Use Astra Control Service

Astra Control Service

NetApp
April 17, 2024

Table of Contents

Use Astra Control Service

Log in to Astra Control Service

Astra Control Service is accessible through a SaaS-based user interface by going to https://astra.netapp.io.

> 💡 You can use single sign-on to log in using credentials from your corporate directory (federated identity). To learn more, go to the Help Center and then select **Cloud Central sign-in options**.

Before you begin

- A BlueXP user ID.
- A new Astra Control account or an invitation to an existing account.
- A supported web browser.

  Astra Control Service supports recent versions of Firefox, Safari, and Chrome with a minimum resolution of 1280 x 720.

Steps

1. Open a web browser and go to https://astra.netapp.io.
2. Log in using your NetApp BlueXP credentials.

Manage and protect apps

Start managing apps

After you add a Kubernetes cluster to Astra Control, you can install apps on the cluster (outside of Astra Control), and then go to the Applications page in Astra Control to define the apps.

You can define and manage apps that include storage resources with running pods, or apps that include storage resources without any running pods. Apps that have no running pods are known as data-only applications.

App management requirements

Astra Control has the following app management requirements:

- **Licensing**: To manage more than 10 namespaces, you need an Astra Control subscription.
- **Namespaces**: Apps can be defined within one or more specified namespaces on a single cluster using Astra Control. An app can contain resources spanning multiple namespaces within the same cluster. Astra Control does not support the ability for apps to be defined across multiple clusters.
- **Storage class**: If you install an app with a storage class explicitly set and you need to clone the app, the target cluster for the clone operation must have the originally specified storage class. Cloning an application with an explicitly set storage class to a cluster that does not have the same storage class will fail.
- **Kubernetes resources**: Apps that use Kubernetes Resources not collected by Astra Control might not

have full app data management capabilities. Astra Control collects the following Kubernetes resources:

`ClusterRole`	`ClusterRoleBinding`	`ConfigMap`
`CronJob`	`CustomResourceDefinition`	`CustomResource`
`DaemonSet`	`DeploymentConfig`	`HorizontalPodAutoscaler`
`Ingress`	`MutatingWebhook`	`NetworkPolicy`
`PersistentVolumeClaim`	`Pod`	`PodDisruptionBudget`
`PodTemplate`	`ReplicaSet`	`Role`
`RoleBinding`	`Route`	`Secret`
`Service`	`ServiceAccount`	`StatefulSet`
`ValidatingWebhook`		

Supported app installation methods

Astra Control supports the following application installation methods:

- **Manifest file**: Astra Control supports apps installed from a manifest file using kubectl. For example:

  ```
  kubectl apply -f myapp.yaml
  ```

- **Helm 3**: If you use Helm to install apps, Astra Control requires Helm version 3. Managing and cloning apps installed with Helm 3 (or upgraded from Helm 2 to Helm 3) are fully supported. Managing apps installed with Helm 2 is not supported.
- **Operator-deployed apps**: Astra Control supports apps installed with namespace-scoped operators that are, in general, designed with a "pass-by-value" rather than "pass-by-reference" architecture. An operator and the app it installs must use the same namespace; you might need to modify the deployment .yaml file for the operator to ensure this is the case.

  The following are some operator apps that follow these patterns:

  - Apache K8ssandra

    > ⓘ For K8ssandra, in-place restore operations are supported. A restore operation to a new namespace or cluster requires that the original instance of the application to be taken down. This is to ensure that the peer group information carried over does not lead to cross-instance communication. Cloning of the app is not supported.

  - Jenkins CI
  - Percona XtraDB Cluster

  Astra Control might not be able to clone an operator that is designed with a “pass-by-reference” architecture (for example, the CockroachDB operator). During these types of cloning operations, the cloned operator attempts to reference Kubernetes secrets from the source operator despite having its own new secret as part of the cloning process. The clone operation might fail because Astra Control is unaware of the Kubernetes secrets in the source operator.

Install apps on your cluster

After you’ve added your cluster to Astra Control, you can install apps or manage existing apps on the cluster. Any app that is scoped to one or more namespaces can be managed.

Astra Control will manage stateful apps only if the storage is on a storage class supported by Astra Control. Astra Control Service supports any storage class that is supported by Astra Control Provisioner or a generic CSI driver.

- Learn about storage classes for GKE clusters
- Learn about storage classes for AKS clusters
- Learn about storage classes for AWS clusters

Define apps

After Astra Control discovers namespaces on your clusters, you can define applications that you want to manage. You can choose to manage an app spanning one or more namespaces or manage an entire namespace as a single application. It all comes down to the level of granularity that you need for data protection operations.

Although Astra Control enables you to separately manage both levels of the hierarchy (the namespace and the apps in that namespace or spanning namespaces), the best practice is to choose one or the other. Actions that you take in Astra Control can fail if the actions take place at the same time at both the namespace and app level.

> As an example, you might want to set a backup policy for "maria" that has a weekly cadence, but you might need to back up "mariadb" (which is in the same namespace) more frequently than that. Based on those needs, you would need to manage the apps separately and not as a single-namespace app.

Before you begin

- A Kubernetes cluster added to Astra Control.
- One or more installed apps on the cluster. Read more about supported app installation methods.
- Existing namespaces on the Kubernetes cluster that you added to Astra Control.
- (Optional) A Kubernetes label on any supported Kubernetes resources.

> A label is a key/value pair you can assign to Kubernetes objects for identification. Labels make it easier to sort, organize, and find your Kubernetes objects. To learn more about Kubernetes labels, refer to the official Kubernetes documentation.

About this task

- Before you begin, you should also understand managing standard and system namespaces.
- If you plan to use multiple namespaces with your apps in Astra Control, consider modifying user roles with namespace constraints before defining apps.
- For instructions on how to manage apps using the Astra Control API, refer to the Astra Automation and API information.

App management options

- Define resources to manage as an app

- Define a namespace to manage as an app

Define resources to manage as an app

You can specify the Kubernetes resources that make up an app that you want to manage with Astra Control. Defining an app enables you to group elements of your Kubernetes cluster into a single app. This collection of Kubernetes resources is organized by namespace and label selector criteria.

Defining an app gives you more granular control over what to include in an Astra Control operation, including clone, snapshot, and backups.

> ! When defining apps, ensure that you do not include a Kubernetes resource in multiple apps with protection policies. Overlapping protection policies on a Kubernetes resources can cause data conflicts.

Read more about adding cluster-scoped resources to your app namespaces.

You can import cluster resources that are associated with the namespace resources in addition to those Astra Control included automatically. You can add a rule that will include resources of a specific group, kind, version and optionally, label. You might want to do this if there are resources that Astra Control does not include automatically.

You cannot exclude any of the cluster-scoped resources that are automatically included by Astra Control.

You can add the following `apiVersions` (which are the groups combined with the API version):

Resource kind	apiVersions (group + version)
`ClusterRole`	rbac.authorization.k8s.io/v1
`ClusterRoleBinding`	rbac.authorization.k8s.io/v1
`CustomResource`	apiextensions.k8s.io/v1, apiextensions.k8s.io/v1beta1
`CustomResourceDefinition`	apiextensions.k8s.io/v1, apiextensions.k8s.io/v1beta1
`MutatingWebhookConfiguration`	admissionregistration.k8s.io/v1
`ValidatingWebhookConfiguration`	admissionregistration.k8s.io/v1

Steps

1. From the Applications page, select **Define**.
2. In the **Define application** window, enter the app name.
3. Choose the cluster on which your application is running in the **Cluster** drop-down list.
4. Choose a namespace for your application from the **Namespace** drop-down list.

   > i Apps can be defined within one or more specified namespaces on a single cluster using Astra Control. An app can contain resources spanning multiple namespaces within the same cluster. Astra Control does not support the ability for apps to be defined across multiple clusters.

5. (Optional) Enter a label for the Kubernetes resources in each namespace. You can specify a single label or label selector criteria (query).

   > To learn more about Kubernetes labels, refer to the official Kubernetes documentation.

6. (Optional) Add additional namespaces for the app by selecting **Add namespace** and choosing the namespace from the drop-down list.
7. (Optional) Enter single label or label selector criteria for any additional namespaces you add.
8. (Optional) To include cluster-scoped resources in addition to those that Astra Control automatically includes, check **Include additional cluster-scoped resources** and complete the following:
   a. Select **Add include rule**.
   b. **Group**: From the drop-down list, select the API group of resources.
   c. **Kind**: From the drop-down list, select the name of the object schema.
   d. **Version**: Enter the API version.
   e. **Label selector**: Optionally, include a label to add to the rule. This label is used to retrieve only those resources matching this label. If you don't provide a label, Astra Control collects all instances of the resource kind specified for that cluster.
   f. Review the rule that is created based on your entries.
   g. Select **Add**.

      > You can create as many cluster-scoped resource rules as you want. The rules appear in the Define application Summary.

9. Select **Define**.
10. After you select **Define**, repeat the process for other apps, as needed.

After you finish defining an app, the app appears in `Healthy` state in the list of apps on the Applications page. You are now able to clone it and create backups and snapshots.

> The app you just added might have a warning icon under the Protected column, indicating that it is not backed up and not scheduled for backups yet.

> To see details of a particular app, select the app name.

To see the resources added to this app, select the **Resources** tab. Select the number after the resource name in the Resource column or enter the resource name in Search to see the additional cluster-scoped resources included.

Define a namespace to manage as an app

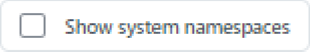
You can add all Kubernetes resources in a namespace to Astra Control management by defining the resources of that namespace as an application. This method is preferable to defining apps individually if you intend to manage and protect all resources in a particular namespace in a similar way and at common intervals.

Steps

1. From the Clusters page, select a cluster.
2. Select the **Namespaces** tab.

3. Select the Actions menu for the namespace that contains the app resources you want to manage and select **Define as application**.

> If you want to define multiple applications, select from the namespaces list and select the **Actions** button in the upper-left corner and select **Define as application**. This will define multiple individual applications in their individual namespaces. For multi-namespace applications, refer to Define resources to manage as an app.

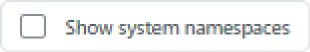
> Select the **Show system namespaces** checkbox to reveal system namespaces that are usually not used in app management by default.  □ Show system namespaces Read more.

After the process completes, the applications that are associated with the namespace appear in the `Associated applications` column.

What about system namespaces?

Astra Control also discovers system namespaces on a Kubernetes cluster. We don't show you these system namespaces by default because it's rare that you'd need to back up system app resources.

You can display system namespaces from the Namespaces tab for a selected cluster by selecting the **Show system namespaces** check box.



□ Show system namespaces

> Astra Control itself is not a standard app; it is a "system app." You should not try to manage Astra Control itself. Astra Control itself isn't shown by default for management.

Protect apps with snapshots and backups

Protect your apps by taking snapshots and backups using an automated protection policy or on an ad-hoc basis. You can use the Astra UI or the Astra Control API to protect apps.

Learn more about data protection in Astra Control.

You can do the following tasks related to protecting your app data:

- Configure a protection policy
- Create a snapshot
- Create a backup
- Enable backup and restore for ontap-nas-economy operations
- Create an immutable backup
- View snapshots and backups
- Delete snapshots
- Cancel backups
- Delete backups

Configure a protection policy

A protection policy protects an app by creating snapshots, backups, or both at a defined schedule. You can choose to create snapshots and backups hourly, daily, weekly, and monthly, and you can specify the number of copies to retain.

If you need backups or snapshots to run more frequently than once per hour, you can use the Astra Control REST API to create snapshots and backups.

> If you are defining a protection policy that creates immutable backups to write once read many (WORM) buckets, ensure that the retention time for the backups is not shorter than the retention period configured for the bucket.

