VERITAS

NetBackup™ Kubernetes Installation and Configuration Guide

Release 10.5

Ports and Requirements

Ports required for communication

Port number	From	To	Used for
(6)(8)443	Primary server	Kubernetes cluster	HTTPS communications
443	Media servers	Kubernetes cluster	HTTPS communications
1556 outbound	Kubernetes cluster	Primary server	For certificate deployment and PBX communication with the Primary server
1556 outbound	Kubernetes cluster	Media servers	For certificate deployment
13724 bi-directional	Kubernetes cluster	Primary server	VNETD for data movement
13724 bi-directional	Kubernetes cluster	Media servers	VNETD for data movement

Deployment - Prerequisites for NetBackup Kubernetes Operators

- Customer needs to install **helm chart** on Kubernetes cluster access host to deploy operator.
- Check the Software Compatibility List (SCL) and Hardware Compatibility List (HCL) guide for supported configurations.
- Kubernetes operator requires 10Gi of storage, 100m CPU and 500Mi memory up to 150m max CPU and 600mi memory. All configuration is done during deployment.
- To install NetBackup Kubernetes Operator (NBKOPs) you require administrative privileges.
- Customers either need access to a local repository to place the NetBackup Kubernetes Operator (NBKOPs) and Data mover packages for deployment or customers need internet access to the Veritas Customer Repository and use whatever tools customer generally use to copy mentioned packages into their local repository.
- A namespace must be configured to deploy the NetBackup Kubernetes Operator (NBKOPs), and Data mover packages.

Deployment - Prerequisites for NetBackup Kubernetes Operators

- Primary and Media servers must be created with FQDN, if they are created with short names. For more information, refer the section Prerequisites for backup from snapshot and restore from backup operations (Point 3 and 8) in the NetBackup™ Web UI Kubernetes Administrator's Guide.

- Namespaces with persistent storage must use CSI enabled storage with snapshot support, for more details refer to the Hardware Compatibility List (HCL). NBKOPs supports only snapshot API version 'v1' for backup operations.

- Kubernetes supports DTE mode setting, Customer can configure DTE mode setting that is set on the datamover via backupserver specific configmap. Data-in-transit encryption of backup images is carried out based on the global DTE mode and the client DTE mode.

NetBackup upgrade

Important notes:

- All components (NBU Primary, Media, Kubernetes operators, and Data mover) must be same version.
- Existing policies continue to take backups but must be restored manually until the Kubernetes operator is updated.

Note: This is applicable to the NetBackup version 9.1 to 10.x upgrade.

Steps to upgrade

1. Download the tar package from Veritas Support website: https://www.veritas.com/content/support
2. Extract the package to the home directory. The **netbackupkops-helm-chart** folder must be in the home directory.
3. To list all cluster contexts, run the command: **kubectl config get-contexts**
4. To switch to the cluster where you want to deploy the operator service, run the command:

   kubectl config use-context <cluster-context-name>
5. To change the current directory to your home directory, run the command: **cd ~**
6. NetBackup supports any Container Image Repositories complied to OCI standards. you can use any tools to push the operators and data mover images. If you use a private docker registry, follow the instructions in this step to create a secret **nb-docker-cred** in NetBackup namespace. Otherwise, skip to the next step.

Steps to upgrade

- To load the image to the docker cache and push the image to the docker image repository, run the commands:
  - Load the tar file for Netbackup Kubernetes Operator.

    <docker load -i <nameof the tar file> ./>
  - Tag the loaded docker image as per requirement.

    docker tag <imagename:tagof the loadedimage> <repo-name/image-name:tag-name>
  - Push the image to a repository from where Kubernetes can fetch the image at the time of NetBackup Kubernetes Operator deployment.

    docker push <repo-name/image-name:tag-name>

Note: In the example, Docker is used for reference. You can use any other CLI tool that provides equivalent functionality.

Steps to upgrade

7. Edit the **netbackupkops-helm-chart/values.yaml** in a text editor:

   Replace the image value in the manager section with your image name and tag in the format **reponame/image-name:tag-name**. Aslo, replace the datamover image in the **netbackup_config_pod** section with datamover image name and tag.

8. To upgrade the NetBackup Kubernetes operator, run the command:

   helm upgrade <plugin-name> <chart-path> -n <namespace>

   Example:

   helm upgrade veritas-netbackupkops ./netbackupkops-helm-chart -n netbackup

   Note: Upgrading the NetBackup Kubernetes operator will reset the Helm values to their defaults. You must ensure that you back up the old configmap and reapply any patches if the values have changed after the upgrade

Download the Veritas NetBackup Kubernetes Operator, Data mover tar packages and load images into local repository

Download and extract Veritas NetBackup Kubernetes packages

Get packages from Veritas Entitlement Management System (VEMS)

1. Go to the support.veritas.com, to log onto the **Veritas Entitlement Management System** (VEMS) and the, sign in and click **Licensing**.
2. Click **Entitlements** (within Veritas Entitlement Management System (VEMS) main menu).
3. Click **More Options** to expand filters.
4. Set the **Product Line** filter to NetBackup and click **Apply Filters**.
5. Look for entitlement's references, the version matches to the release note mentioned in the subject line.
6. To access your new software and license key(s) utilize the **Download Software** and **Generate License** buttons located in the **Actions** column.
7. Download two packages:
   - NetBackup Kubernetes operator package (**netbackupkops-10.4.1.tar.gz**).
   - NetBackup Data mover image (**veritasnetbackup-datamover-10.4.1.tar**).

Download and extract Veritas NetBackup Kubernetes packages

Package names and content

8. Extract the package to the home directory of a system that has access to the cluster where you can run **kubectl** commands. The **netbackupkops-helm-chart** is part of the NetBackup Kubernetes Operator package and is in the home directory.

9. To list all cluster contexts, run the following command: **kubectl config get-contexts**

10. To switch to the cluster where you want to deploy the operator service, run the following command:

    kubectl config use-context <cluster-context-name>

Add the NetBackup Kubernetes operator and Data mover image into your local registry

Follow the steps only if you have a private docker registry

1. Create a secret netbackupkops-docker-cred in the NetBackup operator namespace if container registry requires authentication. This secret is not needed if authentication is not configured.

   To log on to the private container registry, run the following command:

   docker login -u <username> <container-registry-url>

2. After sign in, the config.json file containing the authorization token is created or updated. To view the config.json file, run the following command:

   cat ~/.docker/config.json

   The output looks like below:

```
{
"auths": {
        "https://index.docker.io/v1/": {
        "auth": "c3R...zE2"
      } }
      }
```

Add the NetBackup Kubernetes operator and Data mover image into your local registry

3. To create a secret named netbackupkops-docker-cred in the NetBackup operator namespace, run the following command:

   kubectl create secret generic netbackupkops-docker-cred \ --from-file=.dockerconfigjson=.docker/config.json \ --type=kubernetes.io/dockerconfigjson -n <name of the namespace where the NetBackup operator will be deployed>

4. To check if the secret netbackupkops-docker-cred is created in the NetBackup operator namespace, run the following command:

   kubectl get secrets -n <name of the namespace where the NetBackup operator will be deployed>

Add the NetBackup Kubernetes operator and Data mover image into your local registry

Push the NetBackup Kubernetes Operator and Data mover images to your registry

1. To load the image to the docker cache and push the image to the docker image repository, run the following commands:
   - docker load -i <name of the tar file>
   - docker tag <image name:tag of the loaded image> <repo-name/image-name:tag-name>
   - docker push <repo-name/image-name:tag-name>

NetBackup Kubernetes Operator installation and configuration

NetBackup Kubernetes Operator installation and configuration

There are two ways to configure NetBackup Kubernetes Operator with NetBackup Primary server:

1. Using Automated configuration by providing required parameters in values.yaml file
2. Using manual configuration steps

