



# VMware Tanzu with NetApp

## NetApp Solutions

NetApp  
May 17, 2024

# Table of Contents

- NVA-1166: VMware Tanzu with NetApp ..... 1
  - Use cases ..... 1
  - Business value ..... 1
  - Technology overview ..... 2
  - Current support matrix for validated releases ..... 3
  - VMware Tanzu overview ..... 4
  - NetApp storage systems overview ..... 8
  - NetApp storage integration overview ..... 9
  - Videos and demos: VMware Tanzu with NetApp ..... 36
  - Additional Information: VMware Tanzu with NetApp ..... 37

# NVA-1166: VMware Tanzu with NetApp

Alan Cowles and Nikhil M Kulkarni, NetApp

This reference document provides deployment validation of different flavors of VMware Tanzu Kubernetes solutions, deployed either as Tanzu Kubernetes Grid (TKG), Tanzu Kubernetes Grid Service (TKGS), or Tanzu Kubernetes Grid Integrated (TKGI) in several different data center environments as validated by NetApp. It also describes storage integration with NetApp storage systems and the Astra Trident storage orchestrator for the management of persistent storage and Astra Control Center for the backup and cloning of the stateful applications using that persistent storage. Lastly, the document provides video demonstrations of the solution integrations and validations.

## Use cases

The VMware Tanzu with NetApp solution is architected to deliver exceptional value for customers with the following use cases:

- Easy to deploy and manage VMware Tanzu Kubernetes Grid offerings deployed on VMware vSphere and integrated with NetApp storage systems.
- The combined power of enterprise container and virtualized workloads with VMware Tanzu Kubernetes Grid offerings.
- Real world configuration and use cases highlighting the features of VMware Tanzu when used with NetApp storage and the NetApp Astra suite of products.
- Application-consistent protection or migration of containerized workloads deployed on VMware Tanzu Kubernetes Grid clusters whose data resides on NetApp storage systems using Astra Control Center.

## Business value

Enterprises are increasingly adopting DevOps practices to create new products, shorten release cycles, and rapidly add new features. Because of their innate agile nature, containers and microservices play a crucial role in supporting DevOps practices. However, practicing DevOps at a production scale in an enterprise environment presents its own challenges and imposes certain requirements on the underlying infrastructure, such as the following:

- High availability at all layers in the stack
- Ease of deployment procedures
- Non-disruptive operations and upgrades
- API-driven and programmable infrastructure to keep up with microservices agility
- Multitenancy with performance guarantees
- Ability to run virtualized and containerized workloads simultaneously
- Ability to scale infrastructure independently based on workload demands
- Ability to deploy in a hybrid-cloud model with containers running in both on-premises data centers as well as in the cloud.

VMware Tanzu with NetApp acknowledges these challenges and presents a solution that helps address each

concern by deploying VMware Tanzu Kubernetes offerings in the customer's choice of hybrid cloud environment.

## Technology overview

The VMware Tanzu with NetApp solution is comprised of the following major components:

### VMware Tanzu Kubernetes platforms

VMware Tanzu comes in a variety of flavors that the solutions engineering team at NetApp has validated in our labs. Each Tanzu release successfully integrates with the NetApp storage portfolio, and each can help meet certain infrastructure demands. The following bulleted highlights describe the features and offerings of each version of Tanzu described in this document.

#### VMware Tanzu Kubernetes Grid (TKG)

- Standard upstream Kubernetes environment deployed in a VMware vSphere environment.
- Formerly known as Essential PKS (from Heptio acquisition, Feb 2019).
- TKG is deployed with a separate management cluster instance for support on vSphere 6.7U3 onward.
- TKG deployments can be deployed in the cloud as well with AWS or Azure.
- Allows for use of Windows or Linux worker nodes (Ubuntu/Photon).
- NSX-T, HA Proxy, AVI networking, or load balancers can be used for control plane.
- TKG supports MetalLB for the application/data plane.
- Can use vSphere CSI as well as third party CSIs like NetApp Astra Trident.

#### VMware Tanzu Kubernetes Grid Service (TKGS)

- Standard upstream Kubernetes environment deployed in a VMware vSphere environment.
- Formerly known as Essential PKS (from Heptio acquisition, Feb 2019).
- TKGS deployed with supervisor cluster and workload clusters only on vSphere 7.0U1 onward.
- Allows for use of Windows or Linux worker nodes (Ubuntu/Photon).
- NSX-T, HA Proxy, AVI networking, or load balancers can be used for control plane.
- TKGS supports MetalLB for application/data plane.
- Can use vSphere CSI as well as third party CSIs like NetApp Astra Trident.
- Provides support for vSphere Pods with Tanzu, allowing pods to run directly on enabled ESXi hosts in the environment.

#### VMware Tanzu Kubernetes Grid Integrated (TKGI)

- Formerly known as Enterprise PKS (from Heptio acquisition, Feb 2019).
- Can use NSX-T, HA Proxy, or Avi. You can also provide your own load balancer.
- Supported from vSphere 6.7U3 onward, as well as AWS, Azure, and GCP.
- Setup via wizard to allow for ease of deployment.
- Runs Tanzu in controlled immutable VMs managed by BOSH.
- Can make use vSphere CSI as well as third party CSIs like NetApp Astra Trident (some conditions apply).

## vSphere with Tanzu (vSphere Pods)

- vSphere-native pods run in a thin, photon-based layer with prescribed virtual hardware for complete isolation.
- Requires NSX-T, but that allows for additional feature support such as a Harbor image registry.
- Deployed and managed in vSphere 7.0U1 onward using a virtual Supervisor cluster like TKGS. Runs pods directly on ESXi nodes.
- Fully vSphere integrated, highest visibility and control by vSphere administration.
- Isolated CRX-based pods for the highest level of security.
- Only supports vSphere CSI for persistent storage. No third-party storage orchestrators supported.

## NetApp storage systems

NetApp has several storage systems perfect for enterprise data centers and hybrid cloud deployments. The NetApp portfolio includes NetApp ONTAP, NetApp Element, and NetApp e-Series storage systems, all of which can provide persistent storage for containerized applications.

For more information, visit the NetApp website [here](#).

## NetApp storage integrations

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

For more information, visit the NetApp Astra website [here](#).

Astra Trident is an open-source, fully-supported storage orchestrator for containers and Kubernetes distributions, including VMware Tanzu.

For more information, visit the Astra Trident website [here](#).

## Current support matrix for validated releases

Technology	Purpose	Software version
NetApp ONTAP	Storage	9.9.1
NetApp Astra Control Center	Application Aware Data Management	22.04
NetApp Astra Trident	Storage Orchestration	22.04.0
VMware Tanzu Kubernetes Grid	Container orchestration	1.4+
VMware Tanzu Kubernetes Grid Service	Container orchestration	0.0.15 [vSphere Namespaces]
		1.22.6 [Supervisor Cluster Kubernetes]
VMware Tanzu Kubernetes Grid Integrated	Container orchestration	1.13.3
VMware vSphere	Data center virtualization	7.0U3

VMware NSX-T Data Center	Networking and Security	3.1.3
VMware NSX Advanced Load Balancer	Load Balancer	20.1.3

## VMware Tanzu overview

VMware Tanzu is a portfolio of products that enables enterprises to modernize their applications and the infrastructure they run on. VMware Tanzu’s full stack of capabilities unites the development and IT operations teams on a single platform to embrace modernization in both their applications and their infrastructure consistently across on-premises and hybrid cloud environments to continuously deliver better software to production.



To understand more about the different offerings and their capabilities in the Tanzu portfolio, visit the documentation [here](#).

Regarding Tanzu’s Kubernetes Operations catalog, VMware has a variety of implementations for Tanzu Kubernetes Grid, all of which provision and manage the lifecycle of Tanzu Kubernetes clusters on a variety of platforms. A Tanzu Kubernetes cluster is a full-fledged Kubernetes distribution that is built and supported by VMware.