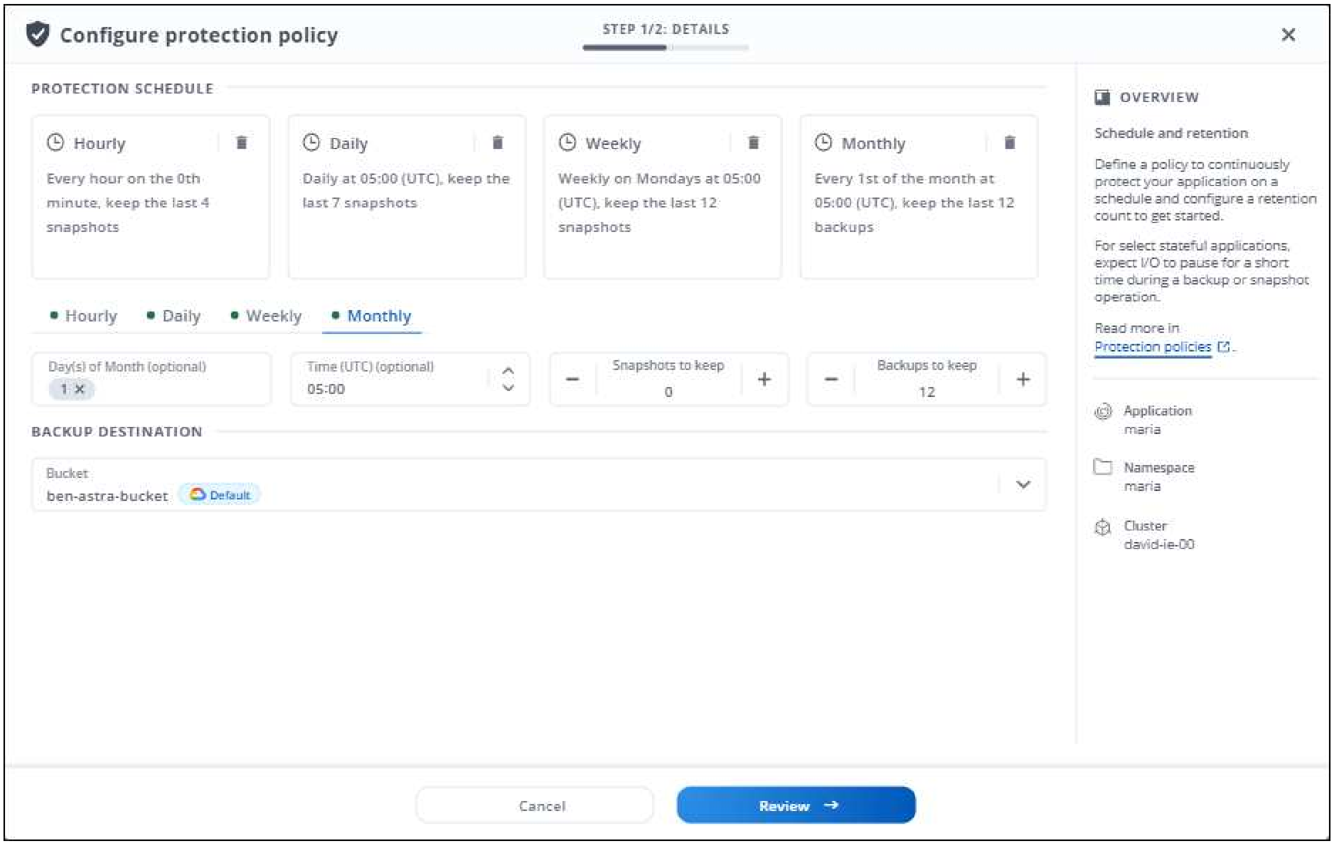
Steps

1. Select **Applications** and then select the name of a managed app.
2. Select **Data Protection**.
3. Select **Configure Protection Policy**.
4. Define a protection schedule by choosing the number of snapshots and backups to keep for the hourly, daily, weekly, and monthly schedules.

   You can define the hourly, daily, weekly, and monthly schedules concurrently. A schedule won't turn active until you set a retention level for snapshots and backups.

   When you set a retention level for backups, you can choose the bucket where you'd like to store the backups.

   The following example sets four protection schedules: hourly, daily, weekly, and monthly for snapshots and backups.



Hourly: Every hour on the 0th minute, keep the last 4 snapshots

Daily: Daily at 05:00 (UTC), keep the last 7 snapshots

Weekly: Weekly on Mondays at 05:00 (UTC), keep the last 12 snapshots

Monthly: Every 1st of the month at 05:00 (UTC), keep the last 12 backups

Day(s) of Month (optional): 1

Time (UTC) (optional): 05:00

Snapshots to keep: 0

Backups to keep: 12

Bucket: ben-astra-bucket (Default)

Schedule and retention

Define a policy to continuously protect your application on a schedule and configure a retention count to get started.

For select stateful applications, expect I/O to pause for a short time during a backup or snapshot operation.

Read more in Protection policies.

Application: maria

Namespace: maria

Cluster: david-ie-00

5. Select **Review**.
6. Select **Set Protection Policy**.

Result

Astra Control implements the data protection policy by creating and retaining snapshots and backups using the schedule and retention policy that you defined.

Create a snapshot

You can create an on-demand snapshot at any time.

About this task

Astra Control supports snapshot creation using storage classes backed by the following drivers:

- `ontap-nas`
- `ontap-san`
- `ontap-san-economy`

ⓘ If your app uses a storage class backed by the `ontap-nas-economy` driver, snapshots can’t be created. Use an alternate storage class for snapshots.

Steps

1. Select **Applications**.
2. From the Options menu in the **Actions** column for the desired app, select **Snapshot**.
3. Customize the name of the snapshot and then select **Next**.

4. Review the snapshot summary and select **Snapshot**.

Result

The snapshot process begins. A snapshot is successful when the status is **Healthy** in the **State** column on the **Data protection** > **Snapshots** page.

Create a backup

You can also back up an app at any time.

> ⓘ Be aware of how storage space is handled when you back up an application hosted on Azure NetApp Files storage. Refer to Application backups for more information.

> ⓘ Astra Control supports backup creation using storage classes backed by the following drivers:
>
> - `ontap-nas`
> - `ontap-nas-economy`
> - `ontap-san`
> - `ontap-san-economy`

About this task

Buckets in Astra Control do not report available capacity. Before backing up or cloning apps managed by Astra Control, check bucket information in the appropriate storage management system.

If your app uses a storage class backed by the `ontap-nas-economy` driver, you need to enable backup and restore functionality. Be sure that you have defined a `backendType` parameter in your Kubernetes storage object with a value of `ontap-nas-economy` before performing any protection operations.

Steps

1. Select **Applications**.
2. From the Options menu in the **Actions** column for the desired app, select **Back up**.
3. Customize the name of the backup.
4. Choose whether to back up the app from an existing snapshot. If you select this option, you can choose from a list of existing snapshots.
5. Choose a destination bucket for the backup from the list of storage buckets.
6. Select **Next**.
7. Review the backup summary and select **Back up**.

Result

Astra Control creates a backup of the app.

ⓘ

- If your network has an outage or is abnormally slow, a backup operation might time out. This causes the backup to fail.
- If you need to cancel a running backup, use the instructions in Cancel backups. To delete the backup, wait until it has completed and then use the instructions in Delete backups.
- After a data protection operation (clone, backup, restore) and subsequent persistent volume resize, there is up to a twenty-minute delay before the new volume size is shown in the UI. The data protection operation is successful within minutes, and you can use the management software for the storage backend to confirm the change in volume size.

Enable backup and restore for ontap-nas-economy operations

Astra Control Provisioner provides backup and restore functionality that can be enabled for storage backends that are using the `ontap-nas-economy` storage class.

Before you begin

- You have enabled Astra Control Provisioner or Astra Trident.
- You have defined an application in Astra Control. This application will have limited protection functionality until you complete this procedure.
- You have `ontap-nas-economy` selected as the default storage class for your storage backend.

Expand for configuration steps

1. Do the following on the ONTAP storage backend:
   a. Find the SVM that is hosting the `ontap-nas-economy`-based volumes of the application.
   b. Log in to a terminal connected to ONTAP where the volumes are created.
   c. Hide the snapshot directory for the SVM:

      > This change affects the entire SVM. The hidden directory will continue to be accessible.

      ```
      nfs modify -vserver <svm name> -v3-hide-snapshot enabled
      ```

      > Verify that the snapshot directory on the ONTAP storage backend is hidden. Failure to hide this directory might lead to loss of access to your application, especially if it is using NFSv3.

2. Do the following in Astra Control Provisioner or Astra Trident:
   a. Enable the snapshot directory for each PV that is ontap-nas-economy based and associated with the application:

      ```
      tridentctl update volume <pv name> --snapshot-dir=true --pool
      -level=true -n trident
      ```

   b. Confirm that the snapshot directory has been enabled for each associated PV:

      ```
      tridentctl get volume <pv name> -n trident -o yaml | grep
      snapshotDir
      ```

      Response:

      ```
      snapshotDirectory: "true"
      ```

3. In Astra Control, refresh the application after enabling all associated snapshot directories so that Astra Control recognizes the changed value.

Result

The application is ready to backup and restore using Astra Control. Each PVC is also available to be used by other applications for backups and restores.

Create an immutable backup

An immutable backup cannot be modified, deleted, or overwritten as long as the retention policy on the bucket that stores the backup forbids it. You can create immutable backups by backing up applications to buckets that have a retention policy configured. Refer to Data protection for important information about working with immutable backups.

Before you begin

You need to configure the destination bucket with a retention policy. How you do this will differ depending on which storage provider you use. Refer to the storage provider documentation for more information:

- **Amazon Web Services**: Enable S3 Object Lock when creating the bucket and set a default retention mode of "governance" with a default retention period.
- **Google Cloud**: Configure a bucket with a retention policy and specify a retention period.
- **Microsoft Azure**: Configure a blob storage bucket with a time-based retention policy on container-level scope.
- **NetApp StorageGRID**: Enable S3 Object Lock when creating the bucket and set a default retention mode of "compliance" with a default retention period.

> ⓘ Buckets in Astra Control do not report available capacity. Before backing up or cloning apps managed by Astra Control, check bucket information in the appropriate storage management system.

> ⓘ If your app uses a storage class backed by the `ontap-nas-economy` driver, be sure that you have defined a `backendType` parameter in your Kubernetes storage object with a value of `ontap-nas-economy` before performing any protection operations.

Steps

1. Select **Applications**.
2. From the Options menu in the **Actions** column for the desired app, select **Back up**.
3. Customize the name of the backup.
4. Choose whether to back up the app from an existing snapshot. If you select this option, you can choose from a list of existing snapshots.
5. Choose a destination bucket for the backup from the list of storage buckets. A write once read many (WORM) bucket is indicated with a status of "Locked" next to the bucket name.

   > ⓘ If the bucket is an unsupported type, this is indicated when you hover over or select the bucket.

6. Select **Next**.
7. Review the backup summary and select **Back up**.

Result

Astra Control creates an immutable backup of the app.

- If your network has an outage or is abnormally slow, a backup operation might time out. This causes the backup to fail.
- If you try to create two immutable backups of the same app to the same bucket at the same time, Astra Control prevents the second backup from starting. Wait until the first backup is complete before starting another.
- You cannot cancel a running immutable backup.
- After a data protection operation (clone, backup, restore) and subsequent persistent volume resize, there is up to a twenty-minute delay before the new volume size is shown in the UI. The data protection operation is successful within minutes, and you can use the management software for the storage backend to confirm the change in volume size.

View snapshots and backups

You can view the snapshots and backups of an app from the Data Protection tab.

An immutable backup is indicated with a status of "Locked" next to the bucket it is using.

Steps

1. Select **Applications** and then select the name of a managed app.
2. Select **Data Protection**.

   The snapshots display by default.

3. Select **Backups** to refer to the list of backups.

Delete snapshots

Delete the scheduled or on-demand snapshots that you no longer need.

Steps

1. Select **Applications** and then select the name of a managed app.
2. Select **Data Protection**.
3. From the Options menu in the **Actions** column for the desired snapshot, select **Delete snapshot**.
4. Type the word "delete" to confirm deletion and then select **Yes, Delete snapshot**.

Result

Astra Control deletes the snapshot.

Cancel backups

You can cancel a backup that is in progress.

To cancel a backup, the backup must be in `Running` state. You cannot cancel a backup that is in `Pending` state.

You cannot cancel a running immutable backup.

Steps

1. Select **Applications** and then select the name of an app.
2. Select **Data Protection**.
3. Select **Backups**.
4. From the Options menu in the **Actions** column for the desired backup, select **Cancel**.
5. Type the word "cancel" to confirm the operation and then select **Yes, cancel backup**.

Delete backups

Delete the scheduled or on-demand backups that you no longer need.