Installation and configuration

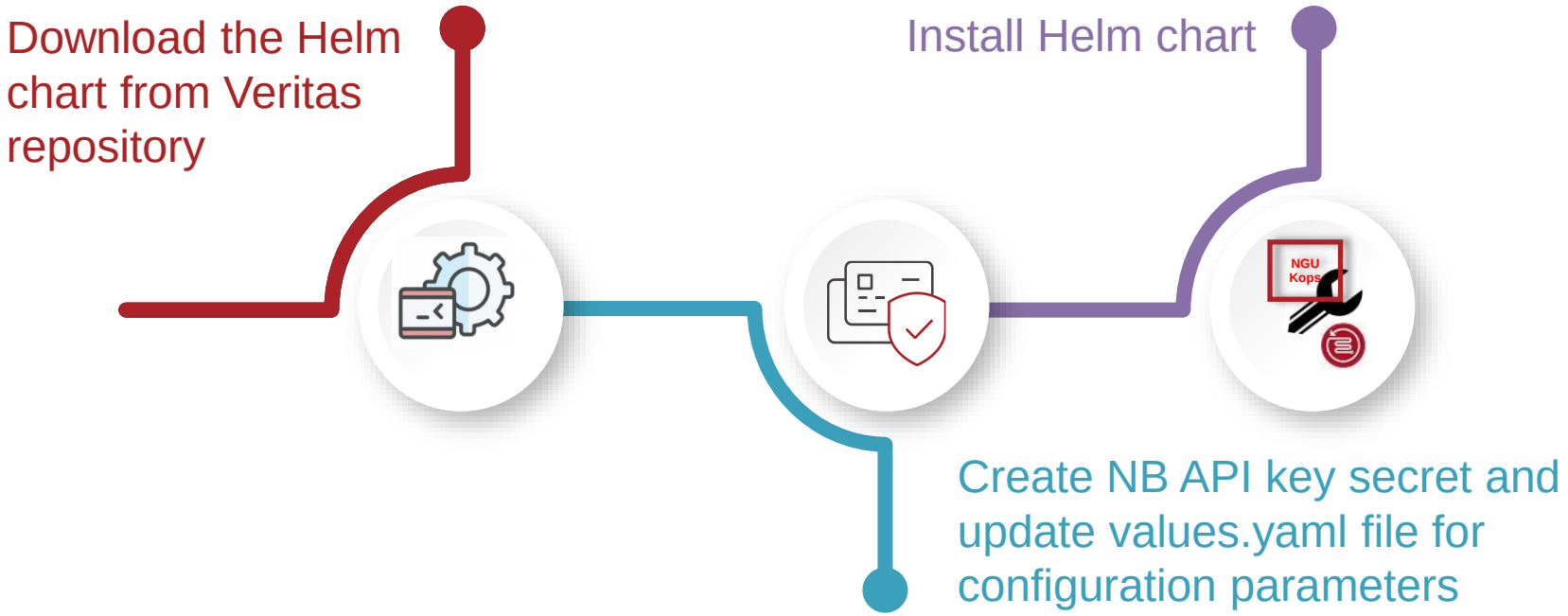
NetBackup Kubernetes operator deployment and configuration

1. Using Automated configuration by providing required parameters in values.yaml file
   - Before you run the helm install, user need to provide required input values in the **netbackupkops-helm-chart/values.yaml** file.
   - As part of configuration deployment, a pod gets created each time user runs helm install. config pod runs the script to configure Kubernetes workload in the NetBackup web UI.
   - This process includes following operations:
     - Prepares storage for backup and restore: Label volmesnapshotclass and storage classes for creating snapshots and PVCs.
     - Read service account token from NetBackup operator namespace.
     - Create NetBackup credentials for Kubernetes in NetBackup.
     - Add Kubernetes cluster to NetBackup.
     - Create NetBackup token and fetch sha256 fingerprint.
     - Create BackupServerCert for establishing secure communication for datamover pod.
     - Configure primary server specific configmap for datamover image.

2. Using manual configuration steps.

NetBackup Kubernetes Operator deployment and configuration using automated configuration

Deployment High-level steps using automated configuration



Automated configuration

Update netbackupkops-helm-chart/values.yaml configuration file

- Untar netbackupkops.tar.gz file (run command: ***tar –xvf netbackupkops.tar.gz***) and provide inputs required for **netbackupkops-helm-chart/values.yaml**

  netbackup_config_pod:

  - **nbprimaryserver :** <FQDN of NetBackup Primary Server>
  - **nbsha256fingerprint :** <Copy sha256 fingerprint from NetBackup Primary server web UI>

    (Go to NetBackup web UI → Security → Certificates → click Certificate Authority)
  - **k8sCluster :** <FQDN of Kubernetes cluster API server (run command : **kubectl cluster-info**)>
  - **k8sPort :** <Port on which Kubernetes API server is listening>
  - **datamoverimage :** <Container registry URL for pulling datamover image>

Automated configuration

Update netbackupkops-helm-chart/values.yaml configuration file

- **storageclassblock :** <Storage class used for provisioning block volumes (run command: **kubectl get storageclasses**)>
- **storageclassfilesystem :** <Storage class used for provisioning filesystem volumes (run command: kubectl get **storageclasses**)>
- **volumesnapshotclassblock :** <Volume snapshot class for creating block volume snapshots (run command : kubectl get volumesnapshotclass)>
- **volumesnapshotclassfilesystem :** <Volume snapshot class for creating filesystem volume snapshots (run command : kubectl get volumesnapshotclass)>
- **storageMap:** It is a dictionary of key, value fields where key is storage class and its value is a tuple consisting of (snapshotClass, storageClassForBackupDataMovement, storageClassForRestoreFromBackup). For more details, refer to the **netbackupkops-helm-chart/values.yaml**

Automated configuration

Update netbackupkops-helm-chart/values.yaml configuration file

- **acceleratorTracklogPvcStorageClass:** The storage class used for provisioning tracklog for Accelerator backups (Run command: kubectl get storageclasses) defaults to None, indicating non-accelerator backups will be performed. To enable accelerator backups, set this value to a valid storage class**.**
- **MinSizeForAcceleratorTracklogPvc**: The Minimum size in Mi/M for creation the accelerator tracklog PVC during installation or upgrade is set to 50Mi by default.

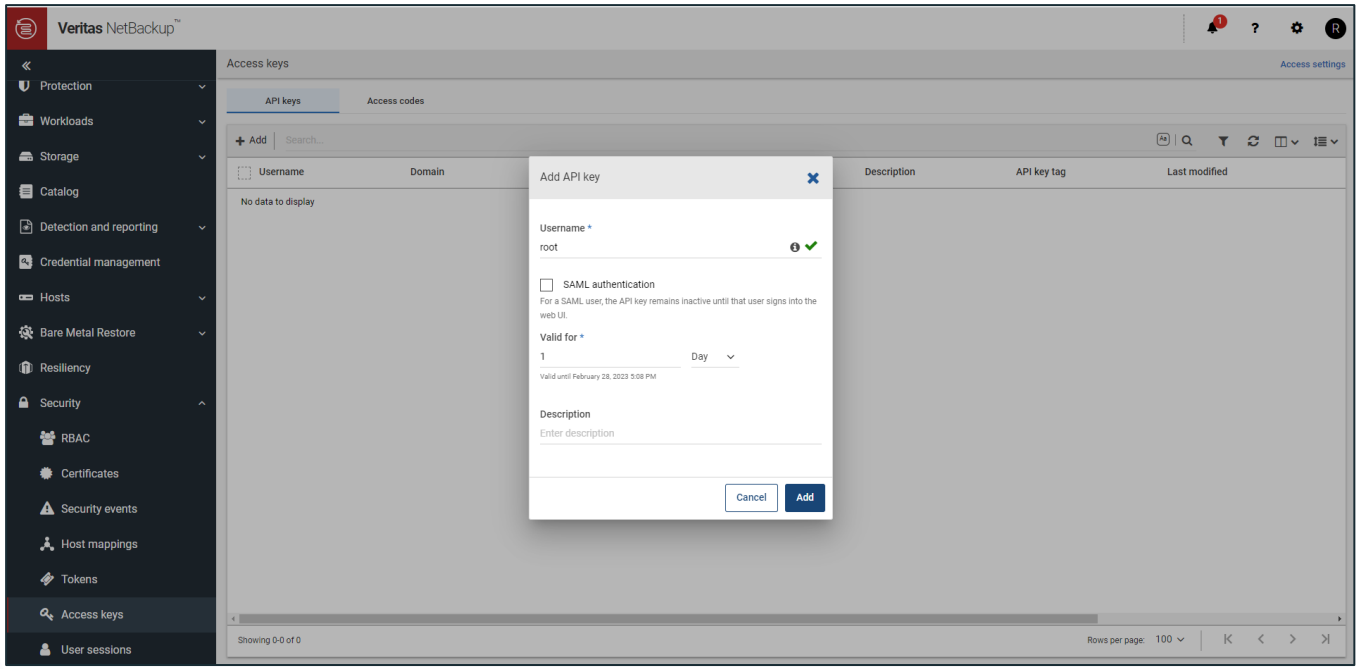
Note: Automated configuration is currently supported **only for NBCA mode.** For more details about the volume snapshot class and storage class names, refer to the *Label Storage for Backup and Restore* section in the *NetBackup Kubernetes Administrator's guide*.