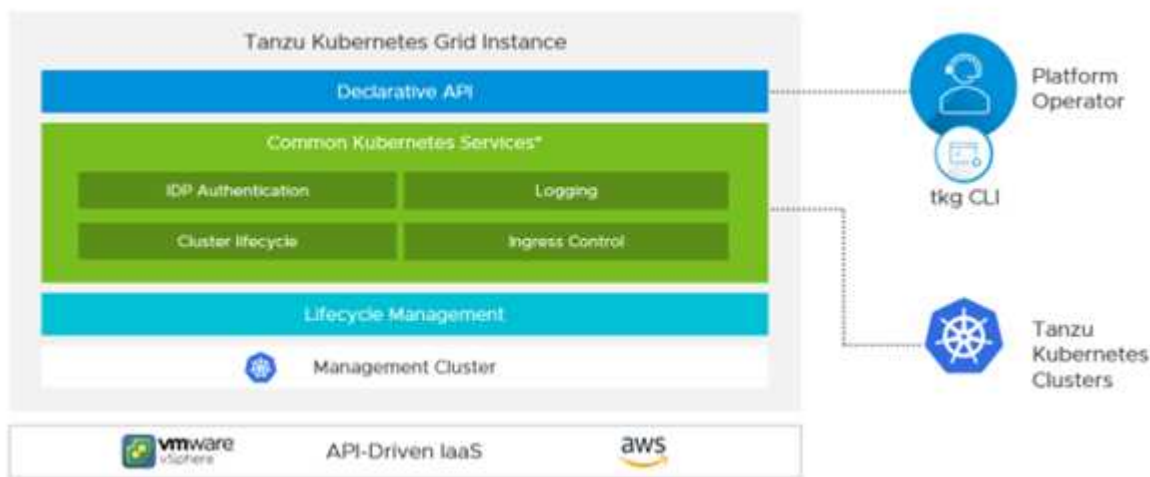
NetApp has tested and validated the deployment and interoperability of the following products from the VMware Tanzu portfolio in its labs:

- [VMware Tanzu Kubernetes Grid \(TKG\)](#)
- [VMware Tanzu Kubernetes Grid Service \(TKGS\)](#)
- [VMware Tanzu Kubernetes Grid Integrated \(TKGI\)](#)
- [VMware vSphere with Tanzu \(vSphere Pods\)](#)

## VMware Tanzu Kubernetes Grid (TKG) overview

VMware Tanzu Kubernetes Grid, also known as TKG, lets you deploy Tanzu Kubernetes clusters across hybrid cloud or public cloud environments. TKG is installed as a management cluster, which is a Kubernetes cluster itself, that deploys and operates the Tanzu Kubernetes clusters. These Tanzu Kubernetes clusters are the workload Kubernetes clusters on which the actual workload is deployed.

Tanzu Kubernetes Grid builds on a few of the promising upstream community projects and delivers a Kubernetes platform that is developed, marketed, and supported by VMware. In addition to Kubernetes distribution, Tanzu Kubernetes Grid provides additional add-ons that are essential production-grade services such as registry, load balancing, authentication, and so on. VMware TKG with management cluster is widely used in vSphere 6.7 environments, and, even though it is supported, it is not a recommended deployment for vSphere 7 environments because TKGS has native integration capabilities with vSphere 7.



For more information on Tanzu Kubernetes Grid, refer to the documentation [here](#).

Depending on whether the Tanzu Kubernetes Grid is being installed on-premises on vSphere cluster or in cloud environments, prepare and deploy Tanzu Kubernetes Grid by following the installation guide [here](#).

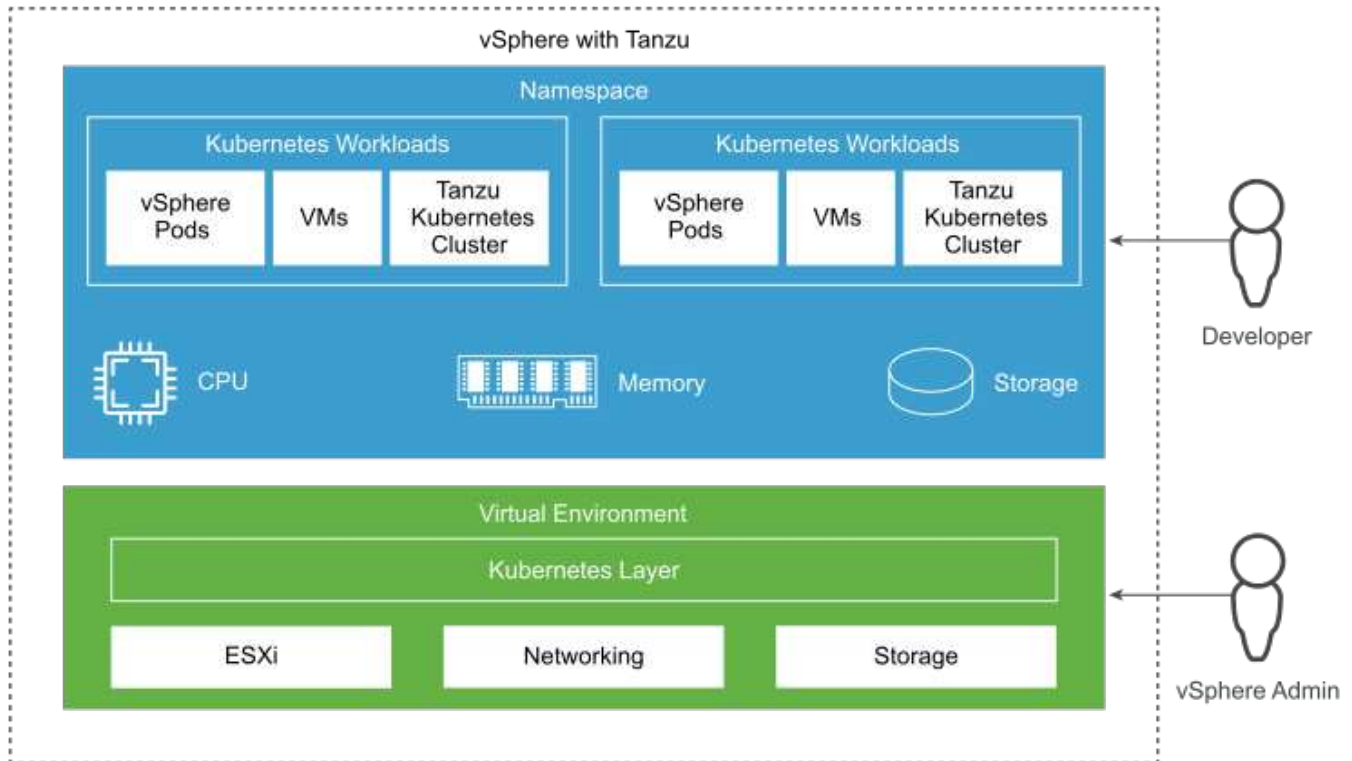
After you have installed the management cluster for Tanzu Kubernetes Grid, deploy the user clusters or workload clusters as needed by following the documentation [here](#). VMware TKG management cluster requires that an SSH key be provided for installation and operation of Tanzu Kubernetes clusters. This key can be used to log into the cluster nodes using the `capv` user.

## VMware Tanzu Kubernetes Grid Service (TKGS) overview

VMware Tanzu Kubernetes Grid Service (also known as vSphere with Tanzu) lets you create and operate Tanzu Kubernetes clusters natively in vSphere and also allows you to run some smaller workloads directly on the ESXi hosts. It allows you to transform vSphere into a platform for running containerized workloads natively on the hypervisor layer. Tanzu Kubernetes Grid Service deploys a supervisor cluster on vSphere when enabled that deploys and operates the clusters required for the workloads. It is natively integrated with vSphere 7 and leverages many reliable vSphere features like vCenter SSO, Content Library, vSphere networking, vSphere storage, vSphere HA and DRS, and

## vSphere security for a more seamless Kubernetes experience.

vSphere with Tanzu offers a single platform for hybrid application environments where you can run your application components either in containers or in VMs, thus providing better visibility and ease of operations for developers, DevOps engineers, and vSphere administrators. VMware TKGS is only supported with vSphere 7 environments and is the only offering in Tanzu Kubernetes operations portfolio that allows you to run pods directly on ESXi hosts.



For more information on Tanzu Kubernetes Grid Service, follow the documentation [here](#).

There are a lot of architectural considerations regarding feature sets, networking, and so on. Depending on the architecture chosen, the prerequisites and the deployment process of Tanzu Kubernetes Grid Service differ. To deploy and configure Tanzu Kubernetes Grid Service in your environment, follow the guide [here](#). Furthermore, to log into the Tanzu Kubernetes cluster nodes deployed via TKGS, follow the procedure laid out in this [link](#).

NetApp recommends that all the production environments be deployed in multiple master deployments for fault tolerance with the choice of worker nodes' configuration to meet the requirements of the intended workloads. Thus, a recommended VM class for a highly intensive workload would have at least four vCPUs and 12GB of RAM.

When Tanzu Kubernetes clusters are created in a namespace, users with `owner` or `edit` permission can create pods directly in any namespace by using the user account. This is because users with the `owner` or `edit` permission are allotted the cluster administrator role. However, when creating deployments, daemon sets, stateful sets, or others in any namespace, you must assign a role with the required permissions to the corresponding service accounts. This is required because the deployments or daemon sets utilize service accounts to deploy the pods.