> ⓘ If you need to cancel a running backup, use the instructions in Cancel backups. To delete the backup, wait until it has completed and then use these instructions.

> ⓘ You cannot delete an immutable backup before the retention period expires.

Steps

1. Select **Applications** and then select the name of an app.
2. Select **Data Protection**.
3. Select **Backups**.
4. From the Options menu in the **Actions** column for the desired backup, select **Delete backup**.
5. Type the word "delete" to confirm deletion and then select **Yes, Delete backup**.

Result

Astra Control deletes the backup.

[Tech preview] Protect an entire cluster

You can create a scheduled, automatic backup of any or all unmanaged namespaces on a cluster. These workflows are provided by NetApp as a Kubernetes service account, role bindings, and a cron job, orchestrated with a Python script.

How it works

When you configure and install the full-cluster backup workflow, a cron job runs periodically and protects any namespace that is not already managed, automatically creating protection policies based on schedules that you choose during installation.

If you don’t want to protect every unmanaged namespace on the cluster with the full cluster backup workflow, you can instead utilize the label-based backup workflow. The label-based backup workflow also uses a cron task, but instead of protecting all unmanaged namespaces, it identifies namespaces by labels you provide to optionally protect the namespaces based on bronze, silver, or gold backup policies.

When a new namespace is created that falls within the scope of your chosen workflow, it is automatically protected, without any administrator action. These workflows are implemented on a per-cluster basis, so different clusters can make use of either workflow with unique protection levels, depending on cluster importance.

Example: Full cluster protection

As an example, when you configure and install the full cluster backup workflow, any apps in any namespace are periodically managed and protected without further effort by the administrator. The namespace doesn't have to exist at the time you install the workflow; if a namespace is added in the future, it will be protected.

Example: Label-based protection

For more granularity, you can use the label-based workflow. For example, you can install this workflow and tell your users to apply one of several labels to any namespaces they want to protect, depending on the level of protection they need. This enables users to create the namespace with one of those labels, and they don't have to notify an administrator. Their new namespace and all apps within it are automatically protected.

Create a scheduled backup of all namespaces

You can create a scheduled backup of all namespaces on a cluster using the full cluster backup workflow.

Steps

1. Download the following files to a machine that has network access to your cluster:
   - components.yaml CRD file
   - protectCluster.py Python script
2. To configure and install the toolkit, follow the included instructions.

Create a scheduled backup of specific namespaces

You can create a scheduled backup of specific namespaces by their labels using the label-based backup workflow.

Steps

1. Download the following files to a machine that has network access to your cluster:
   - components.yaml CRD file
   - protectCluster.py Python script
2. To configure and install the toolkit, follow the included instructions.

Restore apps

Astra Control can restore your application from a snapshot or backup. Restoring from an existing snapshot will be faster when restoring the application to the same cluster. You can use the Astra Control UI or the Astra Control API to restore apps.

> If you add a namespace filter to an execution hook that runs after a restore or clone operation and the restore or clone source and destination are in different namespaces, the namespace filter is only applied to the destination namespace.

Before you begin

- **Protect your apps first**: It is strongly recommended that you take a snapshot or backup of your application before restoring it. This will enable you to clone from the snapshot or backup if the restore is unsuccessful.
- **Check destination volumes**: If you restore to a different storage class, ensure that the storage class uses the same persistent volume access mode (for example, ReadWriteMany). The restore operation will fail if the destination persistent volume access mode is different. For example, if your source persistent volume uses the RWX access mode, selecting a destination storage class that is not able to provide RWX, such as

Azure Managed Disks, AWS EBS, Google Persistent Disk or `ontap-san`, will cause the restore operation to fail. For more information about persistent volume access modes, refer to the Kubernetes documentation.

- **Plan for space needs**: When you perform an in-place restore of an application that uses NetApp ONTAP storage, the space used by the restored app can double. After performing an in-place restore, remove any unwanted snapshots from the restored application to free up storage space.
- **Supported storage class drivers**: Astra Control supports restoring backups using storage classes backed by the following drivers:
  - `ontap-nas`
  - `ontap-nas-economy`
  - `ontap-san`
  - `ontap-san-economy`
- **(ontap-nas-economy driver only) Backups and restores**: Before backing up or restoring an app that uses a storage class backed by the `ontap-nas-economy` driver, verify that the snapshot directory on the ONTAP storage backend is hidden. Failure to hide this directory might lead to loss of access to your application, especially if it is using NFSv3.

> ! Performing an in-place restore operation on an app that shares resources with another app can have unintended results. Any resources that are shared between the apps are replaced when an in-place restore is performed on one of the apps.

Steps

1. Select **Applications** and then select the name of an app.
2. From the Options menu in the Actions column, select **Restore**.
3. Choose the restore type:
   - **Restore to original namespaces**: Use this procedure to restore the app in-place to the original cluster.
     a. Select the snapshot or backup to use to restore the app in-place, which reverts the app to an earlier version of itself.
     b. Select **Next**.

     > ⓘ If you restore to a namespace that was previously deleted, a new namespace with the same name is created as part of the restore process. Any users that had rights to manage apps in the previously deleted namespace need to manually restore rights to the newly re-created namespace.

   - **Restore to new namespaces**: Use this procedure to restore the app to another cluster or with different namespaces from the source. You can also use this procedure to migrate an app to a different storage class.
     a. Specify the name for the restored app.
     b. Choose the destination cluster for the app you intend to restore.
     c. Enter a destination namespace for each source namespace associated with the app.

     > ⓘ Astra Control creates new destination namespaces as part of this restore option. Destination namespaces that you specify must not be already present on the destination cluster.

   d. Select **Next**.

   e. Select the snapshot or backup to use to restore the app.

   f. Select **Next**.

   g. Choose one of the following:

      - **Restore using original storage classes**: The application uses the originally associated storage class unless it does not exist on the target cluster. In this case, the default storage class for the cluster will be used.

      - **Restore using a different storage class**: Select a storage class that exists on the target cluster. All application volumes, regardless of their originally associated storage classes, will be migrated to this different storage class as part of the restore.

   h. Select **Next**.

4. Choose any resources to filter:

   - **Restore all resources**: Restore all resources associated with the original app.

   - **Filter resources**: Specify rules to restore a sub-set of the original application resources:

     i. Choose to include or exclude resources from the restored application.

     ii. Select either **Add include rule** or **Add exclude rule** and configure the rule to filter the correct resources during application restore. You can edit a rule or remove it and create a rule again until the configuration is correct.

     > ⓘ To learn about configuring include and exclude rules, see Filter resources during an application restore.

5. Select **Next**.

6. Review details about the restore action carefully, type "restore" (if prompted), and select **Restore**.

Result

Astra Control restores the app based on the information that you provided. If you restored the app in-place, the content of existing persistent volumes is replaced with the content of persistent volumes from the restored app.

> ⓘ After a data protection operation (clone, backup, or restore) and subsequent persistent volume resize, there is a delay of up to twenty minutes before the new volume size is shown in the web UI. The data protection operation is successful within minutes, and you can use the management software for the storage backend to confirm the change in volume size.

> ⓘ Any member user with namespace constraints by namespace name/ID or by namespace labels can clone or restore an app to a new namespace on the same cluster or to any other cluster in their organization’s account. However, the same user cannot access the cloned or restored app in the new namespace. After a clone or restore operation creates a new namespace, the account admin/owner can edit the member user account and update role constraints for the affected user to grant access to the new namespace.

Filter resources during an application restore

You can add a filter rule to a restore operation that will specify existing application resources to be included or excluded from the restored application. You can include or exclude resources based on a specified namespace, label, or GVK (GroupVersionKind).

Read more about include and exclude scenarios

- **You select an include rule with original namespaces (in-place restore)**: Existing application resources that you define in the rule will be deleted and replaced by those from the selected snapshot or backup you are using for the restore. Any resources that you do not specify in the include rule will remain unchanged.
- **You select an include rule with new namespaces**: Use the rule to select the specific resources you want in the restored application. Any resources that you do not specify in the include rule will not be included in the restored application.
- **You select an exclude rule with original namespaces (in-place restore)**: The resources you specify to be excluded will not be restored and remain unchanged. Resources that you do not specify to exclude will be restored from the snapshot or backup. All data on persistent volumes will be deleted and recreated if the corresponding StatefulSet is part of the filtered resources.
- **You select an exclude rule with new namespaces**: Use the rule to select the specific resources you want to remove from the restored application. Resources that you do not specify to exclude will be restored from the snapshot or backup.

Rules are either include or exclude types. Rules combining resource inclusion and exclusion are not available.

Steps

1. After you have chosen to filter resources and selected an include or exclude option in the Restore App wizard, select **Add include rule** or **Add exclude rule**.

   ⓘ You cannot exclude any cluster-scoped resources that are automatically included by Astra Control.

2. Configure the filter rule:

   ⓘ You must specify at least one namespace, label, or GVK. Ensure that any resources you retain after the filter rules are applied are sufficient to keep the restored application in a healthy state.

   a. Select a specific namespace for the rule. If you don’t make a selection, all namespaces will be used in the filter.

      ⓘ If your application originally contained multiple namespaces and you restore it to new namespaces, all namespaces will be created even if they don’t contain resources.

   b. (Optional) Enter a resource name.

   c. (Optional) **Label selector**: Include a label selector to add to the rule. The label selector is used to filter only those resources matching the selected label.

   d. (Optional) Select **Use GVK (GroupVersionKind) set to filter resources** for additional filtering options.

      ⓘ If you use a GVK filter, you must specify Version and Kind.

      i. (Optional) **Group**: From the drop-down list, select the Kubernetes API group.

      ii. **Kind**: From the drop-down list, select the object schema for the Kubernetes resource type to use in the filter.

iii. **Version**: Select the Kubernetes API version.

3. Review the rule that is created based on your entries.
4. Select **Add**.

> You can create as many resource include and exclude rules as you want. The rules appear in the restore application summary before you initiate the operation.

Clone and migrate apps

You can clone an existing app to create a duplicate app on the same Kubernetes cluster or on another cluster. When Astra Control clones an app, it creates a clone of your application configuration and persistent storage.

Cloning can help if you need to move applications and storage from one Kubernetes cluster to another. For example, you might want to move workloads through a CI/CD pipeline and across Kubernetes namespaces.

> If you add a namespace filter to an execution hook that runs after a restore or clone operation and the restore or clone source and destination are in different namespaces, the namespace filter is only applied to the destination namespace.

Before you begin

- **Check destination volumes**: If you clone to a different storage class, ensure that the storage class uses the same persistent volume access mode (for example, ReadWriteMany). The clone operation will fail if the destination persistent volume access mode is different. For example, if your source persistent volume uses the RWX access mode, selecting a destination storage class that is not able to provide RWX, such as Azure Managed Disks, AWS EBS, Google Persistent Disk or `ontap-san`, will cause the clone operation to fail. For more information about persistent volume access modes, refer to the Kubernetes documentation.
- To clone apps to a different cluster, you need to make sure that you have assigned a default bucket for the cloud instance containing the source cluster. If the source cloud instance does not have a default bucket set, the cross-cluster clone operation will fail.
- During clone operations, apps that need an IngressClass resource or webhooks to function properly must not have those resources already defined on the destination cluster.

Clone limitations

- **Explicit storage classes**: If you deploy an app with a storage class explicitly set and you need to clone the app, the target cluster must have the originally specified storage class. Cloning an application with an explicitly set storage class to a cluster that does not have the same storage class will fail.
- **ontap-nas-economy-backed applications**: You can’t use clone operations if your application’s storage class is backed by the `ontap-nas-economy` driver. You can, however, enable backup and restore for ontap-nas-economy operations.
- **Clones and user constraints**: Any member user with namespace constraints by namespace name/ID or by namespace labels can clone or restore an app to a new namespace on the same cluster or to any other cluster in their organization’s account. However, the same user cannot access the cloned or restored app in the new namespace. After a clone or restore operation creates a new namespace, the account admin/owner can edit the member user account and update role constraints for the affected user to grant access to the new namespace.
- **Clones use default buckets**:

    - During an app backup or app restore, you can specify a bucket to use. You need to specify a default bucket when you clone across clusters, but specifying a bucket is optional when cloning within the same cluster.
    - When you clone across clusters, the cloud instance containing the source cluster of the clone operation must have a default bucket set.
    - There is no option to change buckets for a clone. If you want control over which bucket is used, you can either change the bucket default or do a backup followed by a restore separately.
- **With Jenkins CI**: If you clone an operator-deployed instance of Jenkins CI, you need to manually restore the persistent data. This is a limitation of the app's deployment model.