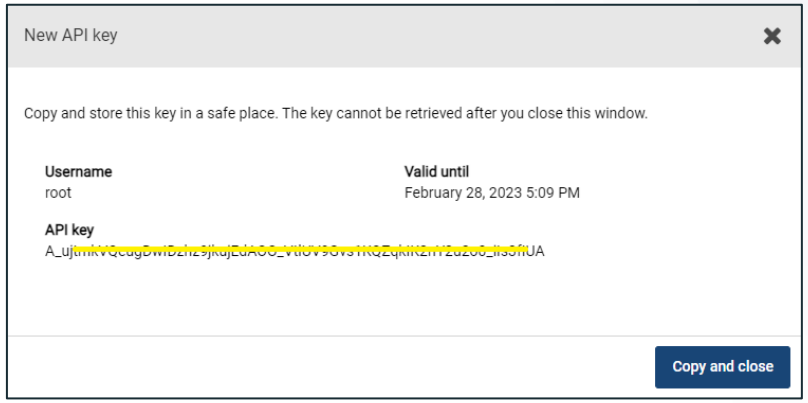
Automated configuration: API key creation

Create Netbackup API key for nb-config-deploy-secret

1. If an API key already exists, same could be used in next step without creating a new API key. The details are hidden for already created apikey and must be collected from NetBackup admin.

To create a new API key :
Go to the NetBackup web UI → Security → Access keys and click Add. Add username and select validity of one day to avoid misuse of API key. You must delete the Secret after the configuration is done.





Automated configuration : API Key creation

Create secret for NetBackup configuration pod

2. Create a secret file as shown below containing NetBackup API key

   Sample: **nb-config-deploy-secret.yaml**

3. Enter the API key value here which was extracted in the previous step.

```
apiVersion: v1
kind: Secret
metadata:
  name: <nbkops-namespace>-nb-config-deploy-secret
  namespace: <nbkops-namespace>
type: Opaque
stringData:
  apikey: A6XXX_KpZFNi8JSqbwWdBrd7YncicmHdrLIiatkUdoJtjWu5_-wyRNeJlfXXXX7B
```

4. Once the file is ready, create a namespace <kops-namespace> and secret:
   kubectl create namespace <kops-namespace>
   kubectl apply -f ***nb-config-deploy-secret.yaml***

Automated configuration netbackupkops-helm-chart/values.yaml file

sample values.yaml

```yaml
netbackupkops:
  containers:
    manager:
      image: reg.domain.com/10.4.1/nbk8splugin:netbackupkops_10.4.1
      resources:
        limits:
          cpu: 150m
          memory: 600Mi
        requests:
          cpu: 100m
          memory: 500Mi
  kopsPvcStorageClass: ocs-storagecluster-ceph-rbd
  kopsPvcSize: 10Gi
  pvMountPath: /usr/openv
  imagePullSecrets: []
  # Example to add imagePullSecrets
  # imagePullSecrets:
  # - name: netbackupkops-docker-cred
  fipsMode: DISABLE
  acceleratorTracklogPvcStorageClass: None
  minSizeForAcceleratorTracklogPvc: 50Mi
  hostAliases: []
  # Example to add hostAliases
  # hostAliases:
  #   - hostnames:
  #     - example1.domain.tld
  #     - example2.domain.tld
  #     ip: 1.2.3.4
  #   - hostnames:
  #     - example3.domain.tld
  #     ip: 5.6.7.8

nbsetup:
  replicas: 1
  containers:
    netbackup_config_pod:
      nbprimaryserver: primarybackupserver.sample.domain.com
      nbsha256fingerprint: 8B:12:XX:D3:XX:AB:XX:FF:XX:3F:XX:AF:XX:94:XX:A4:XX:64:XX:52:XX:25:XX:05:XX:32:XX:10:XX:50:XX:8A
      k8sCluster: testcluster.sample.domain.com
      k8sPort: 6443
      datamoverimage: reg.domain.com/10.4.1/datamover:10.4.1
      logLevel: INFO
      storageclassblock: ocs-storagecluster-ceph-rbd
      volumesnapshotclassblock: ocs-storagecluster-rbdplugin-snapclass
      storageclassfilesystem: ocs-storagecluster-cephfs
      volumesnapshotclassfilesystem: ocs-storagecluster-cephfsplugin-snapclass
  storageMap:
    ocs-storagecluster-cephfs:
      storageClassForBackupDataMovement: ocs-storagecluster-cephfs
      storageClassForRestoreFromBackup: ocs-storagecluster-cephfs
      snapshotClass: ocs-storagecluster-cephfsplugin-snapclass
    ocs-storagecluster-ceph-rbd:
      storageClassForBackupDataMovement: ocs-storagecluster-ceph-rbd
      storageClassForRestoreFromBackup: ocs-storagecluster-ceph-rbd
      snapshotClass: ocs-storagecluster-rbdplugin-snapclass
```

Automated configuration : Custom CA cert config (Optional)

Create a secret for external cert configuration

External CA certificate configuration for Kubernetes clusters

Note: This configuration step is necessary if you have custom CA certificates configured on your cluster's API server for external access. This step can be ignored if there is no custom CA setup

- If you have the custom CA certificates available, then simply enter them directly in the configuration yaml file mentioned in the next slide.
- If you do not have the certificates available, run the command shown to extract the certificate on the NetBackup primary host.
- You can also use the **openssl** command tool to perform the same step on the Linux host.

```
<NBU_Install_Path>/bin/goodies/vxsslcmd s_client -showcerts -connect <cluster-fqdn>:<port-no> 2>/dev/null </dev/null
|  sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p'
```

- Enter the certificate value which will be generated from the output of this command into the template file in the next step.

Automated configuration: Custom CA cert config (Optional)

Create a secret for external cert configuration

External CA certificate configuration for Kubernetes clusters using custom CA certs

- Prepare a yaml file based on the format shown.
- **Note:** Ensure the proper indentations as shown in the following template.
- Enter the value which was extracted in the previous step into the **k8scacert** field and ensure the indentation is properly followed for the entire value

```
apiVersion: v1
kind: Secret
metadata:
  name: <kops-namespace>-nb-config-deploy-secret
  namespace: <kops-namespace>
type: Opaque
stringData:
  apikey: <Netbackup API Key>
  k8scacert: |
    -----BEGIN CERTIFICATE-----
    MIIDDDCCAfSgAwIBAgIBATANBgkqhkiG9w0BAQsFADAmMSQwIgYDVQQDDBtpbmdy
    ZXNzLW9wZXJhdG9yQDE2ODc1MzY4NjgwHhcNMjMwNjIzMTYxNDI3WhcNMjUwNjIy
    XtXqbaBGrXIuCCo90mxv4g==
    -----END CERTIFICATE-----
```

Automated configuration for Rancher managed clusters

Extract Rancher cacert and token

Rancher managed clusters use custom cacert provided by Rancher manager to communicate with NetBackup.