See the following example of ClusterRoleBinding to assign the cluster administrator role to all service accounts in the cluster:



```

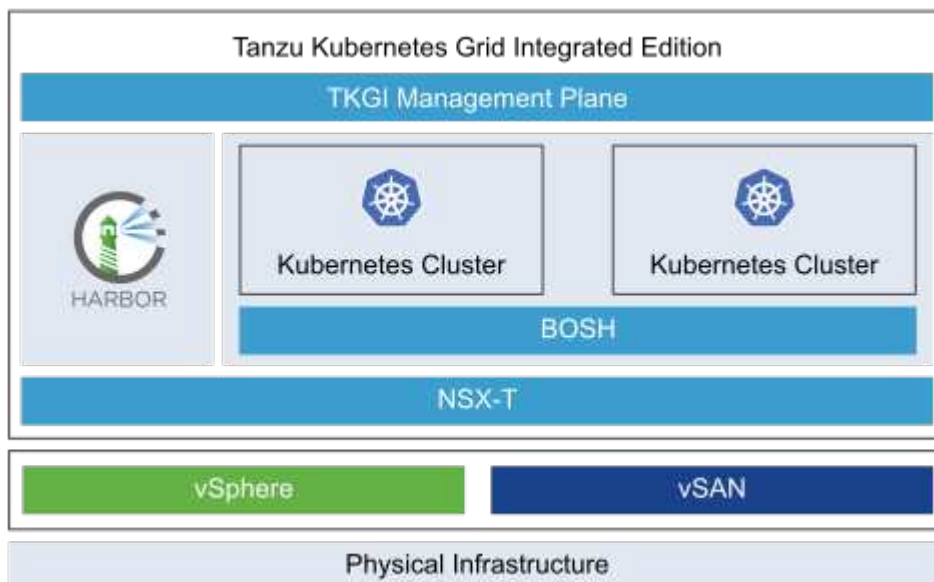
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: all_sa_ca
subjects:
- kind: Group
  name: system:serviceaccounts
  namespace: default
roleRef:
  kind: ClusterRole
  name: psp:vmware-system-privileged
  apiGroup: rbac.authorization.k8s.io

```

## VMware Tanzu Kubernetes Grid Integrated Edition (TKGI) overview

VMware Tanzu Kubernetes Grid Integrated (TKGI) Edition, formerly known as VMware Enterprise PKS, is a standalone container orchestration platform based on Kubernetes with capabilities such as life cycle management, cluster health monitoring, advanced networking, a container registry, and so on. TKGI provisions and manages Kubernetes clusters with the TKGI control plane, which consists of BOSH and Ops Manager.

TKGI can be installed and operated either on vSphere or OpenStack environments on-premises or in any of the major public clouds on their respective IaaS offerings. Furthermore, the integration of TKGI with NSX-T and Harbour enables wider use cases for enterprise workloads. To know more about TKGI and its capabilities, visit the documentation [here](#).



TKGI is installed in a variety of configurations on a variety of platforms based on different use-cases and designs. Follow the guide [here](#) to install and configure TKGI and its prerequisites. TKGI uses Bosh VMs as nodes for Tanzu Kubernetes clusters which run immutable configuration images and any manual changes on Bosh VMs do not remain persistent across reboots.

## Important notes:

- NetApp Trident requires privileged container access. So, during TKGI installation, make sure to select the Enable Privileged Containers checkbox in the step to configure Tanzu Kubernetes cluster node plans.

The screenshot shows the configuration interface for a Tanzu Kubernetes cluster node plan. It is divided into several sections:

- Worker Node Instances:** Set to 3.
- Worker Persistent Disk Size:** Set to 50 GB.
- Worker Availability Zones:** Enabled (checked) with 'az' selected.
- Worker VM Type:** Set to 'medium.disk (cpu: 2, ram: 4 GB, disk: 32 GB)'.
- Max Worker Node Instances:** Set to 50.
- Errand VM Type:** Set to 'medium.disk (cpu: 2, ram: 4 GB, disk: 32 GB)'.
- Enable Privileged Containers (Use with caution):** Checked.
- Admission Plugins:**
  - PodSecurityPolicy: Disabled.
  - SecurityContextDeny: Disabled.
- Cluster Services:**
  - Force node to drain even if it has running pods not managed by a ReplicationController, ReplicaSet, Job, DaemonSet or Stateful Set: Enabled.
  - Force node to drain even if it has running DaemonSet managed pods: Enabled.
  - Force node to drain even if it has running pods using emptyDir: Enabled.
  - Force node to drain even if pods are still running after timeout: Disabled.
- Node Drain Timeout (minutes, min: 0, max: 1440):** Set to 0.
- Pod Shutdown Grace Period (seconds, min: -1, max: 86400):** Set to 10.

At the bottom, there are two buttons: 'SAVE PLAN' (highlighted in blue) and 'DELETE'.

- NetApp recommends that all production environments be deployed in multiple master deployments for fault tolerance with the choice of worker nodes' configuration to meet the requirements of the intended workloads. Thus, a recommended TKGI cluster plan would consist of at least three masters and three workers with at least four vCPUs and 12GB of RAM for a highly intensive workload.

## NetApp storage systems overview

NetApp has several storage platforms that are qualified with Astra Trident and Astra Control to provision, protect and manage data for containerized applications and thus help in defining and maximizing DevOps throughput.

Unresolved directive in containers/vtwn\_overview\_netapp.adoc - include::.../\_include/containers\_common\_intro\_sections.adoc[tags=netapp\_overview\_page;!netapp\_overview\_page\_element]

### NetApp ONTAP

NetApp ONTAP is a powerful storage-software tool with capabilities such as an intuitive GUI, REST APIs with automation integration, AI-informed predictive analytics and corrective action, non-disruptive hardware upgrades, and cross-storage import.

Unresolved directive in containers/vtwn\_netapp\_ontap.adoc - include::.../\_include/containers\_common\_intro\_sections.adoc[tags=netapp\_ontap\_page]

# NetApp storage integration overview

NetApp provides a number of products which assist our customers with orchestrating and managing persistent data in container based environments.

Unresolved directive in containers/vtwn\_overview\_storint.adoc -  
include::.../../\_include/containers\_common\_intro\_sections.adoc[tags=storage\_integration\_overview]

## NetApp Astra Control overview

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

Unresolved directive in containers/vtwn\_overview\_astra.adoc -  
include::.../../\_include/containers\_common\_intro\_sections.adoc[tags=astra\_cc\_overview]

### Astra Control Center automation

Astra Control Center has a fully functional REST API for programmatic access. Users can use any programming language or utility to interact with Astra Control REST API endpoints. To learn more about this API, see the documentation [here](#).

If you are looking for a ready-made software development toolkit for interacting with Astra Control REST APIs, NetApp provides a toolkit with the Astra Control Python SDK that you can download [here](#).

If programming is not appropriate for your situation and you would like to use a configuration management tool, you can clone and run the Ansible playbooks that NetApp publishes [here](#).

### Astra Control Center installation prerequisites

Astra Control Center installation requires the following prerequisites:

- One or more Tanzu Kubernetes clusters, managed either by a management cluster or TKGS or TKGI. TKG workload clusters 1.4+ and TKGI user clusters 1.12.2+ are supported.
- Astra Trident must already be installed and configured on each of the Tanzu Kubernetes clusters.
- One or more NetApp ONTAP storage systems running ONTAP 9.5 or greater.



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

- A Trident storage backend must be configured on each Tanzu Kubernetes cluster with an SVM backed by an ONTAP cluster.
- A default StorageClass configured on each Tanzu Kubernetes cluster with Astra Trident as the storage provisioner.
- A load balancer must be installed and configured on each Tanzu Kubernetes cluster for load balancing and exposing Astra Control Center if you are using ingressType `AccTraefik`.

- An ingress controller must be installed and configured on each Tanzu Kubernetes cluster for exposing Astra Control Center if you are using `ingressType Generic`.
- A private image registry must be configured to host the NetApp Astra Control Center images.
- You must have Cluster Admin access to the Tanzu Kubernetes cluster where Astra Control Center is being installed.
- You must have Admin access to NetApp ONTAP clusters.
- A RHEL or Ubuntu admin workstation.

## Install Astra Control Center

This solution describes an automated procedure for installing Astra Control Center using Ansible playbooks. If you are looking for a manual procedure to install Astra Control Center, follow the detailed installation and operations guide [here](#).

Unresolved directive in containers/vtwn\_overview\_astra.adoc -  
include::.../../\_include/containers\_astra\_cc\_install\_ansible.adoc[Install Astra Control Center using Ansible]

### Post Install Steps

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

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

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

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



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

```
{"level":"info","ts":1624054318.029971,"logger":"controllers.AstraControlCenter","msg":"Successfully Reconciled AstraControlCenter in [seconds]s","AstraControlCenter":"netapp-astra-cc/astra","ae.Version":"[22.04.0]"}
```

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

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



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

4. Get the traefik service load balancer IP if the ingressType is AccTraefik.

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

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

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

**New Host**

Name (uses parent domain name if blank):  
astra-control-center

Fully qualified domain name (FQDN):  
astra-control-center.cie.netapp.com.