Steps

1. Select **Applications**.
2. Do one of the following:
    - Select the Options menu in the **Actions** column for the desired app.
    - Select the name of the desired app, and select the status drop-down list at the top right of the page.
3. Select **Clone**.
4. Specify details for the clone:
    - Enter a name.
    - Choose a destination cluster for the clone.
    - Enter destination namespaces for the clone. Each source namespace associated with the app maps to a destination namespace.

      ⓘ Astra Control creates new destination namespaces as part of the clone operation. Destination namespaces that you specify must not be already present on the destination cluster.

    - Select **Next**.
    - Choose to keep the original storage class associated with the app or select a different storage class.

      ⓘ You can migrate an app's storage class to a native cloud provider storage class or other supported storage class, migrate an app from a storage class backed by `ontap-nas-economy` to a storage class backed by `ontap-nas` on the same cluster, or copy the app to another cluster with a storage class backed by the `ontap-nas-economy` driver.

      ⓘ If you select a different storage class and this storage class doesn't exist at the moment of restore, an error will be returned.

5. Select **Next**.
6. Review the information about the clone and select **Clone**.

Result

Astra Control clones the app based on the information that you provided. The clone operation is successful when the new app clone is in `Healthy` state on the **Applications** page.

After a clone or restore operation creates a new namespace, the account admin/owner can edit the member user account and update role constraints for the affected user to grant access to the new namespace.

Manage app execution hooks

An execution hook is a custom action that you can configure to run in conjunction with a data protection operation of a managed app. For example, if you have a database app, you can use an execution hook to pause all database transactions before a snapshot, and resume transactions after the snapshot is complete. This ensures application-consistent snapshots.

Types of execution hooks

Astra Control Service supports the following types of execution hooks, based on when they can be run:

- Pre-snapshot
- Post-snapshot
- Pre-backup
- Post-backup
- Post-restore

Execution hook filters

When you add or edit an execution hook to an application, you can add filters to an execution hook to manage which containers the hook will match. Filters are useful for applications that use the same container image on all containers, but might use each image for a different purpose (such as Elasticsearch). Filters allow you to create scenarios where execution hooks run on some but not necessarily all identical containers. If you create multiple filters for a single execution hook, they are combined with a logical AND operator. You can have up to 10 active filters per execution hook.

Each filter you add to an execution hook uses a regular expression to match containers in your cluster. When a hook matches a container, the hook will run its associated script on that container. Regular expressions for filters use the Regular Expression 2 (RE2) syntax, which does not support creating a filter that excludes containers from the list of matches. For information on the syntax that Astra Control supports for regular expressions in execution hook filters, see Regular Expression 2 (RE2) syntax support.

> ⓘ If you add a namespace filter to an execution hook that runs after a restore or clone operation and the restore or clone source and destination are in different namespaces, the namespace filter is only applied to the destination namespace.

Important notes about custom execution hooks

Consider the following when planning execution hooks for your apps.

> ⓘ Because execution hooks often reduce or completely disable the functionality of the application they are running against, you should always try to minimize the time your custom execution hooks take to run.
> If you start a backup or snapshot operation with associated execution hooks but then cancel it, the hooks are still allowed to run if the backup or snapshot operation has already begun. This means that the logic used in a post-backup execution hook cannot assume that the backup was completed.

- The execution hooks feature is disabled by default for new Astra Control deployments.

    - You need to enable the execution hooks feature before you can use execution hooks.
    - Owner or Admin users can enable or disable the execution hooks feature for all users defined in the current Astra Control account. Refer to Enable the execution hooks feature and Disable the execution hooks feature for instructions.
    - The feature enablement status is preserved during Astra Control upgrades.
- An execution hook must use a script to perform actions. Many execution hooks can reference the same script.
- Astra Control requires the scripts that execution hooks use to be written in the format of executable shell scripts.
- Script size is limited to 96KB.
- Astra Control uses execution hook settings and any matching criteria to determine which hooks are applicable to a snapshot, backup, or restore operation.
- All execution hook failures are soft failures; other hooks and the data protection operation are still attempted even if a hook fails. However, when a hook fails, a warning event is recorded in the **Activity** page event log.
- To create, edit, or delete execution hooks, you must be a user with Owner, Admin, or Member permissions.
- If an execution hook takes longer than 25 minutes to run, the hook will fail, creating an event log entry with a return code of "N/A". Any affected snapshot will time out and be marked as failed, with a resulting event log entry noting the timeout.
- For ad hoc data protection operations, all hook events are generated and saved in the **Activity** page event log. However, for scheduled data protection operations, only hook failure events are recorded in the event log (events generated by the scheduled data protection operations themselves are still recorded).

Order of execution

When a data protection operation is run, execution hook events take place in the following order:

1. Any applicable custom pre-operation execution hooks are run on the appropriate containers. You can create and run as many custom pre-operation hooks as you need, but the order of execution of these hooks before the operation is neither guaranteed nor configurable.
2. The data protection operation is performed.
3. Any applicable custom post-operation execution hooks are run on the appropriate containers. You can create and run as many custom post-operation hooks as you need, but the order of execution of these hooks after the operation is neither guaranteed nor configurable.

If you create multiple execution hooks of the same type (for example, pre-snapshot), the order of execution of those hooks is not guaranteed. However, the order of execution of hooks of different types is guaranteed. For example, the order of execution of a configuration that has all of the different types of hooks would look like this:

1. Pre-backup hooks executed
2. Pre-snapshot hooks executed
3. Post-snapshot hooks executed
4. Post-backup hooks executed
5. Post-restore hooks executed

You can see an example of this configuration in scenario number 2 from the table in Determine whether a hook will run.

> You should always test your execution hook scripts before enabling them in a production environment. You can use the 'kubectl exec' command to conveniently test the scripts. After you enable the execution hooks in a production environment, test the resulting snapshots and backups to ensure they are consistent. You can do this by cloning the app to a temporary namespace, restoring the snapshot or backup, and then testing the app.

Determine whether a hook will run

Use the following table to help determine if a custom execution hook will run for your app.

Note that all high-level app operations consist of running one of the basic operations of snapshot, backup, or restore. Depending on the scenario, a clone operation can consists of various combinations of these operations, so what execution hooks a clone operation runs will vary.

In-place restore operations require an existing snapshot or backup, so these operations don't run snapshot or backup hooks.

> If you start but then cancel a backup that includes a snapshot and there are associated execution hooks, some hooks might run, and others might not. This means that a post-backup execution hook cannot assume that the backup was completed. Keep in mind the following points for cancelled backups with associated execution hooks:
>
> - The pre-backup and post-backup hooks are always run.
> - If the backup includes a new snapshot and the snapshot has started, the pre-snapshot and post-snapshot hooks are run.
> - If the backup is cancelled prior to the snapshot starting, the pre-snapshot and post-snapshot hooks are not run.

Scenario	Operation	Existing snapshot	Existing backup	Namespace	Cluster	Snapshot hooks run	Backup hooks run	Restore hooks run
1	Clone	N	N	New	Same	Y	N	Y
2	Clone	N	N	New	Different	Y	Y	Y
3	Clone or restore	Y	N	New	Same	N	N	Y
4	Clone or restore	N	Y	New	Same	N	N	Y
5	Clone or restore	Y	N	New	Different	N	N	Y
6	Clone or restore	N	Y	New	Different	N	N	Y
7	Restore	Y	N	Existing	Same	N	N	Y
8	Restore	N	Y	Existing	Same	N	N	Y
9	Snapshot	N/A	N/A	N/A	N/A	Y	N/A	N/A
10	Backup	N	N/A	N/A	N/A	Y	Y	N/A

Scenario	Operation	Existing snapshot	Existing backup	Namespace	Cluster	Snapshot hooks run	Backup hooks run	Restore hooks run
11	Backup	Y	N/A	N/A	N/A	N	N	N/A

Execution hook examples

Visit the NetApp Verda GitHub project to download real execution hooks for popular apps such as Apache Cassandra and Elasticsearch. You can also see examples and get ideas for structuring your own custom execution hooks.

Enable the execution hooks feature

If you are an Owner or Admin user, you can enable the execution hooks feature. When you enable the feature, all users defined in this Astra Control account can use execution hooks and view existing execution hooks and hook scripts.

Steps

1. Go to **Applications** and then select the name of a managed app.
2. Select the **Execution hooks** tab.
3. Select **Enable execution hooks**.

   The **Account** > **Feature settings** tab appears.

4. In the **Execution hooks** pane, select the settings menu.
5. Select **Enable**.
6. Note the security warning that appears.
7. Select **Yes, enable execution hooks**.

Disable the execution hooks feature

If you are an Owner or Admin user, you can disable the execution hooks feature for all users defined in this Astra Control account. You must delete all existing execution hooks before you can disable the execution hooks feature. Refer to Delete an execution hook for instructions on deleting an existing execution hook.

Steps

1. Go to **Account** and then select the **Feature settings** tab.
2. Select the **Execution hooks** tab.
3. In the **Execution hooks** pane, select the settings menu.
4. Select **Disable**.
5. Note the warning that appears.
6. Type `disable` to confirm that you want to disable the feature for all users.
7. Select **Yes, disable**.

View existing execution hooks

You can view existing custom execution hooks for an app.

Steps

1. Go to **Applications** and then select the name of a managed app.
2. Select the **Execution hooks** tab.

   You can view all enabled or disabled execution hooks in the resulting list. You can see a hook's status, how many containers it matches, creation time, and when it runs (pre- or post-operation). You can select the + icon next to the hook name to expand the list of containers it will run on. To view event logs surrounding execution hooks for this application, go to the **Activity** tab.

View existing scripts

You can view the existing uploaded scripts. You can also see which scripts are in use, and what hooks are using them, on this page.

Steps

1. Go to **Account**.
2. Select the **Scripts** tab.

   You can see a list of existing uploaded scripts on this page. The **Used by** column shows which execution hooks are using each script.

Add a script

Each execution hook must use a script to perform actions. You can add one or more scripts that execution hooks can reference. Many execution hooks can reference the same script; this allows you to update many execution hooks by only changing one script.

Steps

1. Ensure that the execution hooks feature is enabled.
2. Go to **Account**.
3. Select the **Scripts** tab.
4. Select **Add**.
5. Do one of the following:
   - Upload a custom script.
     a. Select the **Upload file** option.
     b. Browse to a file and upload it.
     c. Give the script a unique name.
     d. (Optional) Enter any notes other administrators should know about the script.
     e. Select **Save script**.
   - Paste in a custom script from the clipboard.
     a. Select the **Paste or type** option.
     b. Select the text field and paste the script text into the field.
     c. Give the script a unique name.
     d. (Optional) Enter any notes other administrators should know about the script.
6. Select **Save script**.

Result

The new script appears in the list on the **Scripts** tab.

Delete a script

You can remove a script from the system if it is no longer needed and not used by any execution hooks.

Steps

1. Go to **Account**.
2. Select the **Scripts** tab.
3. Choose a script you want to remove, and select the menu in the **Actions** column.
4. Select **Delete**.

> If the script is associated with one or more execution hooks, the **Delete** action is unavailable. To delete the script, first edit the associated execution hooks and associate them with a different script.

Create a custom execution hook

You can create a custom execution hook for an app and add it to Astra Control. Refer to Execution hook examples for hook examples. You need to have Owner, Admin, or Member permissions to create execution hooks.

> When you create a custom shell script to use as an execution hook, remember to specify the appropriate shell at the beginning of the file, unless you are running specific commands or providing the full path to an executable.

Steps

1. Ensure that the execution hooks feature is enabled.
2. Select **Applications** and then select the name of a managed app.
3. Select the **Execution hooks** tab.
4. Select **Add**.
5. In the **Hook Details** area:
   a. Determine when the hook should run by selecting an operation type from the **Operation** drop-down menu.
   b. Enter a unique name for the hook.
   c. (Optional) Enter any arguments to pass to the hook during execution, pressing the Enter key after each argument you enter to record each one.
6. (Optional) In the **Hook Filter Details** area, you can add filters to control which containers the execution hook runs on:
   a. Select **Add filter**.
   b. In the **Hook filter type** column, choose an attribute on which to filter from the drop-down menu.
   c. In the **Regex** column, enter a regular expression to use as the filter. Astra Control uses the Regular Expression 2 (RE2) regex syntax.