NetBackup also requires token from the managed cluster for authentication.

- **Rancher managed cacert**:
    1. Navigate to the **Rancher Management Server UI** > Open the left side panel **Global Settings** > Under CA Certs, click the Show CA Certs button.
    2. Extract the complete CA cert value in a temporary file.
- **Cluster Token**:
    1. Open the **Rancher Management Server UI** > Open the left side panel > Under the **Explore Cluster** Section > Navigate to the cluster you want to protect > Click the Download KubeConfig icon on the top right corner.
    2. Token field is present inside the kubeconfig file. Extract the token value without the double quotes " " from this downloaded kubeconfig file.

Automated configuration for Rancher managed clusters

Create a secret for external cert and token configuration

Follow the step to create a secret for Rancher managed cluster

1. Create a yaml file nb-config-deploy-secret.yaml with the following format shown.
2. Enter the values for k8stoken and k8scacert which were extracted earlier under stringData.
3. Once the **k8stoken** and **k8scacert** values have been entered, move to the API Key creation step on next slide to get the value for **apikey**

Sample file : nb-config-deploy-secret.yaml

```
apiVersion: v1
kind: Secret
metadata:
  name: <kops-namespace>-nb-config-deploy-secret
  namespace: <kops-namespace>
type: Opaque
stringData:
# All the 3 fields are mandatory here to add a Rancher managed RKE2 cluster in Netbackup
  apikey: A_YoUkgYQwkPLUkmyj9Q6A1-6RX8RNY-PtYX0SukbqCwIK_osPz8qVm9zCL9phje
  k8stoken: kubeconfig-user-mvvgcm8sq8:nrsvcnx8hj46t24r2tjrxd2kn8tzo2bg4kj8waxpw36k8ktrchp826
  k8scacert: |
    -----BEGIN CERTIFICATE-----
    MIIDDDCCAfSgAwIBAgIBATANBgkqhkiG9w0BAQsFADAmMSQwIgYDVQQDDBtpbmdy
    ZXNzLW9wZXJhdG9yQDE2ODc1MzY4NjgwHhcNMjMwNjIzMTYxNDI3WhcNMjUwNjIy
    XtXqbaBGrXIuCCo90mxv4g==
    -----END CERTIFICATE-----
```

Automated configuration: Deploy the NetBackup Kubernetes Operator

Install operator via the Helm Chart

1. Ensure the **<kops-namespace>-nb-config-deploy-secret** has been created before running the helm install on the next step.
2. Run the following command to install the NetBackup Kubernetes Operator:

   helm install <user defined name of the deployment> ./netbackupkops-helm-chart -n <kops-namespace>

   Example: ***helm install veritas-netbackupkops ./netbackupkops-helm-chart -n netbackup***
3. To check the status of the deployments, run the command: *kubectl get pods -n <kops-namespace>*
   Example: ***kubectl get pods -n netbackup***
4. To verify that Kubernetes cluster is added to NetBackup, Go to the NetBackup web UI → Workloads → Kubernetes → Kubernetes clusters.

   Kubernetes cluster should be listed on this page. If there is an issue, check the troubleshooting steps on next page.

Automated configuration

Troubleshooting the NetBackup configuration pod

Troubleshooting the configuration pod:

1. To check configuration pod logs using the following commands:
   kubectl get pods -n <kops-namespace>
   kubectl logs <netbackup-config-pod-name> -n <kops-namespace> > config-deploy.log

2. If you see any failures in the deployment due to incorrect input values, set the replica count to zero for deployment <kops-namespace>-netbackup-config-deploy to remove the deploy pod.
   kubectl scale deployment <kops-namespace>-netbackup-config-deploy -n <kops-namespace> --replicas=0

3. Correct the input values in deployment config.
   kubectl edit deployment <kops-namespace>-netbackup-config-deploy -n <kops-namespace>

4. Again, set the replica count to 1 to restart configuration. Use the following command to set the replica count:
   kubectl scale deployment <kops-namespace>-netbackup-config-deploy -n <kops-namespace> --replicas=1

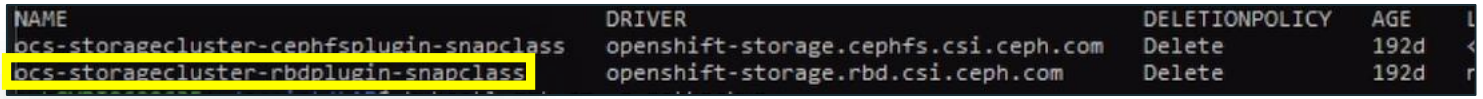
NetBackup Kubernetes Operator deployment and configuration using manual steps

Label storage for backup and restore

Prepare storage

Check for the valid volume snapshot class available in your environment

1. To see what volume snapshot classes are available in the environment, run the following command:
   - ***kubectl get volumesnapshotclass***
   - Following is the output example, you can see:



   - The parameter you will use is the value in the name field from this command that is associated with the appropriate CSI driver.
2. User must label a valid volume snapshot classes on the block and file system volume snapshot classes to create a block and file system volume snapshots for NetBackup usage.

Prepare storage

Label a valid volume snapshot class for NetBackup usage

3. Add the following label on the block and file system volume snapshot classes to create block and file system volume snapshots:

   netbackup.veritas.com/default-csi-volume-snapshot-class=true

4. Run the following commands:

   - *kubectl label volumesnapshotclass <block-vol-snap-class-name> netbackup.veritas.com/default-csi-volume-snapshot-class=true*
   - *kubectl label volumesnapshotclass <filesystem-vol-snap-class-name> netbackup.veritas.com/default-csi-volume-snapshot-class=true*

5. If the NetBackup labeled VolumeSnapshotClass class is not found, then snapshot of a namespace consisting of persistent volume fails with an error message: Failed to create snapshot of the Kubernetes namespace.

Prepare storage

Check for the storage available in your environment

6. To see storage classes available in the environment, run the following command:

   kubectl get sc

   - Following is the output example, you can see:

```
[root@dl380g10-066-v38 ~]# kubectl get sc
NAME                                 PROVISIONER                                RECLAIMPOLICY   VOLUMEBINDINGMODE      ALLOWVOLUMEEXPANSION   AGE
localblock                           kubernetes.io/no-provisioner               Delete          WaitForFirstConsumer   false                  53d
ocs-storagecluster-ceph-rbd (default)   openshift-storage.rbd.csi.ceph.com      Delete          Immediate              true                   53d
ocs-storagecluster-ceph-rgw          openshift-storage.ceph.rook.io/bucket      Delete          Immediate              false                  53d
ocs-storagecluster-cephfs            openshift-storage.cephfs.csi.ceph.com      Delete          Immediate              true                   53d
openshift-storage.noobaa.io          openshift-storage.noobaa.io/obc            Delete          Immediate              false                  53d
thin                                 kubernetes.io/vsphere-volume               Delete          Immediate              false                  55d
thin-csi                             csi.vsphere.vmware.com                     Delete          WaitForFirstConsumer   true                   55d
[root@dl380g10-066-v38 ~]#
```

7. Look for the storage that has CSI drivers listed under provisioner which consists of CSI in the name.
8. You must label each of the CSI supported storage classes with the labels in this section.