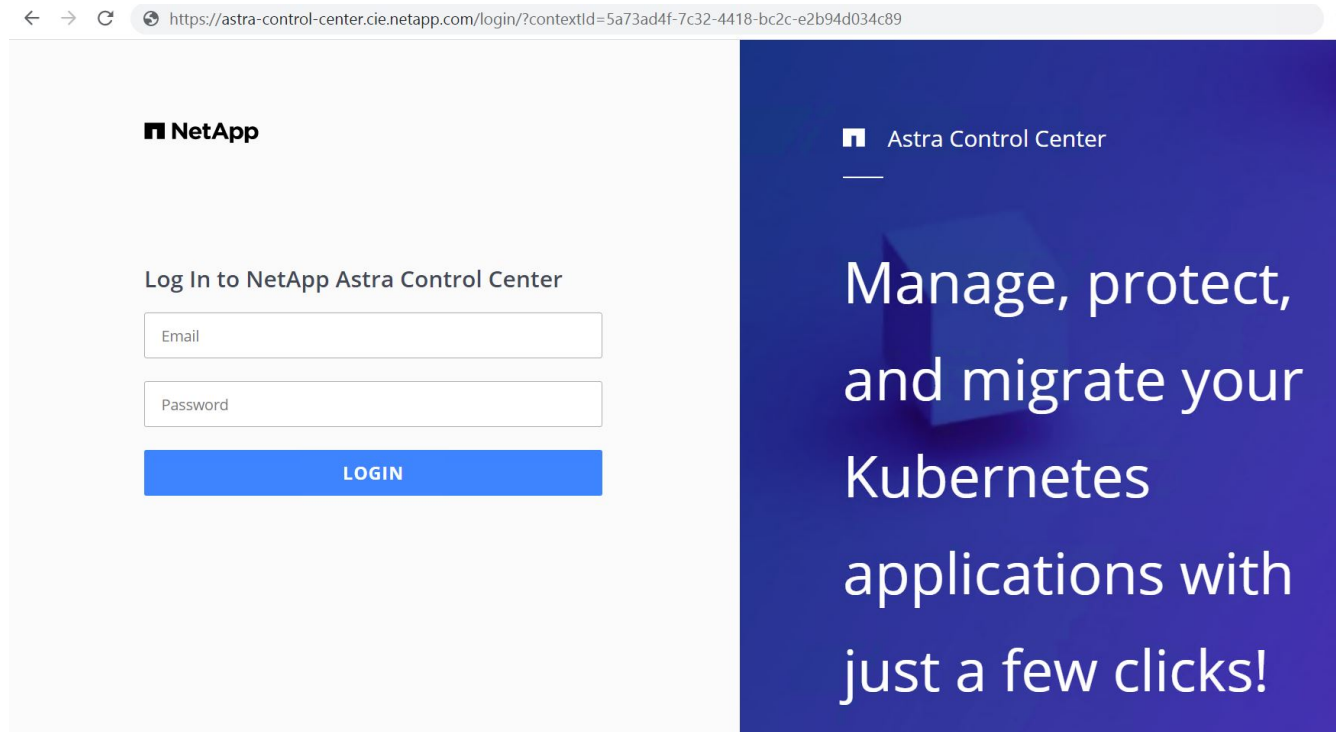
IP address:  
10.61.186.181

Create associated pointer (PTR) record

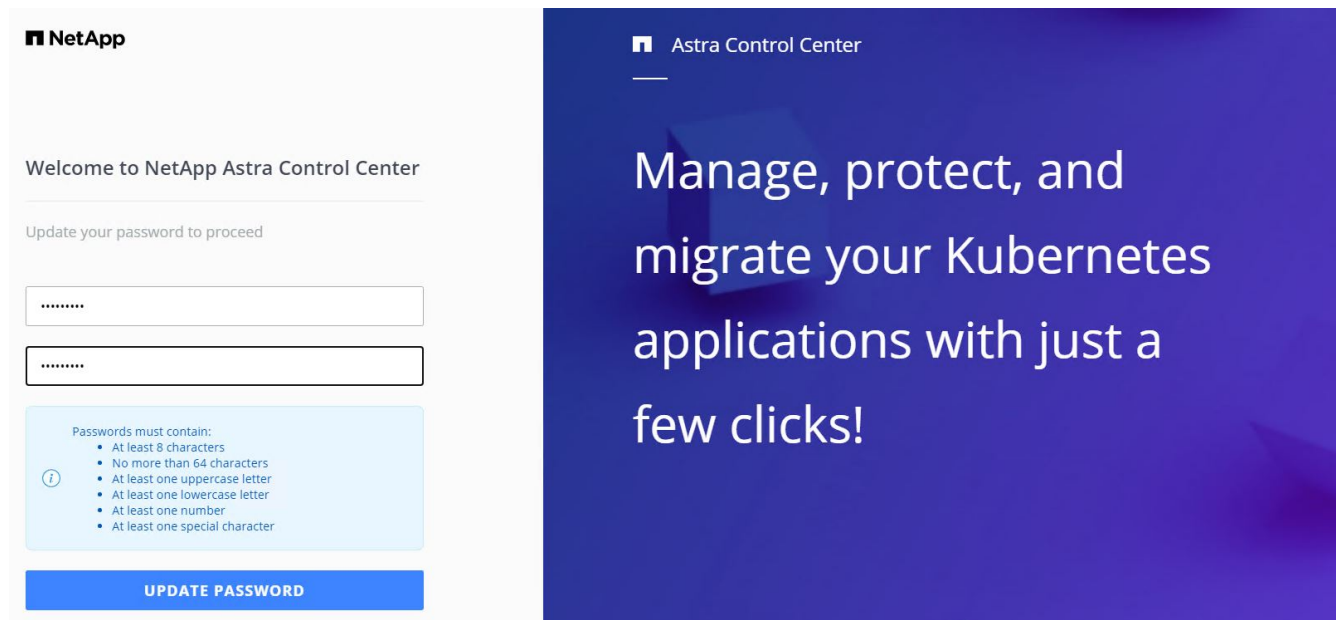
Allow any authenticated user to update DNS records with the same owner name

Add Host Cancel

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



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



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

**Add user**

**USER DETAILS**

First name: Nikhil

Last name: Kulkarni

Email address: tme\_nik@netapp.com

**PASSWORD**

Temporary password: .....

Confirm temporary password: .....

Passwords must contain:

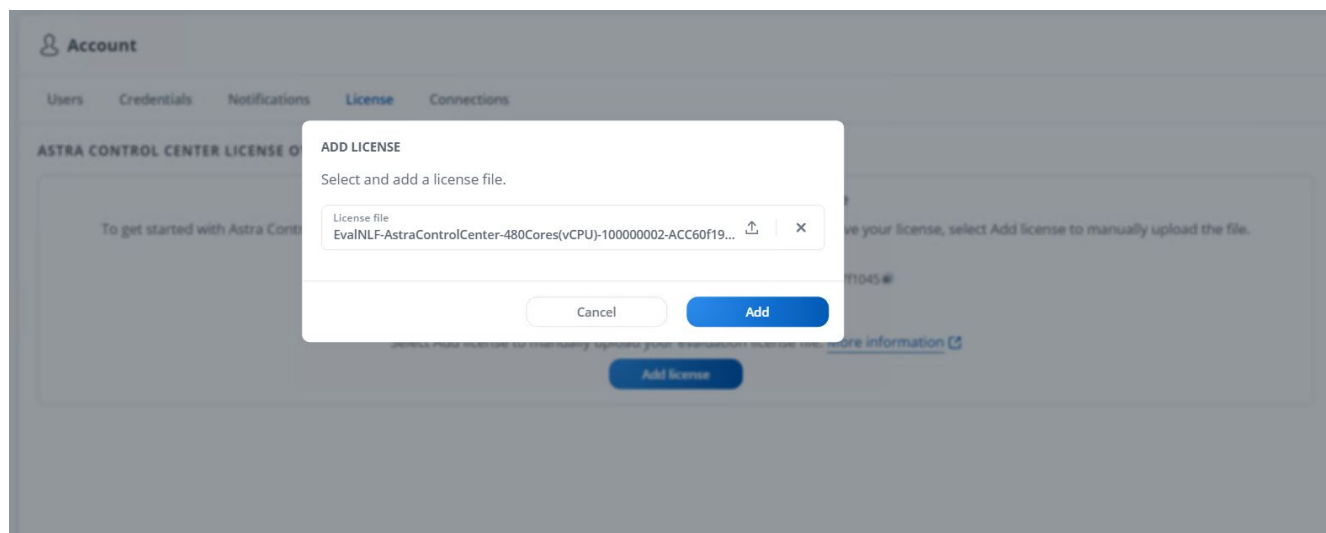
- At least 8 characters
- No more than 64 characters
- At least one lowercase letter
- At least one uppercase letter
- At least one number
- At least one special character

**USER ROLE**

Role: Owner

Buttons: Cancel, Add ✓

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



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

## Register your VMware Tanzu Kubernetes Clusters with the Astra Control Center

To enable the Astra Control Center to manage your workloads, you must first register your Tanzu Kubernetes clusters.



## Register VMware Tanzu Kubernetes clusters

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

**Add Kubernetes cluster** STEP 1/3: CREDENTIALS

**CREDENTIALS**

Provide Astra Control access to your Kubernetes and OpenShift clusters by entering a kubeconfig credential.

Follow [instructions](#) on how to create a dedicated admin-role kubeconfig.

[Upload file](#) Paste from clipboard

Kubeconfig YAML file  
tkgi-kubeconfig.txt

Credential name  
tkgi-acc

Cancel Next →

**ADDING CLUSTERS**

Adding a cluster allows Astra Control to install its storage services, and enable data management operations on your containerized applications.

For more details on required versions or cloud specific setup refer to the documentation.

Read more in [Adding clusters](#).

2. Astra Control Center detects the eligible storage classes. Now select the way that storageclass provisions volumes using Trident backed by an SVM on NetApp ONTAP and click Review. In the next pane, verify the details and click Add Cluster.
3. When the cluster is added, it moves to the Discovering status while Astra Control Center inspects it and installs the necessary agents. The cluster status changes to `Healthy` after it is successfully registered.