> If you filter on the exact name of an attribute (such as a pod name) with no other text in the regular expression field, a substring match is performed. To match an exact name and only that name, use the exact string match syntax (for example, `^exact_podname$`).

   d. To add more filters, select **Add filter**.

> Multiple filters for an execution hook are combined with a logical AND operator. You can have up to 10 active filters per execution hook.

7. When done, select **Next**.
8. In the **Script** area, do one of the following:
   - Add a new script.
     a. Select **Add**.
     b. Do one of the following:
        - Upload a custom script.
          i. Select the **Upload file** option.
          ii. Browse to a file and upload it.
          iii. Give the script a unique name.
          iv. (Optional) Enter any notes other administrators should know about the script.
          v. Select **Save script**.
        - Paste in a custom script from the clipboard.
          i. Select the **Paste or type** option.
          ii. Select the text field and paste the script text into the field.
          iii. Give the script a unique name.
          iv. (Optional) Enter any notes other administrators should know about the script.
   - Select an existing script from the list.

     This instructs the execution hook to use this script.
9. Select **Next**.
10. Review the execution hook configuration.
11. Select **Add**.

Check the state of an execution hook

After a snapshot, backup, or restore operation finishes running, you can check the state of execution hooks that ran as part of the operation. You can use this status information to determine if you want to keep the execution hook, modify it, or delete it.

Steps

1. Select **Applications** and then select the name of a managed app.
2. Select the **Data protection** tab.
3. Select **Snapshots** to see running snapshots, or **Backups** to see running backups.

The **Hook state** shows the status of the execution hook run after the operation is complete. You can hover over the state for more details. For example, if there are execution hook failures during a snapshot, hovering over the hook state for that snapshot gives a list of failed execution hooks. To see reasons for each failure, you can check the **Activity** page in the left-side navigation area.

View script usage

You can see which execution hooks use a particular script in the Astra Control web UI.

Steps

1. Select **Account**.
2. Select the **Scripts** tab.

   The **Used by** column in the list of scripts contains details on which hooks are using each script in the list.

3. Select the information in the **Used by** column for a script you are interested in.

   A more detailed list appears, with the names of hooks that are using the script and the type of operation they are configured to run with.

Edit an execution hook

You can edit an execution hook if you want to change its attributes, filters, or the script that it uses. You need to have Owner, Admin, or Member permissions to edit execution hooks.

Steps

1. Select **Applications** and then select the name of a managed app.
2. Select the **Execution hooks** tab.
3. Select the Options menu in the **Actions** column for a hook that you wish to edit.
4. Select **Edit**.
5. Make any needed changes, selecting **Next** after you complete each section.
6. Select **Save**.

Disable an execution hook

You can disable an execution hook if you want to temporarily prevent it from running before or after a snapshot of an app. You need to have Owner, Admin, or Member permissions to disable execution hooks.

Steps

1. Select **Applications** and then select the name of a managed app.
2. Select the **Execution hooks** tab.
3. Select the Options menu in the **Actions** column for a hook that you wish to disable.
4. Select **Disable**.

Delete an execution hook

You can remove an execution hook entirely if you no longer need it. You need to have Owner, Admin, or Member permissions to delete execution hooks.

Steps

1. Select **Applications** and then select the name of a managed app.
2. Select the **Execution hooks** tab.
3. Select the Options menu in the **Actions** column for a hook that you wish to delete.
4. Select **Delete**.
5. In the resulting dialog, type "delete" to confirm.
6. Select **Yes, delete execution hook**.

For more information

- NetApp Verda GitHub project

View app and compute health

View a summary of app and cluster health

Click the **Dashboard** to see a high-level view of your apps, clusters, and their health.

The Apps tile helps you identify the following:

- How many apps you're currently managing.
- Whether those managed apps are healthy.
- Whether the apps are fully protected (they're protected if recent backups are available).

Note that these aren't just numbers or statuses—you can drill down from each of these. For example, if apps aren't fully protected, you can hover over the icon to identify which apps aren't fully protected, which includes a reason why.

The Clusters tile provides similar details about the health of the cluster and you can drill down to get more details just like you can with an app.

View the health and details of clusters

After you add Kubernetes clusters to Astra Control, you can view details about the cluster, such as its location, the worker nodes, persistent volumes, and storage classes.

Steps

1. In the Astra Control Service UI, select **Clusters**.
2. On the **Clusters** page, select the cluster whose details you want to view.

   > ⓘ If a cluster is in `removed` state yet cluster and network connectivity appears healthy (external attempts to access the cluster using Kubernetes APIs are successful), the kubeconfig you provided to Astra Control might no longer be valid. This can be due to certificate rotation or expiration on the cluster. To correct this issue, update the credentials associated with the cluster in Astra Control using the Astra Control API.

3. View the information on the **Overview**, **Storage**, and **Activity** tabs to find the information that you're looking for.

  - **Overview**: Details about the worker nodes, including their state.
  - **Storage**: The persistent volumes associated with the compute, including the storage class and state.
  - **Activity**: The activities related to the cluster.

> ⓘ You can also view cluster information starting from the Astra Control Service **Dashboard**. On the **Clusters** tab under **Resource summary**, you can select the managed clusters, which takes you to the **Clusters** page. After you get to the **Clusters** page, follow the steps outlined above.

View the health and details of an app

After you start managing an app, Astra Control provides details about the app that enables you to identify its communication status (whether Astra Control can communicate with the app), its protection status (whether it's fully protected in case of failure), the pods, persistent storage, and more.

Steps

1. Select **Applications** and then select the name of an app.
2. Find the information that you're looking for:

   App Status
   Provides a status that reflects whether Astra Control can communicate with the application.

   App Protection Status
   Provides a status of how well the app is protected:

   - **Fully protected**: The app has an active backup schedule and a successful backup that's less than a week old
   - **Partially protected**: The app has an active backup schedule, an active snapshot schedule, or a successful backup or snapshot
   - **Unprotected**: Apps that are neither fully protected or partially protected.

     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 it's persistent storage, then you need a backup to recover. A snapshot wouldn't enable you to recover.

   Overview
   Information about the state of the pods that are associated with the app.

   Data protection
   Enables you to configure a data protection policy and to view the existing snapshots and backups.

   Storage
   Shows you the app-level persistent volumes. The state of a persistent volume is from the perspective of the Kubernetes cluster.

   Resources
   Enables you to verify which resources are being backed up and managed.

Activity

The Astra Control activities related to the app.

Manage buckets

You can manage the buckets that Astra uses for backups and clones. You can add additional buckets, remove existing buckets, and change the default bucket for the Kubernetes clusters in a cloud instance.

Only Owners and Admins can manage buckets.

How Astra Control uses buckets

When you start managing your first Kubernetes cluster for a cloud instance, Astra Control Service creates the initial bucket for that cloud instance.

You can manually designate a bucket as the default bucket for a cloud instance. If you do so, Astra Control Service uses this bucket by default for the backups and clones that you create on any managed cluster in that cloud instance (you can select a different bucket for backups). If you perform a live clone of an application from any of the managed clusters in a cloud instance to another cluster, Astra Control Service uses the default bucket for the source cloud instance to perform the clone operation.

You can set the same bucket as the default bucket for multiple cloud instances.

You can select from any buckets when you create a protection policy or start an ad-hoc backup.

> Astra Control Service checks whether a destination bucket is accessible prior to starting a backup or a clone.

View existing buckets

View the list of buckets that are available to Astra Control Service to determine their status and to identify the default bucket (if defined) for your cloud instance.

A bucket can have any of the following states:

Pending

After you add a bucket, it starts in the pending state while Astra Control discovers it.

Available

The bucket is available for use by Astra Control.

Removed

The bucket isn’t operational at the moment. Hover your mouse over the status icon to identify what the problem is.

If a bucket is in the Removed state, you can still set it as the default bucket and assign it to a protection schedule. But if the bucket isn’t in the Available state by the time a data protection operation starts, then that operation will fail.

Step

1. Go to **Buckets**.

   The list of buckets available to Astra Control Service is displayed.

Add an additional bucket

You can add additional buckets at any time. This enables you to choose between buckets when creating a protection policy or starting an ad-hoc backup, and enables you to change the default bucket that a cloud instance uses.

You can add the following types of buckets:

- Amazon Web Services
- Generic S3
- Google Cloud Platform
- Microsoft Azure
- NetApp ONTAP S3
- NetApp StorageGRID S3

Before you begin

- Ensure you know the name of an existing bucket.
- Ensure you have credentials for the bucket that provide Astra Control with the permissions that it needs to manage the bucket.
- If your bucket is in Microsoft Azure:
  - The bucket must belong to the resource group named *astra-backup-rg*.
  - If the Azure storage account instance performance setting is set to "Premium", the "Premium account type" setting must be set to "Block blobs".

Steps

1. Go to **Buckets**.
2. Select **Add** and follow the prompts to add the bucket.
   - **Type**: Choose your cloud provider.
   - **Existing bucket name**: Enter the name of the bucket.
   - **Description**: Optionally enter a description of the bucket.
   - **Storage account** (Azure only): Enter the name of your Azure storage account. This bucket must belong to the resource group named *astra-backup-rg*.
   - **S3 server name or IP address** (AWS and S3 bucket types only): Enter the fully qualified domain name of the S3 endpoint that corresponds with your region, without `https://`. Refer to the Amazon documentation for more information.
   - **Select credentials**: Enter the credentials that provide Astra Control Service with the permissions that it needs to manage the bucket. The information you need to provide varies depending on the bucket type.
3. Select **Add** to add the bucket.

Result

Astra Control Service adds the bucket. You can now choose this bucket when creating a protection policy or performing an ad-hoc backup. You can also set this bucket as the default bucket for a cloud instance.

Change the default bucket

You can change the default bucket for a cloud instance. Astra Control Service will use this bucket by default for backups and clones. Each cloud instance has its own default bucket.

> ⓘ Astra Control does not automatically assign a default bucket for any cloud instance. You need to manually set a default bucket for a cloud instance before performing app clone operations between two clusters.

Steps

1. Go to **Cloud instances**.
2. Select the configuration menu in the **Actions** column for the cloud instance that you want to edit.
3. Select **Edit**.
4. In the list of buckets, select the bucket you want to make the default bucket for this cloud instance.
5. Select **Update**.

Remove a bucket

You can remove a bucket that is no longer in use or is not healthy. You might want to do this to keep your object store configuration simple and up-to-date.

> ⓘ
> - You cannot remove a default bucket. If you want to remove that bucket, first select another bucket as the default.
> - You cannot remove a write once read many (WORM) bucket before the bucket’s cloud provider retention period has expired. WORM buckets are denoted with "Locked" next to the bucket name.

Before you begin

- You should check to ensure that there are no running or completed backups for this bucket before you begin.
- You should check to ensure that the bucket is not being used for any scheduled backups.

If there are, you will not be able to continue.

Steps

1. Go to **Buckets**.
2. From the **Actions** menu, select **Remove**.

   > ⓘ Astra Control ensures first that there are no schedule policies using the bucket for backups and that there are no active backups in the bucket you are about to remove.

3. Type "remove" to confirm the action.
4. Select **Yes, remove bucket**.

Find more information

- Use the Astra Control API

Monitor running tasks

You can view details about running tasks and tasks that have completed, failed, or been cancelled in the last 24 hours in Astra Control. For example, you can view the status of a running backup, restore, or clone operation, and see details like percentage completed and estimated time remaining. You can view the status of a scheduled operation that has run or an operation that you started manually.

While viewing a running or completed task, you can expand the task details to see the status of each of the subtasks. The task progress bar is green for ongoing or completed tasks, blue for cancelled tasks, and red for tasks that failed because of an error.

> ⓘ For clone operations, the task subtasks consist of a snapshot and a snapshot restore operation.

To refer to more information about failed tasks, refer to Monitor account activity.

Steps

1. While a task is running, go to **Applications**.
2. Select the name of an application from the list.
3. In the details of the application, select the **Tasks** tab.

   You can view details of current or past tasks, and filter by task state.