Prepare storage

Validate the storage you use for deploying namespaces

9. The command ‘kubectl get sc’ you ran earlier takes note of the default storage listed:

```
$ kubectl get sc
NAME                          PROVISIONER                             RECLAIMPOLICY   VOLUMEBINDINGMODE      ALLOWVOLUMEEXPANSION   AGE
localvolume                   kubernetes.io/no-provisioner            Delete          WaitForFirstConsumer   false                  191d
ocs-storagecluster-ceph-rbd   openshift-storage.rbd.csi.ceph.com      Delete          Immediate              true                   191d
ocs-storagecluster-ceph-rgw   openshift-storage.ceph.rook.io/bucket   Delete          Immediate              false                  191d
ocs-storagecluster-cephfs     openshift-storage.cephfs.csi.ceph.com   Delete          Immediate              true                   191d
openshift-storage.noobaa.io   openshift-storage.noobaa.io/obc         Delete          Immediate              false                  191d
thin (default)                kubernetes.io/vsphere-volume            Delete          Immediate              false                  196d
```

10. If the default storage for the cluster is not associated with the CSI storage you labeled. Then, any namespaces created with default storage will not be able to protect.

11. Customers either must change the default storage to CSI storage which needs to protect, or explicitly point to the CSI storage when the namespaces are created.

Prepare storage

Label a valid storage class for NetBackup usage

12. Add the following labels on CSI supported storage class:
    - ***netbackup.veritas.com/default-csi-storage-class=true*** is used to label where storage class provisions volumes based on raw block.
    - **netbackup.veritas.com/default-csi-filesystem-storage-class=true** is used to label where storage class provisions volumes based on file system.
13. Run the following commands:
    - ***kubectl label sc <storage class> netbackup.veritas.com/default-csi-storage-class=true***
    - ***Kubectl label sc <storage class> netbackup.veritas.com/default-csi-filesystem-storage-class=true***
    - ***<Storage class> will be from the name section of the 'kubectl get sc' command for each CSI compliant storage you will need to protect.***
14. If NetBackup labeled storage class is not found, then backup from snapshot job for metadata image and restore jobs fails with an error message No eligible storage classes found.

Prepare storage

Validate that labels were applied

15. To verify the result, run the following commands:
    - *kubectl get sc --show-labels*

```
[root@dl380g10-066-v38 ~]# k get sc --show-labels
NAME                                PROVISIONER                             RECLAIMPOLICY   VOLUMEBINDINGMODE      ALLOWVOLUMEEXPANSION   AGE   LABELS
localblock                          kubernetes.io/no-provisioner            Delete          WaitForFirstConsumer   false                  53d   local.storage.openshift.io/owner-name=local-block,local.storage.openshift.io/owner-namespace=local-storage
ocs-storagecluster-ceph-rbd (default)   openshift-storage.rbd.csi.ceph.com      Delete          Immediate              true                   53d   netbackup.veritas.com/default-csi-filesystem-storage-class=true,netbackup.veritas.com/default-csi-storage-class=true
ocs-storagecluster-ceph-rgw         openshift-storage.ceph.rook.io/bucket   Delete          Immediate              false                  53d   <none>
ocs-storagecluster-cephfs           openshift-storage.cephfs.csi.ceph.com   Delete          Immediate              true                   53d   netbackup.veritas.com/default-csi-filesystem-storage-class=true
openshift-storage.noobaa.io         openshift-storage.noobaa.io/obc         Delete          Immediate              false                  53d   <none>
thin                                kubernetes.io/vsphere-volume            Delete          Immediate              false                  55d   <none>
thin-csi                            csi.vsphere.vmware.com                  Delete          WaitForFirstConsumer   true                   55d   <none>
[root@dl380g10-066-v38 ~]#
```

    - *kubectl get volumesnapshotclass --show-labels*

```
[root@dl380g10-066-v38 ~]# kubectl get volumesnapshotclass --show-labels
NAME                                      DRIVER                                  DELETIONPOLICY   AGE   LABELS
csi-vsphere-vsc                           csi.vsphere.vmware.com                  Delete           55d   <none>
ocs-storagecluster-cephfsplugin-snapclass   openshift-storage.cephfs.csi.ceph.com   Delete           53d   netbackup.veritas.com/default-csi-volume-snapshot-class=true
ocs-storagecluster-rbdplugin-snapclass      openshift-storage.rbd.csi.ceph.com      Delete           53d   netbackup.veritas.com/default-csi-volume-snapshot-class=true
[root@dl380g10-066-v38 ~]#
```

Prepare storage

Specify Storage Map in **netbackupkops-helm-chart/values.yaml** for BnR operations

storageMap: Refer values.yaml for more information.

Example:

```
storageMap:
ocs-storagecluster-ceph-rbd:
  storageClassForBackupDataMovement: ocs-storagecluster-ceph-rbd
  storageClassForRestoreFromBackup: ocs-storagecluster-ceph-rbd
  snapshotClass: ocs-storagecluster-rbdplugin-snapclass
```

- Set nbsetup **replica to 0** in **netbackupkops-helm-chart/values.yaml** to disable netbackup config pod

Deploy the NetBackup Kubernetes Operator

Install operator via the Helm Chart

1. Run the following command to install the NetBackup Kubernetes Operator:

   helm install <user defined name of the deployment> ./netbackupkops-helm-chart -n <kops-namespace>
   Example: ***helm install veritas-netbackupkops ./netbackupkops-helm-chart -n netbackup***

2. To check the status of the deployments, run the command: *kubectl get pods -n <kops-namespace>*
   Example: ***kubectl get pods -n netbackup***

Prerequisites for NetBackup backup from snapshot and restore from backup operations

Prerequisites for NetBackup BFS and RFS operations

- Ensure that the user label a valid storage class (Block and Filesystem) for NetBackup usage. (Refer: Prepare Storage section)
- Ensure that the user label a valid snapshotvolumeclass for NetBackup usage. (Refer: Prepare PV for Backup section)
- Each primary server which runs the backup from snapshot and restore from backup copy operations, needs to create a separate **ConfigMap with the primary server's name**. (Refer: Deploy the NetBackup Kubernetes Operator - Create configmap.yaml file for each Primary Server protecting the cluster)
- Ensure that the user deployed certificates on the Kubernetes operator. (Refer: Deploy certificates on the Kubernetes Operator section)

Create configmap file for each Primary server protecting the cluster

1. Create the following configmap yaml file.

```
apiVersion: v1
  data:
    datamover.hostaliases: |
      <Primary server IP Address>=<Primary server FQDN>
      <Media server IP address>=<Media server FQDN>
    datamover.properties: |
      image=<repo-name/image-name:tag-name>
    version: "1"
  kind: ConfigMap
  metadata:
    name: <Primary server FQDN>
    namespace: <Name of namespace where the NBUKops package will be deployed>
```

2. If you have more than one Media server that will access this cluster from that domain add each Media server to the data mover host alias list. You only need this parameter if the cluster can't automatically resolve the primary and media servers with name resolution.
3. repo-name/image-name:tag-name should be the parameters you set when you pushed the data mover images into the registry.

Deploy the NetBackup Kubernetes Operator

Create configmap yaml file for each Primary Server protecting the cluster

4. Save the configmap.yaml to the home directory from where the Kubernetes clusters are accessible.
5. If you have more than one primary server protecting the cluster create one configmap.yaml file for each primary server and give each file a unique name.
6. Run the following command to deploy each file where <*configmap.yaml*> is the name you gave to the file:

   kubectl create -f <configmap.yaml>

Data mover pod schedule mechanism support

Specify the following fields in the backupserver ConfigMap to schedule datamoverpods on the nodes

1. nodeSelector: nodeSelectoris the effortlessway to constrainpods to thenodeswith specificlabels.