**Clusters**

Actions + Add Kubernetes cluster Search

1-1 of 1 entries

Name ↓	State	Type	Version	Actions
<a href="#">tkgi-acc</a>	Healthy	Kubernetes	v1.22.6+vmware.1	



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

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



## Backends

<a href="#">+ Add</a>		<input type="text" value="Search"/>					
1-1 of 1 entries							
Name ↓	State	Capacity	Throughput	Type	Cluster	Cloud	Actions
172.21.224.201(trident)	<a href="#">Discovered</a>	Not available yet	Not available yet	ONTAP	Not applicable	Not applicable	

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

### Manage ONTAP storage backend

STEP 1/2: CREDENTIALS ✕

**CREDENTIALS**

Enter cluster administrator credentials for the ONTAP storage backend you want to manage.


Cluster management IP address 172.21.224.201	User name admin	Password .....
---	--------------------	-------------------

**MANAGING STORAGE BACKENDS**

Storage backends provide storage to your Kubernetes applications.

Managing storage clusters in Astra Control as a storage backend will allow you to get linkages between PVs and the storage backend. You will also see capacity and health details of the storage backend, including performance metrics if Astra Control is connected to Cloud Insights.

Read more in [Storage type](#).

 ONTAP

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

## Backends

<a href="#">+ Add</a>		Search		<a href="#">★</a> <a href="#">Q</a>			
1-1 of 1 entries							<a href="#">&lt;</a> <a href="#">&gt;</a>
Name ↓	State	Capacity	Throughput	Type	Cluster	Cloud	Actions
<a href="#">K8s-Ontap</a>	<span>✓</span> Available	Not available yet	Not available yet	ONTAP 9.9.1	Not applicable	Not applicable	<a href="#">⋮</a>

- For backup and restore across Tanzu Kubernetes clusters using Astra Control Center, you must provision an object storage bucket that supports the S3 protocol. Currently supported options are ONTAP S3, StorageGRID, AWS S3, and Microsoft Azure Blob storage. For the purpose of this installation, we are going to configure an AWS S3 bucket. Go to Buckets, click Add bucket, and select Generic S3. Enter the details about the S3 bucket and credentials to access it, click the checkbox Make this Bucket the Default Bucket for the Cloud, and then click Add.

### Add bucket ✕

Enter the access details of your existing object store bucket to allow Astra Control to store your application backups.

Type <input type="text" value="Generic S3"/>	Existing bucket name <input type="text" value="na-tanzu-astra/na-astra-tkji"/>
Description (optional) <input type="text"/>	S3 server name or IP address <input type="text" value="s3.us-east-1.amazonaws.com"/>

Make this bucket the default bucket for this cloud ?

---

#### SELECT CREDENTIALS

Astra Control requires S3 access credentials with the roles necessary to facilitate Kubernetes application data management.

Add [Use existing](#)

Select credential

[Cancel](#) [Add ✓](#)

#### BUCKETS

Astra Control stores backups in your existing object store buckets. The first bucket added for a selected cloud will be designated as the default bucket for backup and clone operations.

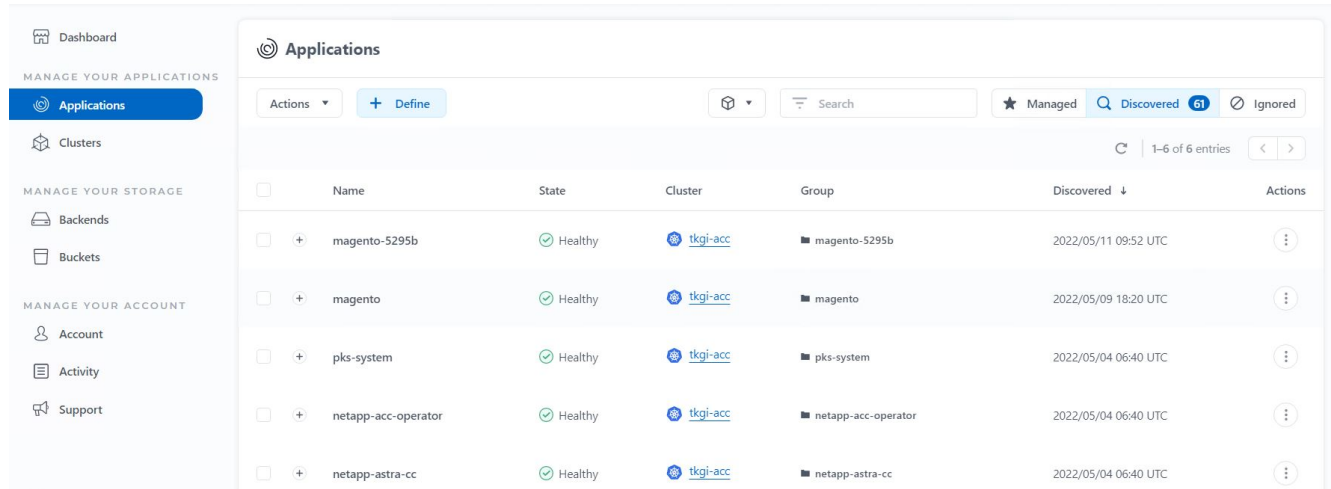
Read more in [Storage buckets](#).

## Choose the applications to protect

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

## Manage applications

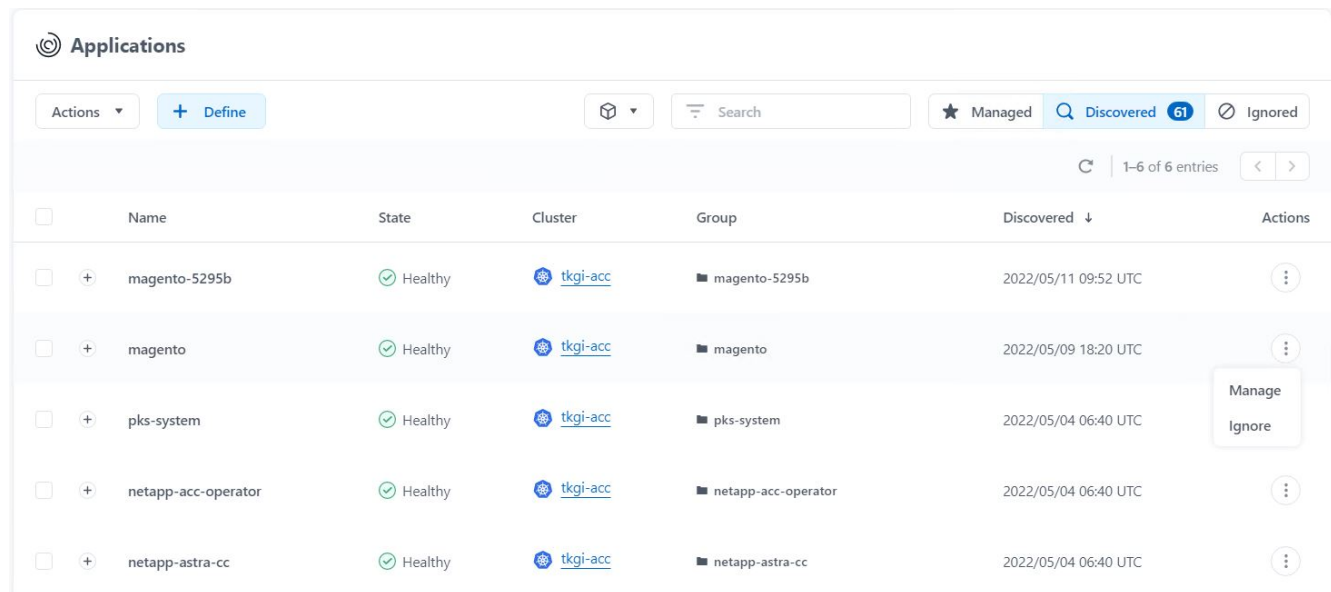
1. After the Tanzu Kubernetes clusters and ONTAP backends are registered with the Astra Control Center, the control center automatically starts discovering the applications in all the namespaces that are using the storageclass configured with the specified ONTAP backend.



The screenshot shows the 'Applications' page in the Astra Control Center. The left sidebar contains navigation options: Dashboard, Applications (selected), Clusters, Backends, Buckets, Account, Activity, and Support. The main content area displays a table of discovered applications. The table has columns for Name, State, Cluster, Group, Discovered, and Actions. The 'Discovered' column shows a count of 61. The table lists six applications, all with a 'Healthy' state.

Name	State	Cluster	Group	Discovered	Actions
magento-5295b	Healthy	tkgi-acc	magento-5295b	2022/05/11 09:52 UTC	
magento	Healthy	tkgi-acc	magento	2022/05/09 18:20 UTC	
pkc-system	Healthy	tkgi-acc	pkc-system	2022/05/04 06:40 UTC	
netapp-acc-operator	Healthy	tkgi-acc	netapp-acc-operator	2022/05/04 06:40 UTC	
netapp-astra-cc	Healthy	tkgi-acc	netapp-astra-cc	2022/05/04 06:40 UTC	

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



This screenshot shows the same 'Applications' page as above, but with the 'Actions' dropdown menu for the 'magento' application open. The dropdown menu shows two options: 'Manage' and 'Ignore'. The 'Manage' option is highlighted.