> ⓘ Tasks are retained in the **Tasks** list for up to 24 hours. You can configure this limit and other task monitor settings using the Astra Control API.

Manage your account

Set up billing

You can use more than one method to manage your Astra Control Service account billing. If you are using Azure or Amazon AWS, you can subscribe to an Astra Control Service plan through the Microsoft Azure Marketplace or the AWS Marketplace. When you do this, you can manage your billing details through the Marketplace. Or, you can subscribe directly with NetApp. If you subscribe directly with NetApp, you can manage your billing details through Astra Control Service. If you use Astra Control Service without a subscription, you are automatically subscribed to the Free Plan.

The Astra Control Service Free Plan enables you to manage up to 10 namespaces in your account. If you want to manage more than 10 namespaces, then you’ll need to set up billing by upgrading from the Free Plan to the Premium Plan, or subscribe through the Azure Marketplace or AWS Marketplace.

Billing overview

There are two types of costs associated with using Astra Control Service: charges from NetApp for the Astra Control Service and charges from your cloud provider for persistent volumes and object storage.

Astra Control Service billing

Astra Control Service offers three plans:

Free Plan

Manage up to 10 namespaces for free.

Premium PayGo

Manage an unlimited amount of namespaces at a specific rate, per namespace.

Premium Subscription

Pre-pay at a discounted rate with an annual subscription that enables you to manage up to 20 namespaces per *namespace pack*. Contact NetApp Sales to purchase as many packs as needed for your organization. For example, purchase 3 packs to manage 60 namespaces from Astra Control Service. If you manage more namespaces than allowed by your annual subscription, then you'll be charged at the subscription-dependent overage rate per extra namespace. If you don't have an Astra Control account yet, purchasing the Premium Subscription automatically creates an Astra Control account for you. If you have an existing Free Plan, then you're automatically converted to the Premium Subscription.

When you create an Astra Control account, you're automatically subscribed to the Free Plan. Astra Control's Dashboard shows you how many namespaces you're currently managing out of the 10 free namespaces that you're allowed. Billing starts for a namespace when the first app containing the namespace is managed, and stops for that namespace when the last app containing the namespace is unmanaged.

If you try to manage an 11th namespace, Astra Control notifies you that you've reached the limit of the Free Plan. It then prompts you to upgrade from the Free Plan to a Premium Plan. You'll be charged at the subscription-dependent overage rate per extra namespace.

You can upgrade to a Premium Plan at any time. After you upgrade, Astra Control starts charging you for *all* namespaces in the account. The first 10 namespaces don't stay in the Free Plan.

Google Cloud billing

Persistent volumes are backed by NetApp Cloud Volumes Service and backups of your apps are stored in a Google Cloud Storage bucket.

- View pricing details for Cloud Volumes Service.

  Note that Astra Control Service supports all service types and service levels. The service type that you use depends on your Google Cloud region.

- View pricing details for Google Cloud storage buckets.

Microsoft Azure billing

Persistent volumes are backed by Azure NetApp Files and backups of your apps are stored in an Azure Blob container.

- View pricing details for Azure NetApp Files.

- View pricing details for Microsoft Azure Blob storage.
- View Astra Control Service plans and pricing in Azure Marketplace

> The Azure billing rate for Astra Control Service is per hour, and a new billing hour starts after 29 minutes of the usage hour has elapsed.

Amazon Web Services billing

Persistent volumes are backed by EBS or FSx for NetApp ONTAP and backups of your apps are stored in an AWS bucket.

- View pricing details for Amazon Web Services.

Subscribe to Astra Control Service in the Azure Marketplace

You can subscribe to Astra Control Service using the Azure Marketplace. Your account and billing details are managed through the Marketplace.

> To see a video walkthrough of the Azure Marketplace subscription process, visit NetApp TV.

Steps

1. Go to the Azure Marketplace.
2. Select **Get It Now**.
3. Follow the instructions to subscribe to a plan.

Subscribe to Astra Control Service in the AWS Marketplace

You can subscribe to Astra Control Service using the AWS Marketplace. Your account and billing details are managed through the Marketplace.

Steps

1. Go to the AWS Marketplace.
2. Select **View purchase options**.
3. If prompted to do so, log in to your AWS account, or create a new account.
4. Follow the instructions to subscribe to a plan.

Subscribe to Astra Control Service directly with NetApp

You can subscribe to Astra Control Service from within the Astra Control Service UI or by contacting NetApp Sales.

Upgrade from the Free Plan to the Premium PayGo Plan

Upgrade your billing plan at any time to start managing more than 10 namespaces from Astra Control by paying as you go. All you need is a valid credit card.

Steps

1. Select **Account** and then select **Billing**.
2. Under **Plans**, go to **Premium PayGo** and select **Upgrade Now**.

3. Provide payment details for a valid credit card and select **Upgrade to Premium Plan**.

   > ⓘ Astra Control will email you if the credit card is nearing expiration.

Result

You can now manage more than 10 namespaces. Astra Control starts charging you for *all* namespaces that you're currently managing.

Upgrade from the Free Plan to the Premium Subscription

Contact NetApp Sales to pre-pay at a discounted rate with an annual subscription.

Steps

1. Select **Account** and then select **Billing**.
2. Under **Plans**, go to **Premium Subscription** and select **Contact Sales**.
3. Provide details to the sales team to start the process.

Result

A NetApp Sales representative will contact you to process your purchase order. After the order is complete, Astra Control will reflect your current plan on the **Billing** tab.

View your current costs and billing history

Astra Control shows you your current monthly costs, as well as a detailed billing history by namespace. If you subscribed to a plan through a Marketplace, the billing history is not visible (but you can view it by logging in to the Marketplace.)

Steps

1. Select **Account** and then select **Billing**.

   Your current costs appear under the billing overview.

2. To view the billing history by namespace, select **Billing history**.

   Astra Control shows you the usage minutes and cost for each namespace. A usage minute is how many minutes Astra Control managed your namespace during a billing period.

3. Select the drop-down list to select a previous month.

Change the credit card for Premium PayGo

If needed, you can change the credit card that Astra Control has on file for billing.

Steps

1. Select **Account > Billing > Payment method**.
2. Select the configure icon.
3. Modify the credit card.

Important notes

- Your billing plan is per Astra Control account.

  If you have multiple accounts, then each has its own billing plan.

- Your Astra Control bill includes charges for managing your namespaces. You’re charged separately by your cloud provider for the storage backend for persistent volumes.

  Learn more about Astra Control pricing.

- Each billing period ends on the last day of the month.
- You can’t downgrade from a Premium Plan to the Free Plan.

Invite and remove users

Invite users to join your Astra Control account and remove users that should no longer have access to the account.

Invite users

Account Owners and Admins can invite other users to join the Astra Control account.

Steps

1. Make sure that the user has a BlueXP login.
2. Select **Account**.
3. In the **Users** tab, select **Invite**.
4. Enter the user’s name, email address, and their role.

   Note the following:

   - The email address must match the email address that the user used to sign up to BlueXP.
   - Each role provides the following permissions:
     - An **Owner** has Admin permissions and can delete accounts.
     - An **Admin** has Member permissions and can invite other users.
     - A **Member** can fully manage apps and clusters.
     - A **Viewer** can view resources.
5. To add constraints to a user with a Member or Viewer role, enable the **Restrict role to constraints** check box.

   For more information on adding constraints, refer to Manage roles.

6. To invite another user, select **Add another user** and enter information for the new user.

   You can invite up to 10 users at a time. You can navigate between the users you are inviting on the left side of the **Invite users** dialog.

7. Select **Invite users**.

Result

The user or users will receive an email that invites them to join your account.

Change a user’s role

An Account Owner can change the role of all users, while an Account Admin can change the role of users who have the Admin, Member, or Viewer role.

Steps

1. Select **Account**.
2. In the **Users** tab, select the menu in the **Actions** column for the user.
3. Select **Edit role**.
4. Select a new role.
5. To add constraints to a user with a Member or Viewer role, enable the **Restrict role to constraints** check box.

   For more information on adding constraints, refer to Manage roles.

6. Select **Confirm**.

Result

Astra Control updates the user’s permissions based on the new role that you selected.

Remove users

A user with the Owner role can remove other users from the account at any time.

Steps

1. Select **Account**.
2. In the **Users** tab, select the users that you want to remove.
3. Select the menu in the **Actions** column and select **Remove user**.
4. When you’re prompted, confirm deletion by typing "remove" and then select **Yes, Remove User**.

Result

Astra Control removes the user from the account.

Manage roles

You can manage roles by adding namespace constraints and restricting user roles to those constraints. This enables you to control access to resources within your organization. You can use the Astra Control UI or the Astra Control API to manage roles.

Add a namespace constraint to a role

An Admin or Owner user can add namespace constraints.

Steps

1. In the **Manage Your Account** navigation area, select **Account**.
2. Select the **Users** tab.

3. In the **Actions** column, select the menu button for a user with the Member or Viewer role.
4. Select **Edit role**.
5. Enable the **Restrict role to constraints** check box.

   The check box is only available for Member or Viewer roles. You can select a different role from the **Role** drop-down list.

6. Select **Add constraint**.

   You can view the list of available constraints by namespace or by namespace label.

7. In the **Constraint type** drop-down list, select either **Kubernetes namespace** or **Kubernetes namespace label** depending on how your namespaces are configured.
8. Select one or more namespaces or labels from the list to compose a constraint that restricts roles to those namespaces.
9. Select **Confirm**.

   The **Edit role** page displays the list of constraints you've chosen for this role.

10. Select **Confirm**.

    On the **Account** page, you can view the constraints for any Member or Viewer role in the **Role** column.

> ⓘ If you enable constraints for a role and select **Confirm** without adding any constraints, the role is considered to have full restrictions (the role is denied access to any resources that are assigned to namespaces).

Remove a namespace constraint from a role

An Admin or Owner user can remove a namespace constraint from a role.

Steps

1. In the **Manage Your Account** navigation area, select **Account**.
2. Select the **Users** tab.
3. In the **Actions** column, select the menu button for a user with the Member or Viewer role that has active constraints.
4. Select **Edit role**.

   The **Edit role** dialog displays the active constraints for the role.

5. Select the **X** to the right of the constraint you need to remove.
6. Select **Confirm**.

For more information

- User roles and namespaces

Add and remove credentials

Add and remove cloud provider credentials from your account at any time. Astra Control uses these credentials to discover a Kubernetes cluster, the apps on the cluster, and to provision resources on your behalf.

Note that all users in Astra Control share the same sets of credentials.

Add credentials

The most common way to add credentials to Astra Control is when you manage clusters, but you can also add credentials from the Account page. The credentials will then be available to choose when you manage additional Kubernetes clusters.

Before you begin

- For Amazon Web Services, you should have the JSON output of the credentials for the IAM account used to create the cluster. Learn how to set up an IAM user.
- For GKE, you should have the service account key file for a service account that has the required permissions. Learn how to set up a service account.
- For AKS, you should have the JSON file that contains the output from the Azure CLI when you created the service principal. Learn how to set up a service principal.

  You'll also need your Azure subscription ID, if you didn't add it to the JSON file.

Steps

1. Select **Account > Credentials**.
2. Select **Add Credentials**.
3. Select **Microsoft Azure**.
4. Select **Google Cloud Platform**.
5. Select **Amazon Web Services**.
6. Enter a name for the credentials that distinguishes them from other credentials in Astra Control.
7. Provide the required credentials.
   a. **Microsoft Azure**: Provide Astra Control with details about your Azure service principal by uploading a JSON file or by pasting the contents of that JSON file from your clipboard.

      The JSON file should contain the output from the Azure CLI when you created the service principal. It can also include your subscription ID so it's automatically added to Astra Control. Otherwise, you need to manually enter the ID after providing the JSON.

   b. **Google Cloud Platform**: Provide the Google Cloud service account key file either by uploading the file or by pasting the contents from your clipboard.

   c. **Amazon Web Services**: Provide the Amazon Web Services IAM user credentials either by uploading the file or by pasting the contents from your clipboard.
8. Select **Add Credentials**.

Result

The credentials are now available to select when you add a cluster to Astra Control.

Remove credentials

Remove credentials from an account at any time. You should only remove credentials after unmanaging all clusters, unless you are rotating credentials (refer to Rotate credentials).

> ⓘ The first set of credentials that you add to Astra Control is always in use because Astra Control uses the credentials to authenticate to the backup bucket. It's best not to remove these credentials.