   Example:

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: backupserver.sample.domain.com
  namespace: netbackup
data:
  datamover.hostaliases: |
    10.20.12.13=backupserver.sample.domain.com
    10.21.12.13=mediaserver.sample.domain.com
  datamover.properties: |
    image=reg.domain.com/datamover/image:latest
  datamover.nodeSelector: |
    kubernetes.io/hostname: test1-l94jm-worker-k49vj
    topology.rook.io/rack: rack1
  version: "1"
```

2. nodeName : nodeName is a more direct form of node selection than affinity or nodeSelector. It allows you to specify the node on which a pod should be scheduled, overriding the default scheduling mechanism.

Example:

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: backupserver.sample.domain.com
  namespace: netbackup
data:
  datamover.hostaliases: |
    10.20.12.13=backupserver.sample.domain.com
    10.21.12.13=mediaserver.sample.domain.com
  datamover.properties: |
    image=reg.domain.com/datamover/image:latest
  datamover.nodeName : test1-l94jm-worker-hbblk
  version: "1"
```

3. Taint & Toleration : Toleration allows you to schedule the pods with similar taints. Taint and toleration work together to ensure that the pods are scheduled onto appropriate nodes. If one or more taints are applied to a node. Then that node must not accept any pods which does not tolerate the taints.

Example:

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: backupserver.sample.domain.com
  namespace: netbackup
data:
  datamover.properties: |
    image=reg.domain.com/datamover/image:latest
  Datamover.tolerations: |
    - key: "dedicated"
      operator: "Equal"
      value: "experimental"
      effect: "NoSchedule"
  version: "1"
```

4. Affinity and Anti-affinity : Node affinity functions like the nodeSelector field but it is more expressive and allows you to specify soft rules. Inter-pod affinity/anti-affinity allows you to constrain Pods against labels on other Pods.

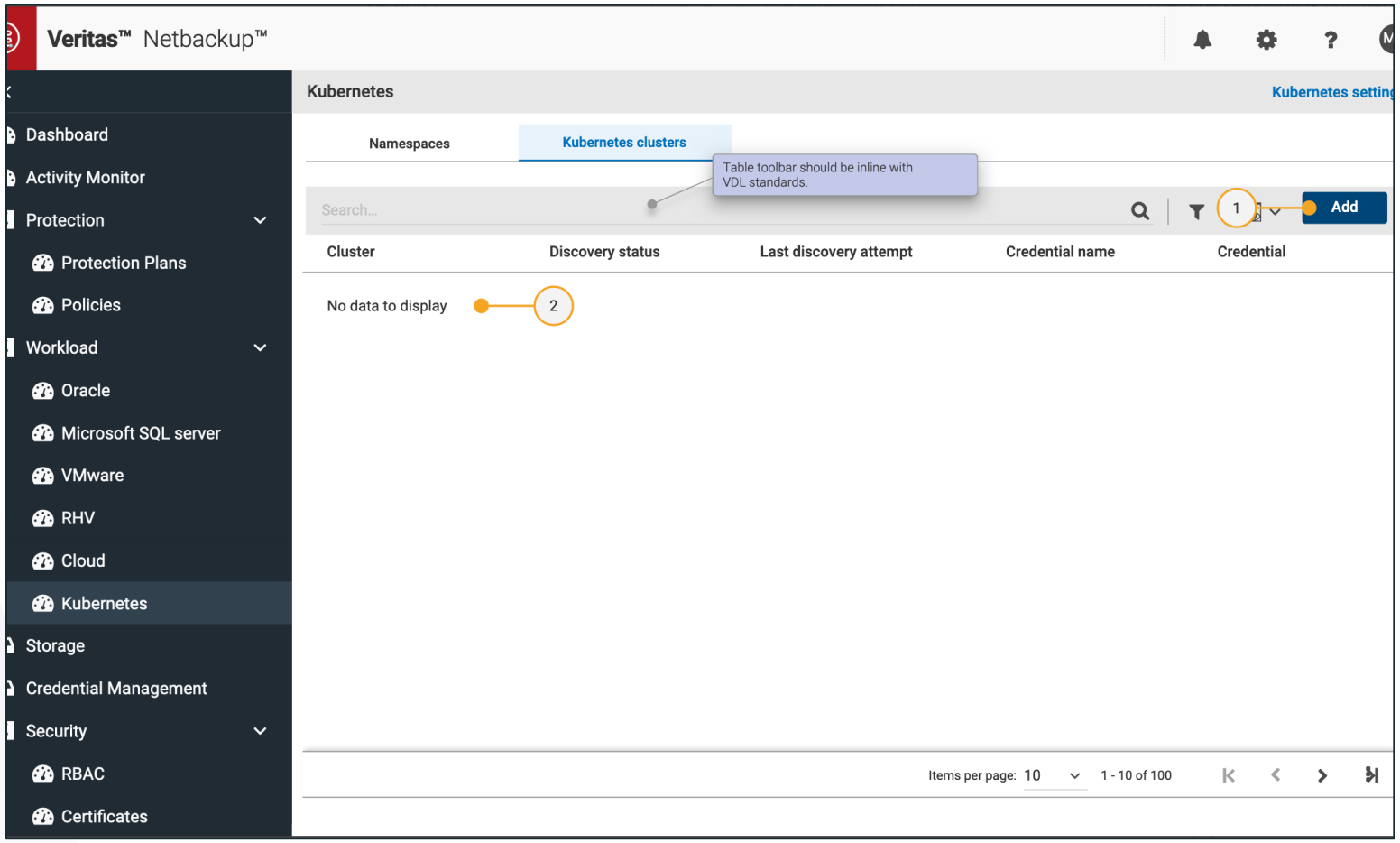
Example: Node Affinity