Name	State	Cluster	Group	Discovered	Actions
magento-5295b	Healthy	tkgi-acc	magento-5295b	2022/05/11 09:52 UTC	
magento	Healthy	tkgi-acc	magento	2022/05/09 18:20 UTC	Manage Ignore
pkc-system	Healthy	tkgi-acc	pkc-system	2022/05/04 06:40 UTC	
netapp-acc-operator	Healthy	tkgi-acc	netapp-acc-operator	2022/05/04 06:40 UTC	
netapp-astra-cc	Healthy	tkgi-acc	netapp-astra-cc	2022/05/04 06:40 UTC	

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

Applications							
Actions ▾		+ Define		All clusters ▾		Search	
				★ Managed		🔍 Discovered 60	🚫 Ignored
							1-1 of 1 entries
<input type="checkbox"/>	Name	State	Protection	Cluster	Group	Discovered ↓	Actions
<input type="checkbox"/>	<a href="#">magento</a>	✔ Healthy	⚠ Unprotected	<a href="#">tkgi-acc</a>	■ magento	2022/05/09 18:20 UTC	⋮

## Protect your applications

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

### Create an application snapshot

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

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

The screenshot shows the Astra Control Center interface for a 'magento' application. The application status is 'Healthy' and protection is 'Unprotected'. An 'Actions' dropdown menu is open, showing options: Snapshot, Backup, Clone, Restore, and Unmanage.

**APPLICATION STATUS**  
 Healthy

**APPLICATION PROTECTION STATUS**  
 Unprotected

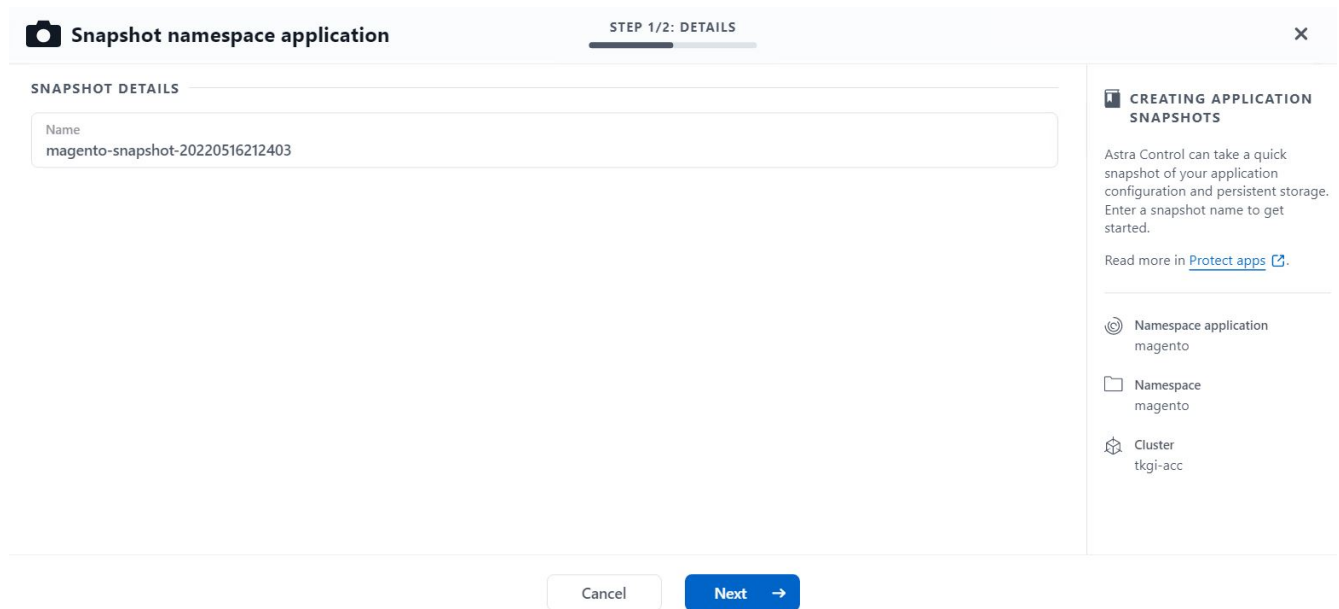
Images  
 docker.io/bitnami/elasticsearch:6.8.12-debian-10-r61  
 docker.io/bitnami/magento:2.4.1-debian-10-r14  
 docker.io/bitnami/mariadb:10.3.24-debian-10-r49

Protection schedule  
 Disabled

Group  
 ■ magento

Cluster  
 tkgi-acc

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



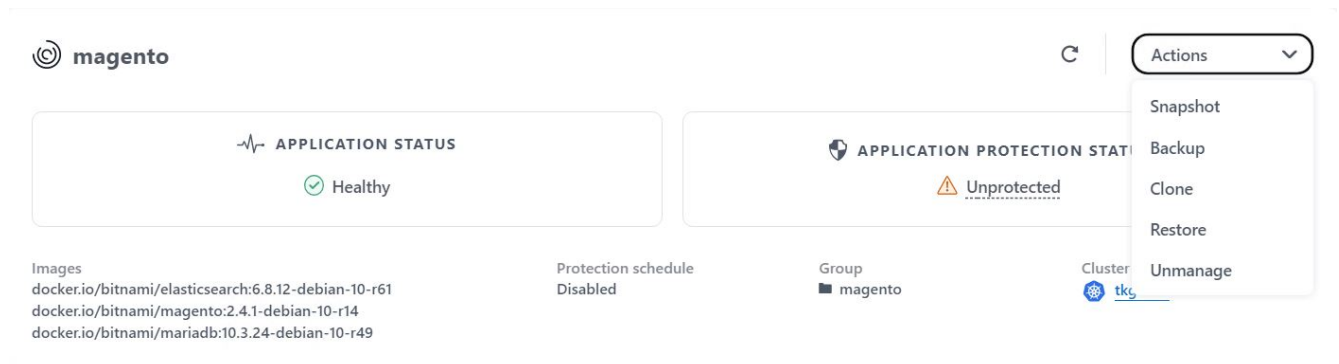
### Create an application backup

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

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

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

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



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

**Back up namespace application**
STEP 1/2: DETAILS
✕

---

**BACKUP DETAILS**

Name  
 magento-backup-20220516212622

Back up from an existing snapshot

**BACKUP DESTINATION**

Bucket  
 na-tanzu-astra/na-astra-tkgi Available Default

**CREATING APPLICATION BACKUPS**

Astra Control can take a backup of your application configuration and persistent storage. Persistent storage backups are transferred to your object store. Enter a backup name to get started.

Read more in [Application backups](#)

---

- Namespace application magento
- Namespace magento
- Cluster tkgi-acc

Cancel
Next →

## Restoring an application

At the push of a button, you can restore an application to the originating namespace in the same cluster or to a remote cluster for application protection and disaster recovery purposes.

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

**magento**
🔄
Actions ▼

**APPLICATION STATUS**

✔ Healthy

**APPLICATION PROTECTION STATUS**

⚠ Unprotected

Images

- docker.io/bitnami/elasticsearch:6.8.12-debian-10-r61
- docker.io/bitnami/magento:2.4.1-debian-10-r14
- docker.io/bitnami/mariadb:10.3.24-debian-10-r49

Protection schedule

Disabled

Group

■ magento

Cluster

tkgi

- Snapshot
- Backup
- Clone
- Restore
- Unmanage

2. Enter the name of the restore namespace, select the cluster you want to restore it to, and choose if you want to restore it from an existing snapshot or from a backup of the application. Click Next.

20

**Restore namespace application** STEP 1/2: DETAILS ✕

---

**RESTORE DETAILS**

Destination cluster: tkgi-acc | Destination namespace: magento

---

**RESTORE SOURCE**

Filter | Snapshots | Backups

Application backup	State	On-Schedule/On-Demand	Created ↑
<input type="radio"/> magento-backup-20220516212730	<span>Healthy</span>	<input checked="" type="radio"/> On-Demand	2022/05/16 21:27 UTC

---

**RESTORING APPLICATIONS**

Astra Control can restore your application configuration and persistent storage. Select a source snapshot or backup for the restored application.

- Namespace application magento
- Namespace magento
- Cluster tkgi-acc

3. On the review pane, enter `restore` and click Restore after you have reviewed the details.

**Restore namespace application** STEP 2/2: SUMMARY ✕

---

**REVIEW RESTORE INFORMATION**

**⚠️** All existing resources associated with this namespace application will be deleted and replaced with the source backup "magento-backup-20220516212730" taken on 2022/05/16 21:27 UTC. Persistent volumes will be deleted and recreated. External resources with dependencies on this namespace application might be impacted.

We recommend taking a snapshot or a backup of your namespace application before proceeding.

**BACKUP**  
magento-backup-20220516212730

**ORIGINAL GROUP**  
magento

**ORIGINAL CLUSTER**  
tkgi-acc

**RESOURCE LABELS**  
Config Maps  
app.kubernetes.io/name: elasticsearch +9  
Deployments

**RESTORE**  
magento

**DESTINATION GROUP**  
magento

**DESTINATION CLUSTER**  
tkgi-acc

**RESOURCE LABELS**  
Config Maps  
app.kubernetes.io/name: elasticsearch +9  
Deployments

Are you sure you want to restore the namespace application "magento"?

