for example, by physical regions or divisions within a company. When you add constraints to a user, you can configure that user to have access to all namespaces or only a specific set of namespaces. You can also assign namespace constraints using namespace labels.

Find more information

Manage local users and roles

Pod security

Astra Control Center supports privilege limitation through pod security policies (PSPs) and pod security admission (PSA). These frameworks enable you to limit what users or groups are able to run containers and what privileges those containers can have.

Some Kubernetes distributions might have a default pod security configuration that is too restrictive, and causes problems when installing Astra Control Center.

You can use the information and examples included here to understand the pod security changes that Astra Control Center makes, and use a pod security approach that provides the protection you need without interfering with Astra Control Center functions.

PSAs enforced by Astra Control Center

Astra Control Center enables the enforcement of a pod security admission by adding the following label to the namespace where Astra is installed (netapp-acc or custom namespace) and namespaces created for backups.

```
pod-security.kubernetes.io/enforce: privileged
```

PSPs installed by Astra Control Center

When you install Astra Control Center on Kubernetes 1.23 or 1.24, several pod security policies are created during installation. Some of these are permanent, and some of them are created during certain operations and are removed once the operation is complete. Astra Control Center does not attempt to install PSPs when the host cluster is running Kubernetes 1.25 or later, as they are not supported on these versions.

PSPs created during installation

During Astra Control Center installation, the Astra Control Center operator installs a custom pod security policy, a `Role` object, and a `RoleBinding` object to support the deployment of Astra Control Center services in the Astra Control Center namespace.

The new policy and objects have the following attributes:

```
kubectl get psp

NAME                              PRIV    CAPS          SELINUX    RUNASUSER
FSGROUP     SUPGROUP    READONLYROOTFS   VOLUMES
netapp-astra-deployment-psp    false                 RunAsAny   RunAsAny
RunAsAny    RunAsAny    false            *

kubectl get role -n <namespace_name>

NAME                                      CREATED AT
netapp-astra-deployment-role              2022-06-27T19:34:58Z

kubectl get rolebinding -n <namespace_name>

NAME                                      ROLE
AGE
netapp-astra-deployment-rb                Role/netapp-astra-deployment-role
32m
```

PSPs created during backup operations

During backup operations, Astra Control Center creates a dynamic pod security policy, a `ClusterRole` object, and a `RoleBinding` object. These support the backup process, which happens in a separate namespace.

The new policy and objects have the following attributes:

```
kubectl get psp

NAME                            PRIV    CAPS
SELINUX    RUNASUSER          FSGROUP     SUPGROUP    READONLYROOTFS
VOLUMES
netapp-astra-backup             false   DAC_READ_SEARCH
RunAsAny   RunAsAny           RunAsAny    RunAsAny    false           *

kubectl get role -n <namespace_name>

NAME                   CREATED AT
netapp-astra-backup    2022-07-21T00:00:00Z

kubectl get rolebinding -n <namespace_name>

NAME                   ROLE                         AGE
netapp-astra-backup    Role/netapp-astra-backup     62s
```

PSPs created during cluster management

When you manage a cluster, Astra Control Center installs the netapp-monitoring operator in the managed cluster. This operator creates a pod security policy, a `ClusterRole` object, and a `RoleBinding` object to deploy telemetry services in the Astra Control Center namespace.

The new policy and objects have the following attributes:

```
kubectl get psp

NAME                            PRIV    CAPS
SELINUX    RUNASUSER          FSGROUP     SUPGROUP    READONLYROOTFS
VOLUMES
netapp-monitoring-psp-nkmo      true    AUDIT_WRITE,NET_ADMIN,NET_RAW
RunAsAny   RunAsAny           RunAsAny    RunAsAny    false           *

kubectl get role -n <namespace_name>

NAME                                          CREATED AT
netapp-monitoring-role-privileged             2022-07-21T00:00:00Z

kubectl get rolebinding -n <namespace_name>

NAME                                                ROLE
AGE
netapp-monitoring-role-binding-privileged           Role/netapp-
monitoring-role-privileged             2m5s
```

Copyright information

Trademark information