Steps

1. Select **Account > Credentials**.
2. Select the drop-down list in the **State** column for the credentials that you want to remove.
3. Select **Remove**.
4. Type the name of the credentials to confirm deletion and then select **Yes, Remove Credentials**.

Result

Astra Control removes the credentials from the account.

Rotate credentials

You can rotate credentials in your account. If you rotate credentials, rotate them during a maintenance window when no backups are in progress (scheduled or on-demand).

Steps

1. Remove the existing credentials by following the steps in Remove credentials.
2. Add the new credentials by following the steps in Add credentials.
3. Update all buckets to use the new credentials:
   a. From the left navigation, select **Buckets**.
   b. Select the drop-down list in the **Actions** column for the bucket that you want to edit.
   c. Select **Edit**.
   d. In the **Select credentials** section, choose the new credentials that you added to Astra Control.
   e. Select **Update**.
   f. Repeat steps **b** through **e** for any remaining buckets on your system.

Result

Astra Control begins using the new cloud provider credentials.

Monitor account activity

You can view details about the activities in your Astra Control account. For example, when new users were invited, when a cluster was added, or when a snapshot was taken. You also have the ability to export your account activity to a CSV file.

View all account activity in Astra Control

1. Select **Activity**.
2. Use the filters to narrow down the list of activities or use the search box to find exactly what you're looking for.

3. Select **Export to CSV** to download your account activity to a CSV file.

View account activity for a specific app

1. Select **Applications** and then select the name of an app.
2. Select **Activity**.

View account activity for clusters

1. Select **Clusters** and then select the name of the cluster.
2. Select **Activity**.

View and manage notifications

Astra Control notifies you when actions have completed or failed. For example, you'll see a notification if a backup of an app completed successfully.

The number of unread notifications is available in the top right of the interface.

You can view these notifications and mark them as read (this can come in handy if you like to clear unread notifications like we do).

Steps

1. Select the number of unread notifications in the top right.
2. Review the notifications and then select **Mark as read** or **Show all notifications**.

   If you selected **Show all notifications**, the Notifications page loads.

3. On the **Notifications** page, view the notifications, select the ones that you want to mark as read, select **Action** and select **Mark as read**.

Close your account

If you no longer need your Astra Control account, you can close it at any time.

> ⓘ Buckets that Astra Control automatically created will be automatically deleted when you close your account.

Steps

1. Unmanage all apps and clusters.
2. Remove credentials from Astra Control.
3. Select **Account > Billing > Payment method**.
4. Select **Close Account**.
5. Enter your account name and confirm to close the account.

Manage cloud instances

A cloud instance is a unique domain within a cloud provider. You can create multiple cloud instances for each cloud provider, and each cloud instance has its own name, credentials, and associated clusters.

You create a cloud instance when you add a new cluster to Astra Control. You can edit a cloud instance to change its name or default bucket using the Astra Control UI, and perform other actions with the cloud instance using the Astra Control API.

Add a cloud instance

You can add a new cloud instance when you add a new cluster to Astra Control. Refer to Start managing Kubernetes clusters from Astra Control Service for more information.

Edit a cloud instance

You can modify an existing cloud instance for a cloud provider.

Steps

1. Go to **Cloud instances**.
2. In the list of cloud instances, select the **Actions** menu for the cloud instance you want to edit.
3. Select **Edit**.

   On this page, you can update the name and default bucket for the cloud instance.

   > ⓘ Each cloud instance in Astra Control must have a unique name.

Rotate the credentials for a cloud instance

You can use the Astra Control API to rotate the credentials for a cloud instance. To learn more, go to the Astra automation docs.

Remove a cloud instance

You can use the Astra Control API to remove a cloud instance from a cloud provider. To learn more, go to the Astra automation docs.

Enable Astra Control Provisioner

Astra Trident versions 23.10 and later include the option to use Astra Control Provisioner, which enables licensed Astra Control users to access advanced storage provisioning functionality. Astra Control Provisioner provides this extended functionality in addition to standard Astra Trident CSI-based functionality. You can use this procedure to enable and install Astra Control Provisioner.

Your Astra Control Service subscription automatically includes the license for Astra Control Provisioner use.

In coming Astra Control updates, Astra Control Provisioner will replace Astra Trident as storage provisioner and orchestrator and be mandatory for Astra Control use. Because of this, it’s strongly recommended that Astra Control users enable Astra Control Provisioner. Astra Trident will continue to remain open source and be released, maintained, supported, and updated with new CSI and other features from NetApp.

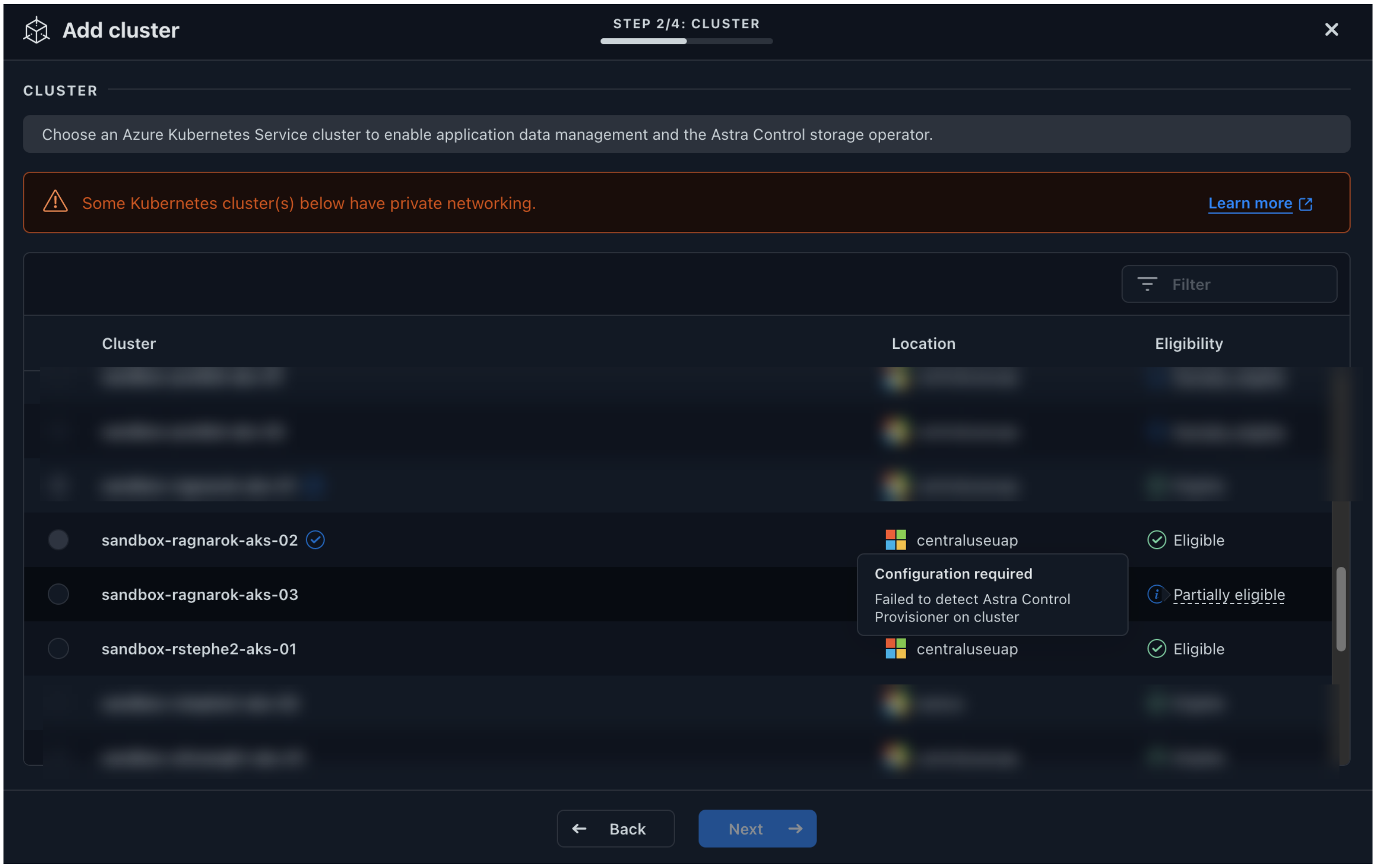
How do I know if I need to enable Astra Control Provisioner?

If you add a cluster to Astra Control Service that does not have Astra Trident previously installed, the cluster will be marked as `Eligible`. After you add the cluster to Astra Control, Astra Control Provisioner will be automatically enabled.

If your cluster is not marked `Eligible`, it will be marked `Partially eligible` because of one of the following:

- It's using an older version of Astra Trident
- It's using an Astra Trident 23.10 that does not yet have the provisioner option enabled
- It's a cluster type that does not allow automatic enablement

For `Partially eligible` cases, use these instructions to manually enable Astra Control Provisioner for your cluster.



Add cluster

STEP 2/4: CLUSTER

CLUSTER

Choose an Azure Kubernetes Service cluster to enable application data management and the Astra Control storage operator.

⚠ Some Kubernetes cluster(s) below have private networking. Learn more

Cluster	Location	Eligibility
○ sandbox-ragnarok-aks-02 ✓	centraluseuap	✓ Eligible
○ sandbox-ragnarok-aks-03		ⓘ Partially eligible
○ sandbox-rstephe2-aks-01	centraluseuap	✓ Eligible

Configuration required
Failed to detect Astra Control Provisioner on cluster

Before you enable Astra Control Provisioner

If you have an existing Astra Trident without Astra Control Provisioner and want to enable Astra Control Provisioner, do the following first:

- **If you have Astra Trident installed, confirm that its version is within a four-release window**: You can perform a direct upgrade to Astra Trident 24.02 with Astra Control Provisioner if your Astra Trident is within a four-release window of version 24.02. For example, you can directly upgrade from Astra Trident 23.04 to 24.02.
- **Confirm that your cluster has an AMD64 system architecture**: The Astra Control Provisioner image is provided in both AMD64 and ARM64 CPU architectures, but only AMD64 is supported by Astra Control.

Steps

1. Access the NetApp Astra Control image registry:

   a. Log on to the Astra Control Service UI and record your Astra Control account ID.

      i. Select the figure icon at the top right of the page.

      ii. Select **API access**.

      iii. Write down your account ID.

   b. From the same page, select **Generate API token** and copy the API token string to the clipboard and save it in your editor.

   c. Log into the Astra Control registry using your preferred method:

```
docker login cr.astra.netapp.io -u <account-id> -p <api-token>
```

```
crane auth login cr.astra.netapp.io -u <account-id> -p <api-token>
```

2. (Custom registries only) Follow these steps to move the image to your custom registry. If you aren’t using a registry, follow the Trident operator steps in the next section.

> ⓘ You can use Podman instead of Docker for the following commands. If you are using a Windows environment, PowerShell is recommended.

Docker

1. Pull the Astra Control Provisioner image from the registry:

   > The image pulled will not support multiple platforms and will only support the same platform as the host that pulled the image, such as Linux AMD64.

   ```
   docker pull cr.astra.netapp.io/astra/trident-acp:24.02.0
   --platform <cluster platform>
   ```

   Example:

   ```
   docker pull cr.astra.netapp.io/astra/trident-acp:24.02.0
   --platform linux/amd64
   ```

2. Tag the image:

   ```
   docker tag cr.astra.netapp.io/astra/trident-acp:24.02.0
   <my_custom_registry>/trident-acp:24.02.0
   ```

3. Push the image to your custom registry:

   ```
   docker push <my_custom_registry>/trident-acp:24.02.0
   ```

Crane

1. Copy the Astra Control Provisioner manifest to your custom registry:

   ```
   crane copy cr.astra.netapp.io/astra/trident-acp:24.02.0
   <my_custom_registry>/trident-acp:24.02.0
   ```

3. Determine if the original Astra Trident installation method used an operator (either manually or with Helm) or `tridentctl`.
4. Enable Astra Control Provisioner in Astra Trident using the installation method you used originally:

Astra Trident operator

1. Download the Astra Trident installer and extract it.
2. Complete these steps if you have not yet installed Astra Trident or if you removed the operator from your original Astra Trident deployment:
   a. Create the CRD:

   ```
   kubectl create -f
   deploy/crds/trident.netapp.io_tridentorchestrators_crd_post1.16.yaml
   ```

   b. Create the trident namespace (`kubectl create namespace trident`) or confirm that the trident namespace still exists (`kubectl get all -n trident`). If the namespace has been removed, create it again.
3. Update Astra Trident to 24.02.0:

   > For clusters running Kubernetes 1.24 or earlier, use `bundle_pre_1_25.yaml`. For clusters running Kubernetes 1.25 or later, use `bundle_post_1_25.yaml`.