Type `restore` below to confirm.

Confirm to restore  
`restore`

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

Applications

Actions ▾ + Define All clusters ▾ Search Managed Discovered 60 Ignored

1-1 of 1 entries

<input type="checkbox"/>	Name	State	Protection	Cluster	Group	Discovered ↓	Actions
<input type="checkbox"/>	<a href="#">magento</a>	Healthy	Unprotected	tkgi-acc	magento	2022/05/09 18:20 UTC	

## Cloning an application

You can clone an application to the originating cluster or to a remote cluster for dev/test or application protection and disaster recovery purposes. Cloning an application within the same cluster on the same storage backend uses NetApp FlexClone technology, which clones the PVCs instantly and saves storage space.

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

magento

APPLICATION STATUS: Healthy

APPLICATION PROTECTION STATUS: Unprotected

Images: docker.io/bitnami/elasticsearch:6.8.12-debian-10-r61, docker.io/bitnami/magento:2.4.1-debian-10-r14, docker.io/bitnami/mariadb:10.3.24-debian-10-r49

Protection schedule: Disabled

Group: magento

Cluster: tksi

Actions: Snapshot, Backup, Clone, Restore, Unmanage

2. Enter the details of the new namespace, select the cluster you want to clone it to, and choose if you want to clone it from an existing snapshot, from a backup, or from the current state of the application. Click Next and then click Clone on the review pane after you have reviewed the details.



Clone namespace application
STEP 1/2: DETAILS
✕

**CLONE DETAILS**

Clone namespace  
magento-bef7f

Destination cluster  
 tkgi-acc

Clone from an existing snapshot or backup

**CLONING APPLICATIONS**

Astra Control can create a clone of your application configuration and persistent storage. Persistent storage backups are transferred from your object store, so choosing a clone from an existing backup will complete the fastest. Enter a clone name to get started.

Not all applications may support cloning.

Read more in [Clone applications](#).

- Namespace application magento
- Namespace magento
- Cluster tkgi-acc

Cancel
Next →

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

Applications

Actions ▾
+ Define
All clusters ▾

Search

★ Managed
🔍 Discovered **60**
🚫 Ignored

🔄
1-2 of 2 entries
< >

	Name	State	Protection	Cluster	Group	Discovered ↓	Actions
<input type="checkbox"/>	<a href="#">magento-bef7f</a>	<span style="color: green;">✔</span> Healthy	<span style="color: orange;">⚠</span> Unprotected	tkgi-acc	■ magento-bef7f	2022/05/16 21:31 UTC	⋮
<input type="checkbox"/>	<a href="#">magento</a>	<span style="color: green;">✔</span> Healthy	<span style="color: blue;">i</span> Partially protected	tkgi-acc	■ magento	2022/05/09 18:20 UTC	⋮

## Astra Trident overview

Astra Trident is an open-source and fully-supported storage orchestrator for containers and Kubernetes distributions, including VMware Tanzu.

Unresolved directive in containers/vtwn\_overview\_trident.adoc - include::.../../\_include/containers\_common\_intro\_sections.adoc[tags=trident\_overview]

### Deploy Trident operator using Helm

- First set the location of the user cluster's kubeconfig file as an environment variable so that you don't have to reference it, because Trident has no option to pass this file.

```
[netapp-user@rhel7]$ export KUBECONFIG=~/.tanzu-install/auth/kubeconfig
```

2. Add the NetApp Astra Trident helm repository.

```
[netapp-user@rhel7]$ helm repo add netapp-trident  
https://netapp.github.io/trident-helm-chart  
"netapp-trident" has been added to your repositories
```

3. Update the helm repositories.

```
[netapp-user@rhel7]$ helm repo update  
Hang tight while we grab the latest from your chart repositories...  
...Successfully got an update from the "netapp-trident" chart repository  
...Successfully got an update from the "bitnami" chart repository  
Update Complete. ☐Happy Helming!☐
```

4. Create a new namespace for the installation of Trident.

```
[netapp-user@rhel7]$ kubectl create ns trident
```

5. Create a secret with DockerHub credentials to download the Astra Trident images.

```
[netapp-user@rhel7]$ kubectl create secret docker-registry docker-  
registry-cred --docker-server=docker.io --docker-username=netapp  
-solutions-tme --docker-password=xxxxxxx -n trident
```

6. For user or workload clusters managed by TKGS (vSphere with Tanzu) or TKG with management cluster deployments, complete the following procedure to install Astra Trident:

- a. Ensure that the logged in user has the permissions to create service accounts in trident namespace and that the service accounts in trident namespace have the permissions to create pods.
- b. Run the below helm command to install Trident operator in the namespace created.

```
[netapp-user@rhel7]$ helm install trident netapp-trident/trident-  
operator -n trident --set imagePullSecrets[0]=docker-registry-cred
```

7. For a user or workload cluster managed by TKGI deployments, run the following helm command to install Trident operator in the namespace created.

```
[netapp-user@rhel7]$ helm install trident netapp-trident/trident-operator -n trident --set imagePullSecrets[0]=docker-registry-cred,kubeletDir="/var/vcap/data/kubelet"
```

## 8. Verify that the Trident pods are up and running.

```
NAME                                READY   STATUS    RESTARTS
AGE
trident-csi-6vv62                   2/2     Running   0
14m
trident-csi-cfd844bcc-sqhcg        6/6     Running   0
12m
trident-csi-dfcmz                   2/2     Running   0
14m
trident-csi-pb2n7                   2/2     Running   0
14m
trident-csi-qsw6z                   2/2     Running   0
14m
trident-operator-67c94c4768-xw978  1/1     Running   0
14m
```

```
[netapp-user@rhel7]$ ./tridentctl -n trident version
+-----+-----+
| SERVER VERSION | CLIENT VERSION |
+-----+-----+
| 22.04.0        | 22.04.0        |
+-----+-----+
```

## Create storage-system backends

After completing the Astra Trident Operator install, you must configure the backend for the specific NetApp storage platform you are using. Follow the links below to continue the setup and configuration of Astra Trident.

- [NetApp ONTAP NFS](#)
- [NetApp ONTAP iSCSI](#)

## NetApp ONTAP NFS configuration

To enable Trident integration with the NetApp ONTAP storage system via NFS, you must create a backend that enables communication with the storage system. We configure a basic backend in this solution, but if you are looking for more customized options, visit the documentation [here](#).

## Create an SVM in ONTAP

1. Log into ONTAP System Manager, navigate to Storage > Storage VMs, and click Add.
2. Enter a name for the SVM, enable the NFS protocol, check the Allow NFS Client Access checkbox, and add the subnets that your worker nodes are on in the export policy rules for allowing the volumes to be mounted as PVs in your workload clusters.

### Add Storage VM ×

STORAGE VM NAME

trident\_svm

### Access Protocol

SMB/CIFS, NFS, S3

iSCSI

Enable SMB/CIFS

Enable NFS

Allow NFS client access

Add at least one rule to allow NFS clients to access volumes in this storage VM. [?](#)

EXPORT POLICY

Default

RULES

Rule Index	Clients	Access Protocols	Read-Only Rule	Read/Wr
	0.0.0.0/0	Any	Any	Any



If you are using NAT'ed deployment of user clusters or workload clusters with NSX-T, you need to add the Egress subnet (in the case of TKGS0 or the Floating IP subnet (in the case of TKGI) to the export policy rules.

3. Provide the details for data LIFs and the details for SVM administration account, and then click Save.

## NETWORK INTERFACE

Use multiple network interfaces when client traffic is high.

### K8s-Ontap-01

IP ADDRESS

172.21.252.180

SUBNET MASK

24

GATEWAY

172.21.252.1 X

BROADCAST DOMAIN

Default v

## Storage VM Administration

Manage administrator account

USER NAME

vsadmin

PASSWORD

.....

CONFIRM PASSWORD

.....

Add a network interface for storage VM management.

4. Assign the aggregates to an SVM. Navigate to Storage > Storage VMs, click the ellipsis next to the newly created SVM and then click Edit. Check the Limit Volume Creation to Preferred Local Tiers checkbox and attach the required aggregates to it.

# Edit Storage VM



STORAGE VM NAME

trident\_svm

DEFAULT LANGUAGE

c.utf\_8



DELETED VOLUME RETENTION PERIOD 


12

HOURS

## Resource Allocation

Limit volume creation to preferred local tiers

LOCAL TIERS

K8s\_Ontap\_01\_SSD\_1 

Cancel

Save

5. In case of NAT'ed deployments of user or workload clusters on which Trident is to be installed, the storage mount request might arrive from a non-standard port due to SNAT. By default, ONTAP only allows the volume mount requests when originated from root port. Thus, log into ONTAP CLI and modify the setting to

allow mount requests from non-standard ports.

```
ontap-01> vserver nfs modify -vserver tanzu_svm -mount-rootonly disabled
```

### Create backends and StorageClasses

1. For NetApp ONTAP systems serving NFS, create a backend config file on the jumphost with the backendName, managementLIF, dataLIF, svm, username, password, and other details.

```
{
  "version": 1,
  "storageDriverName": "ontap-nas",
  "backendName": "ontap-nas+10.61.181.221",
  "managementLIF": "172.21.224.201",
  "dataLIF": "10.61.181.221",
  "svm": "trident_svm",
  "username": "admin",
  "password": "password"
}
```



It is a best practice to define the custom backendName value as a combination of the storageDriverName and the dataLIF that is serving NFS for easy identification.