```
apiVersion: v1
kind: ConfigMap
metadata:
 name: backupserver.sample.domain.com
 namespace: netbackup
data:
  datamover.affinity: |
   nodeAffinity:
     requiredDuringSchedulingIgnoredDuringExecution:
       nodeSelectorTerms:
       - matchExpressions:
         - key: kubernetes.io/hostname
           operator: In
           values:
           - test1-l94jm-worker-hbblk
     preferredDuringSchedulingIgnoredDuringExecution:
     - weight: 1
       preference:
         matchExpressions:
         - key: beta.kubernetes.io/arch
           operator: In
           values:
           - amd64
 version: "1"
```

Add Kubernetes cluster - Create new credentials

Add Kubernetes cluster

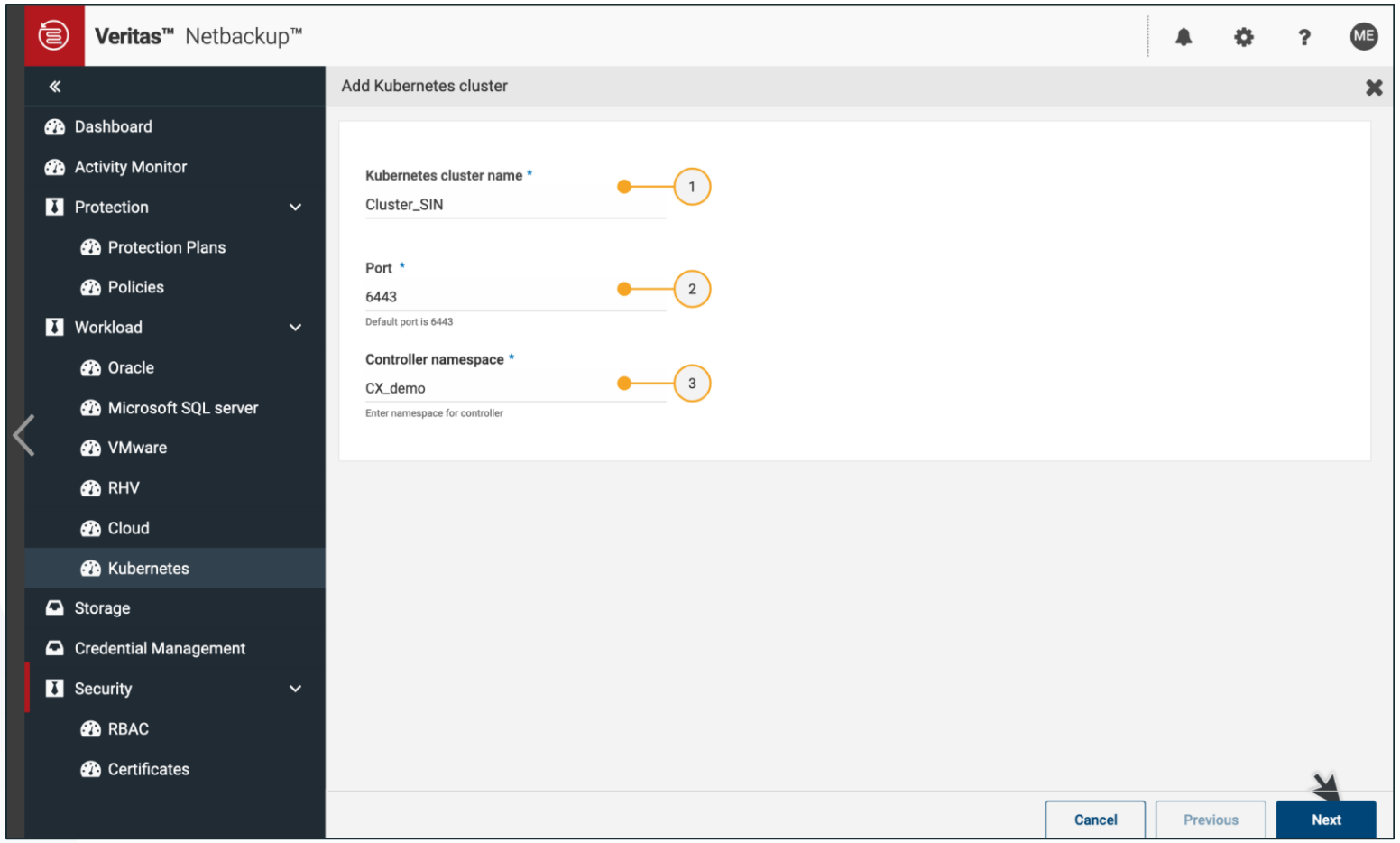
Add new credentials



1. Click **Add** to add a new Kubernetes cluster

Add Kubernetes cluster

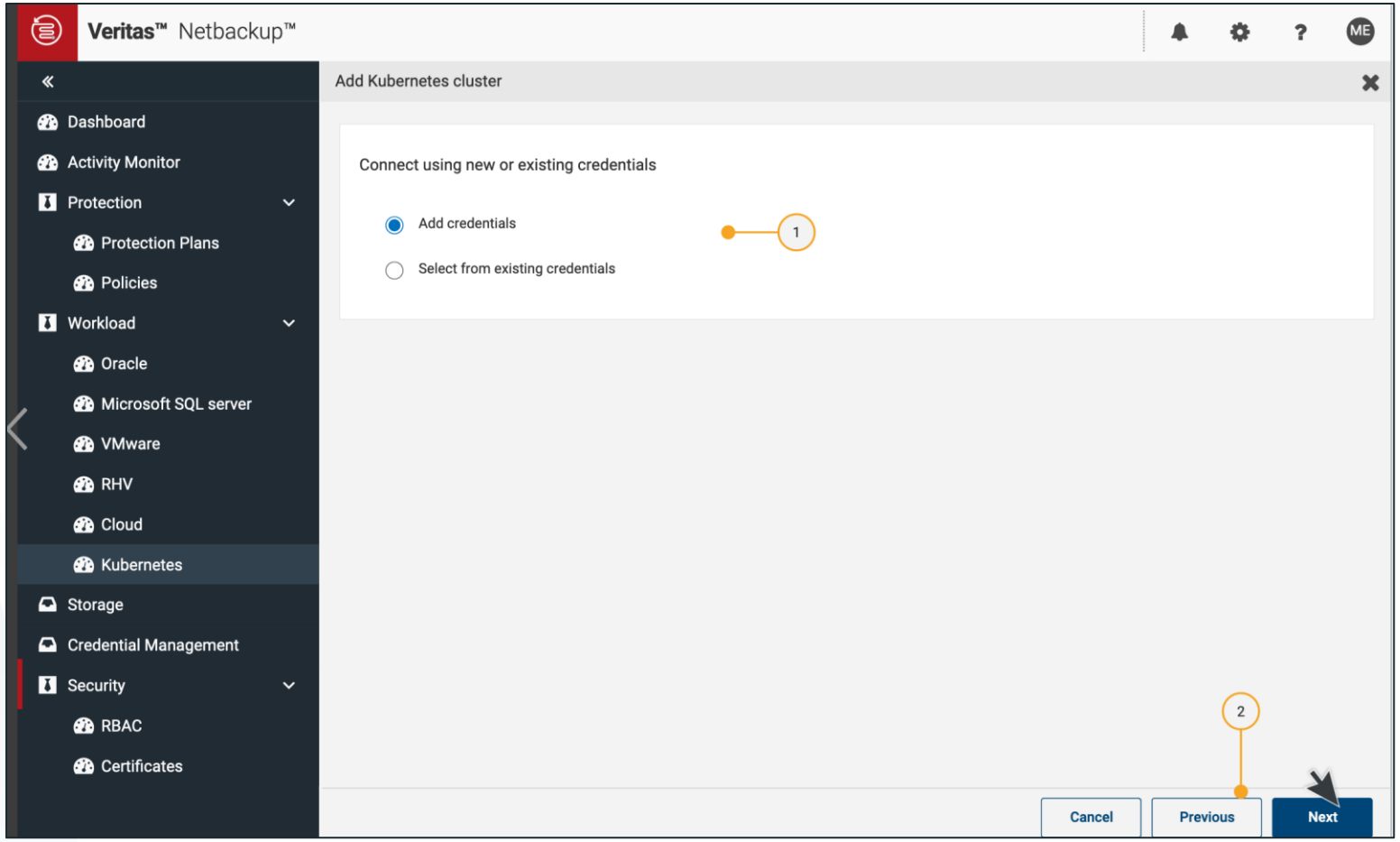
Add new credentials



2. Enter the following information
   1. FQDN for the Kubernetes cluster
   2. Enter the Kubernetes API server port number. Our default port is 6443
      - Default port is 6443 for Openshift, and Tanzu and 443 for GKE, AKS and EKS, and Vanilla
      - Run the command *kubectl cluster-info* for the Cluster name and Port number parameters to insert here
   3. Controller namespace where the NetBackup Kubernetes Operator is deployed
3. Click **Next**

Add Kubernetes cluster

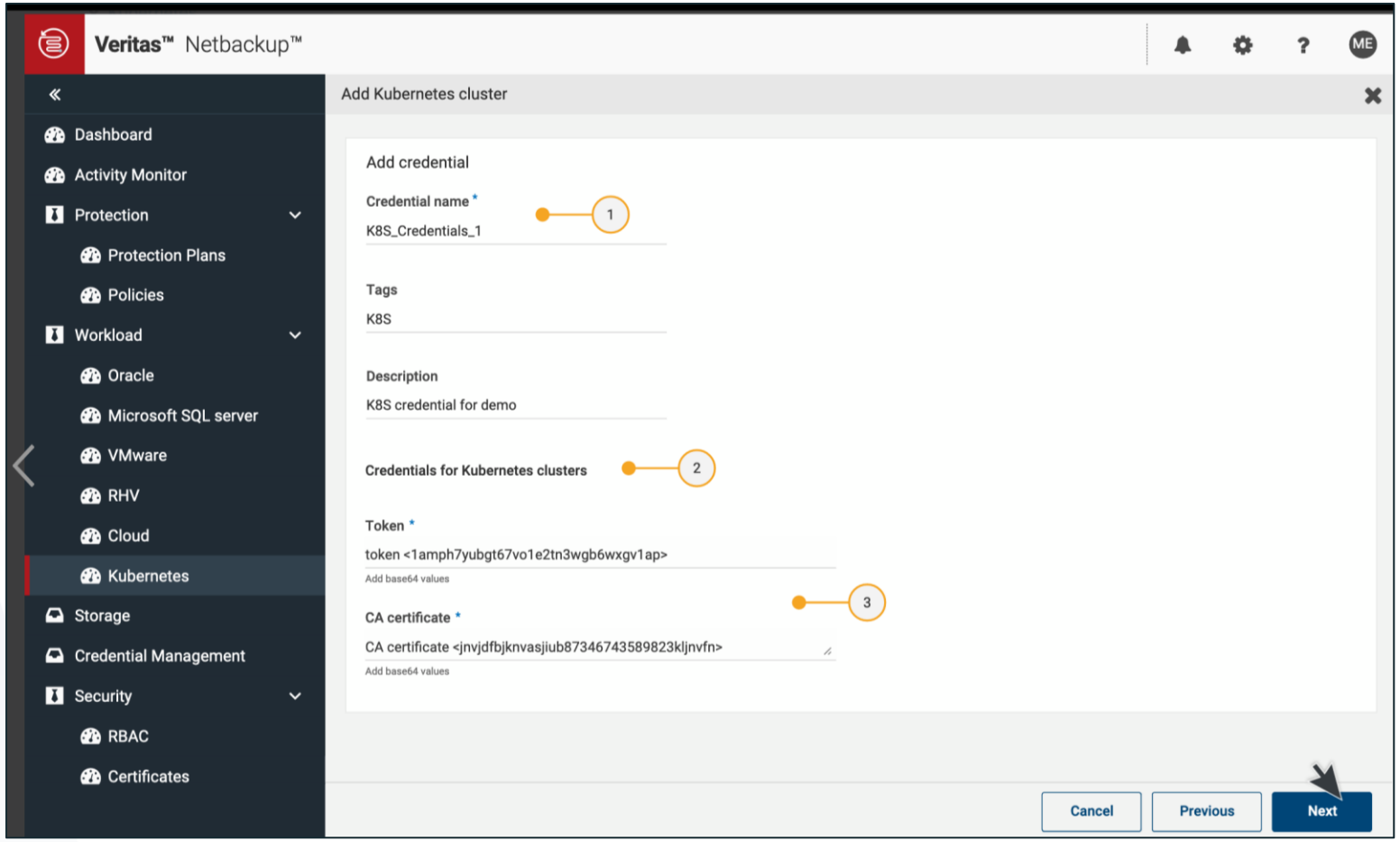