   ```
   kubectl -n trident apply -f trident-installer/deploy/<bundle-name.yaml>
   ```

4. Verify Astra Trident is running:

   ```
   kubectl get torc -n trident
   ```

   Response:

   ```
   NAME      AGE
   trident   21m
   ```

5. If you have a registry that uses secrets, create a secret to use to pull the Astra Control Provisioner image:

   ```
   kubectl create secret docker-registry <secret_name> -n trident
   --docker-server=<my_custom_registry> --docker-username=<username>
   --docker-password=<token>
   ```

6. Edit the TridentOrchestrator CR and make the following edits:

```
kubectl edit torc trident -n trident
```

a. Set a custom registry location for the Astra Trident image or pull it from the Astra Control registry (`tridentImage: <my_custom_registry>/trident:24.02.0` or `tridentImage: netapp/trident:24.02.0`).

b. Enable Astra Control Provisioner (`enableACP: true`).

c. Set the custom registry location for the Astra Control Provisioner image or pull it from the Astra Control registry (`acpImage: <my_custom_registry>/trident-acp:24.02.0` or `acpImage: cr.astra.netapp.io/astra/trident-acp:24.02.0`).

d. If you established image pull secrets earlier in this procedure, you can set them here (`imagePullSecrets: - <secret_name>`). Use the same name secret name you established in the previous steps.

```
apiVersion: trident.netapp.io/v1
kind: TridentOrchestrator
metadata:
  name: trident
spec:
  debug: true
  namespace: trident
  tridentImage: <registry>/trident:24.02.0
  enableACP: true
  acpImage: <registry>/trident-acp:24.02.0
  imagePullSecrets:
  - <secret_name>
```

7. Save and exit the file. The deployment process will begin automatically.
8. Verify the operator, deployment, and replicasets are created.

```
kubectl get all -n trident
```

There should only be **one instance** of the operator in a Kubernetes cluster. Do not create multiple deployments of the Astra Trident operator.

9. Verify the `trident-acp` container is running and that `acpVersion` is `24.02.0` with a status of `Installed`:

```
kubectl get torc -o yaml
```

Response:

```
status:
  acpVersion: 24.02.0
  currentInstallationParams:
    ...
    acpImage: <registry>/trident-acp:24.02.0
    enableACP: "true"
    ...
  ...
  status: Installed
```

tridentctl

1. Download the Astra Trident installer and extract it.
2. If you have an existing Astra Trident, uninstall it from the cluster that hosts it.
3. Install Astra Trident with Astra Control Provisioner enabled (`--enable-acp=true`):

   ```
   ./tridentctl -n trident install --enable-acp=true --acp
   -image=mycustomregistry/trident-acp:24.02
   ```

4. Confirm that Astra Control Provisioner has been enabled:

   ```
   ./tridentctl -n trident version
   ```

   Response:

   ```
   +----------------+----------------+-------------+ | SERVER
   VERSION | CLIENT VERSION | ACP VERSION | +----------------
   +----------------+-------------+ | 24.02.0 | 24.02.0 | 24.02.0. |
   +----------------+----------------+-------------+
   ```

Helm

1. If you have Astra Trident 23.07.1 or earlier installed, uninstall the operator and other components.
2. If your Kubernetes cluster is running 1.24 or earlier, delete psp:

   ```
   kubectl delete psp tridentoperatorpod
   ```

3. Add the Astra Trident Helm repository:

```
helm repo add netapp-trident https://netapp.github.io/trident-
helm-chart
```

4. Update the Helm chart:

```
helm repo update netapp-trident
```

Response:

```
Hang tight while we grab the latest from your chart
repositories...
...Successfully got an update from the "netapp-trident" chart
repository
Update Complete. □Happy Helming!□
```

5. List the images:

```
./tridentctl images -n trident
```

Response:

```
| v1.28.0            | netapp/trident:24.02.0|
|                    | docker.io/netapp/trident-
autosupport:24.02|
|                    | registry.k8s.io/sig-storage/csi-
provisioner:v4.0.0|
|                    | registry.k8s.io/sig-storage/csi-
attacher:v4.5.0|
|                    | registry.k8s.io/sig-storage/csi-
resizer:v1.9.3|
|                    | registry.k8s.io/sig-storage/csi-
snapshotter:v6.3.3|
|                    | registry.k8s.io/sig-storage/csi-node-
driver-registrar:v2.10.0 |
|                    | netapp/trident-operator:24.02.0 (optional)
```

6. Ensure that trident-operator 24.02.0 is available:

```
helm search repo netapp-trident/trident-operator --versions
```

Response:

```
NAME                                CHART VERSION   APP VERSION
DESCRIPTION
netapp-trident/trident-operator 100.2402.0      24.02.0         A
```

7. Use `helm install` and run one of the following options that include these settings:
   - A name for your deployment location
   - The Astra Trident version
   - The name of the Astra Control Provisioner image
   - The flag to enable the provisioner
   - (Optional) A local registry path. If you are using a local registry, your Trident images can be located in one registry or different registries, but all CSI images must be located in the same registry.
   - The Trident namespace

Options

- Images without a registry

```
helm install trident netapp-trident/trident-operator --version
100.2402.0 --set acpImage=cr.astra.netapp.io/astra/trident-
acp:24.02.0 --set enableACP=true --set operatorImage=netapp/trident-
operator:24.02.0 --set
tridentAutosupportImage=docker.io/netapp/trident-autosupport:24.02
--set tridentImage=netapp/trident:24.02.0 --namespace trident
```

- Images in one or more registries

```
helm install trident netapp-trident/trident-operator --version
100.2402.0 --set acpImage=<your-registry>:<acp image> --set
enableACP=true --set imageRegistry=<your-registry>/sig-storage --set
operatorImage=netapp/trident-operator:24.02.0 --set
tridentAutosupportImage=docker.io/netapp/trident-autosupport:24.02
--set tridentImage=netapp/trident:24.02.0 --namespace trident
```

You can use `helm list` to review installation details such as name, namespace, chart, status, app version, and revision number.

If you have any issues deploying Trident using Helm, run this command to fully uninstall Astra Trident:

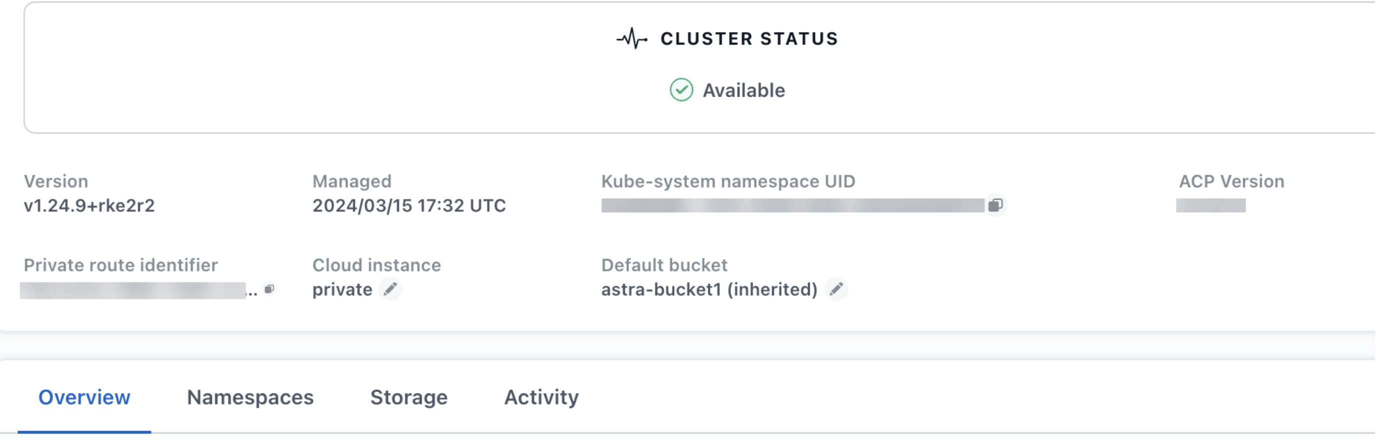
```
./tridentctl uninstall -n trident
```

Do not completely remove Astra Trident CRDs as part of your uninstall before attempting to enable Astra Control Provisioner again.

Result

Astra Control Provisioner functionality is enabled and you can use any features available for the version you are running.

After Astra Control Provisioner is installed, the cluster hosting the provisioner in the Astra Control UI will show an `ACP version` rather than `Trident version` field and current installed version number.



CLUSTER STATUS

Available

Version: v1.24.9+rke2r2

Managed: 2024/03/15 17:32 UTC

Kube-system namespace UID: [redacted]

ACP Version: [redacted]

Private route identifier: [redacted]...

Cloud instance: private

Default bucket: astra-bucket1 (inherited)

For more information

- Astra Trident upgrades documentation

Unmanage apps and clusters

Remove any apps or clusters that you no longer want to manage from Astra Control.

Stop managing an app

Stop managing apps that you no longer want to back up, snapshot, or clone from Astra Control.

When you unmanage an app:

- Any existing backups and snapshots will be deleted.
- Applications and data remain available.

Steps

1. From the left navigation bar, select **Applications**.
2. Select the app.

3. From the Options menu in the Actions column, select **Unmanage**.
4. Review the information.
5. Type "unmanage" to confirm.
6. Select **Yes, Unmanage Application**.

Result

Astra Control stops managing the app.

Stop managing a cluster

Stop managing the cluster that you no longer want to manage from Astra Control.

> ⓘ Before you unmanage the cluster, you should unmanage the apps associated with the cluster.

As a best practice, we recommend that you remove the cluster from Astra Control before you delete it through GCP.

When you unmanage a cluster:

- This action stops your cluster from being managed by Astra Control. It doesn’t make any changes to the cluster’s configuration and it doesn’t delete the cluster.
- Astra Control Provisioner or Astra Trident won’t be uninstalled from the cluster. Learn how to uninstall Astra Trident.

Steps

1. Select **Clusters**.
2. Select the checkbox for the cluster that you no longer want to manage.
3. From the options menu in the **Actions** column, select **Unmanage**.
4. Confirm that you want to unmanage the cluster and then select **Yes, unmanage**.

Result

The status of the cluster changes to **Removing**. After that, the cluster will be removed from the **Clusters** page and it is no longer managed by Astra Control.

Deleting clusters from your cloud provider

Before you delete a Kubernetes cluster that has persistent volumes (PV) residing on NetApp storage classes, you need to first delete the persistent volume claims (PVC) following one of the methods below. Deleting the PVC and PV before deleting the cluster ensures that you don’t receive unexpected bills from your cloud provider.

- **Method #1**: Delete the application workload namespaces from the cluster. Do *not* delete the Trident namespace.
- **Method #2**: Delete the PVCs and the pods, or the deployment where the PVs are mounted.

When you manage a Kubernetes cluster from Astra Control, applications on that cluster use your cloud provider as the storage backend for persistent volumes. If you delete the cluster from your cloud provider without first removing the PVs, the backend volumes are *not* deleted along with the cluster.

Using one of the above methods will delete the corresponding PVs from your cluster. Make sure that there are no PVs residing on NetApp storage classes on the cluster before you delete it.

If you didn’t delete the persistent volumes before you deleted the cluster, then you’ll need to manually delete the backend volumes from your cloud provider.

Deploy a self-managed instance of Astra Control

If you want a self-managed instance of Astra Control that resides inside your network, you can deploy Astra Control Center directly from Astra Control Service.

Steps

1. In the Getting Started area of the Dashboard, select **Deploy a self-managed instance of Astra Control**.
2. Do one of the following:
   - Generate a new API token by selecting **Generate**.
   - Paste in an existing Astra Control REST API token. Refer to the Astra Automation documentation for guidance on generating an API token.
3. Follow the instructions in the **Deploy Astra Control Center** window.

Copyright information

Trademark information