2. Create the Trident backend by running the following command.

```
[netapp-user@rhel7]$ ./tridentctl -n trident create backend -f backend-ontap-nas.json
+-----+-----+
+-----+-----+-----+
|          NAME          | STORAGE DRIVER |                UUID                |
| STATE | VOLUMES | |
+-----+-----+-----+
+-----+-----+-----+
| ontap-nas+10.61.181.221 | ontap-nas      | be7a619d-c81d-445c-b80c-5c87a73c5b1e |
| online |         0 |
+-----+-----+-----+
+-----+-----+-----+
```

3. With the backend created, you must next create a storage class. The following sample storage class definition highlights the required and basic fields. The parameter backendType should reflect the storage driver from the newly created Trident backend.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-nfs
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
```

4. Create the storage class by running the kubectl command.

```
[netapp-user@rhel7 trident-installer]$ kubectl create -f storage-class-nfs.yaml
storageclass.storage.k8s.io/ontap-nfs created
```

5. With the storage class created, you must then create the first persistent volume claim (PVC). A sample PVC definition is given below. Make sure that the `storageClassName` field matches the name of the storage class just created. The PVC definition can be further customized as required depending upon the workload to be provisioned.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: basic
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: ontap-nfs
```

6. Create the PVC by issuing the kubectl command. Creation can take some time depending on the size of the backing volume being created, so you can watch the process as it completes.

```
[netapp-user@rhel7 trident-installer]$ kubectl create -f pvc-basic.yaml
persistentvolumeclaim/basic created
```

```
[netapp-user@rhel7 trident-installer]$ kubectl get pvc
NAME      STATUS    VOLUME                                     CAPACITY
ACCESS MODES  STORAGECLASS  AGE
basic      Bound      pvc-b4370d37-0fa4-4c17-bd86-94f96c94b42d  1Gi
RWO                ontap-nfs      7s
```



## NetApp ONTAP iSCSI configuration

To integrate NetApp ONTAP storage system with VMware Tanzu Kubernetes clusters for persistent volumes via iSCSI, the first step is to prepare the nodes by logging into each node and configuring the iSCSI utilities or packages to mount iSCSI volumes. To do so, follow the procedure laid out in this [link](#).



NetApp does not recommend this procedure for NAT'ed deployments of VMware Tanzu Kubernetes clusters.



TKGI uses Bosh VMs as nodes for Tanzu Kubernetes clusters that run immutable configuration images, and any manual changes of iSCSI packages on Bosh VMs do not remain persistent across reboots. Therefore, NetApp recommends using NFS volumes for persistent storage for Tanzu Kubernetes clusters deployed and operated by TKGI.

After the cluster nodes are prepared for iSCSI volumes, you must create a backend that enables communication with the storage system. We configured a basic backend in this solution, but, if you are looking for more customized options, visit the documentation [here](#).

### Create an SVM in ONTAP

To create an SVM in ONTAP, complete the following steps:

1. Log into ONTAP System Manager, navigate to Storage > Storage VMs, and click Add.
2. Enter a name for the SVM, enable the iSCSI protocol, and then provide details for the data LIFs.

# Add Storage VM



STORAGE VM NAME

## Access Protocol

SMB/CIFS, NFS, S3  iSCSI

Enable iSCSI

NETWORK INTERFACE

### K8s-Ontap-01

IP ADDRESS	SUBNET MASK	GATEWAY	BROADCAST DOMAIN
<input type="text" value="10.61.181.231"/>	<input type="text" value="24"/>	<input type="text" value="10.61.181.1 X"/>	<input type="text" value="Defa..."/>

Use the same subnet mask, gateway, and broadcast domain for all of the following interfaces

IP ADDRESS	SUBNET MASK	GATEWAY	BROADCAST DOMAIN
<input type="text" value="10.61.181.232"/>	<input type="text" value="24"/>	<input type="text" value="10.61.181.1 X"/>	<input type="text" value="Defa..."/>

3. Enter the details for the SVM administration account, and then click Save.

---

## Storage VM Administration

Manage administrator account

USER NAME

vsadmin

PASSWORD

.....

CONFIRM PASSWORD

.....

Add a network interface for storage VM management.

**Save**

Cancel

4. To assign the aggregates to the SVM, navigate to Storage > Storage VMs, click the ellipsis next to the newly created SVM, and then click Edit. Check the Limit Volume Creation to Preferred Local Tiers checkbox, and attach the required aggregates to it.

# Edit Storage VM



STORAGE VM NAME

trident\_svm\_iscsi

DEFAULT LANGUAGE

c.utf\_8



DELETED VOLUME RETENTION PERIOD 

12

HOURS

---

## Resource Allocation

Limit volume creation to preferred local tiers

LOCAL TIERS

K8s\_Ontap\_01\_SSD\_1 

Cancel

Save

### Create backends and StorageClasses

1. For NetApp ONTAP systems serving NFS, create a backend config file on the jumphost with the backendName, managementLIF, dataLIF, svm, username, password, and other details.

```
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "backendName": "ontap-san+10.61.181.231",
  "managementLIF": "172.21.224.201",
  "dataLIF": "10.61.181.231",
  "svm": "trident_svm_iscsi",
  "username": "admin",
  "password": "password"
}
```

2. Create the Trident backend by running the following command.

```
[netapp-user@rhel7 trident-installer]$ ./tridentctl -n trident create
backend -f backend-ontap-san.json
+-----+-----+
+-----+-----+-----+-----+
|          NAME          | STORAGE DRIVER |                               UUID
| STATE | VOLUMES |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
| ontap-san+10.61.181.231 | ontap-san      | 6788533c-7fea-4a35-b797-
fb9bb3322b91 | online |          0 |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
```

3. After you create a backend, you must next create a storage class. The following sample storage class definition highlights the required and basic fields. The parameter `backendType` should reflect the storage driver from the newly created Trident backend. Also note the name-field value, which must be referenced in a later step.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-iscsi
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
```



There is an optional field called `fsType` that is defined in this file. In iSCSI backends, this value can be set to a specific Linux filesystem type (XFS, ext4, and so on) or can be deleted to allow Tanzu Kubernetes clusters to decide what filesystem to use.

4. Create the storage class by running the kubectl command.

```
[netapp-user@rhel7 trident-installer]$ kubectl create -f storage-class-iscsi.yaml
storageclass.storage.k8s.io/ontap-iscsi created
```

5. With the storage class created, you must then create the first persistent volume claim (PVC). A sample PVC definition is given below. Make sure that the `storageClassName` field matches the name of the storage class just created. The PVC definition can be further customized as required depending upon the workload to be provisioned.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: basic
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: ontap-iscsi
```

6. Create the PVC by issuing the kubectl command. Creation can take some time depending on the size of the backing volume being created, so you can watch the process as it completes.

```
[netapp-user@rhel7 trident-installer]$ kubectl create -f pvc-basic.yaml
persistentvolumeclaim/basic created
```

```
[netapp-user@rhel7 trident-installer]$ kubectl get pvc
NAME      STATUS    VOLUME                                     CAPACITY
ACCESS MODES  STORAGECLASS  AGE
basic      Bound      pvc-7ceac1ba-0189-43c7-8f98-094719f7956c  1Gi
RWO                ontap-iscsi      3s
```

## Videos and demos: VMware Tanzu with NetApp

The following videos demonstrate some of the capabilities described in this document:

[Use Astra Trident to Provision Persistent Storage in VMware Tanzu - VMware Tanzu with NetApp](#)

[Use Astra Control Center to Clone Applications in VMWare Tanzu - VMware Tanzu with NetApp](#)



These demos were recorded as a tech preview using version 1.3.1 of TKG and version 21.12 of Astra Control Center. Please see the Support Matrix for official supported versions.

## Additional Information: VMware Tanzu with NetApp

To learn more about the information described in this document, review the following websites:

- NetApp Documentation

<https://docs.netapp.com/>

- Astra Trident Documentation

<https://docs.netapp.com/us-en/trident/>

- NetApp Astra Control Center Documentation

<https://docs.netapp.com/us-en/astra-control-center/>

- Ansible Documentation

<https://docs.ansible.com/>

- VMware Tanzu Documentation

<https://docs.vmware.com/en/VMware-Tanzu/index.html>

- VMware Tanzu Kubernetes Grid Documentation

<https://docs.vmware.com/en/VMware-Tanzu-Kubernetes-Grid/1.5/vmware-tanzu-kubernetes-grid-15/GUID-index.html>

- VMware Tanzu Kubernetes Grid Service Documentation

<https://docs.vmware.com/en/VMware-vSphere/7.0/vmware-vsphere-with-tanzu/GUID-152BE7D2-E227-4DAA-B527-557B564D9718.html>

- VMware Tanzu Kubernetes Grid Integrated Edition Documentation

<https://docs.vmware.com/en/VMware-Tanzu-Kubernetes-Grid-Integrated-Edition/index.html>

## Copyright information

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

## Trademark information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.