Add new credentials



4. Select **Add credentials.**
5. click **Next.**

Add Kubernetes cluster

Add new credentials



6. Provide a name, tag and description for the credential

   The Category type will be selected to Kubernetes by default

7. Add Token and CA Certificate you got from running the `kubectl get secret <[namespace-name]-backup-server-token-<id>> -n <namespace name> -o yaml` command (You want the ca-cert not service-ca-cert certificate)

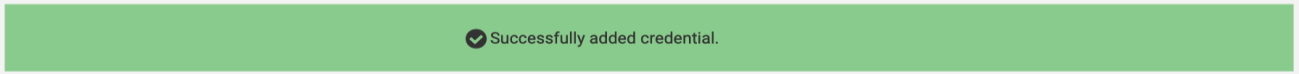
8. Click **Next.**

Add Kubernetes cluster

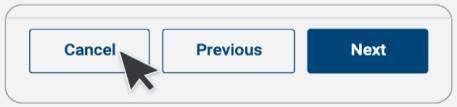
Add new credentials

Note

- On click of the 'Next' button the credential will be created and stored in the CMS.
- The notification message for the successful creation of the Credential will appear on the following screen.



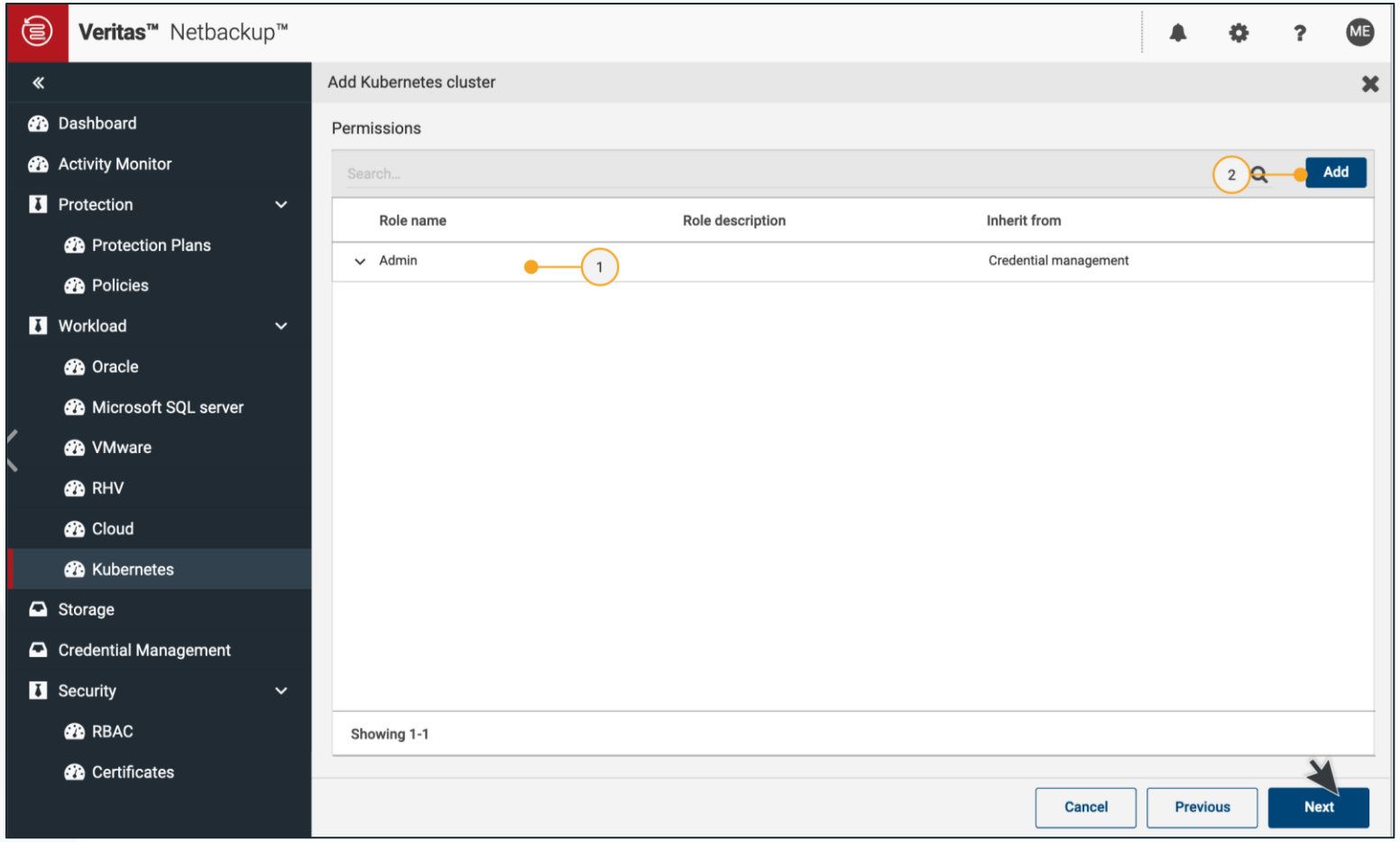
- The Permission screen has a dependency one the 'Manage access' parameter in the RBAC for the selected role.
- If the user does not have the permission then the permissions screen will be skipped.
- If the user click the 'Cancel' button on the footer.



The wizard will close without saving the user inputs.

- The credentials are validated and on successful validation, the cluster is added
- After the cluster is added, autodiscovery runs automatically to discover available assets in the cluster. This first discovery is a full discovery

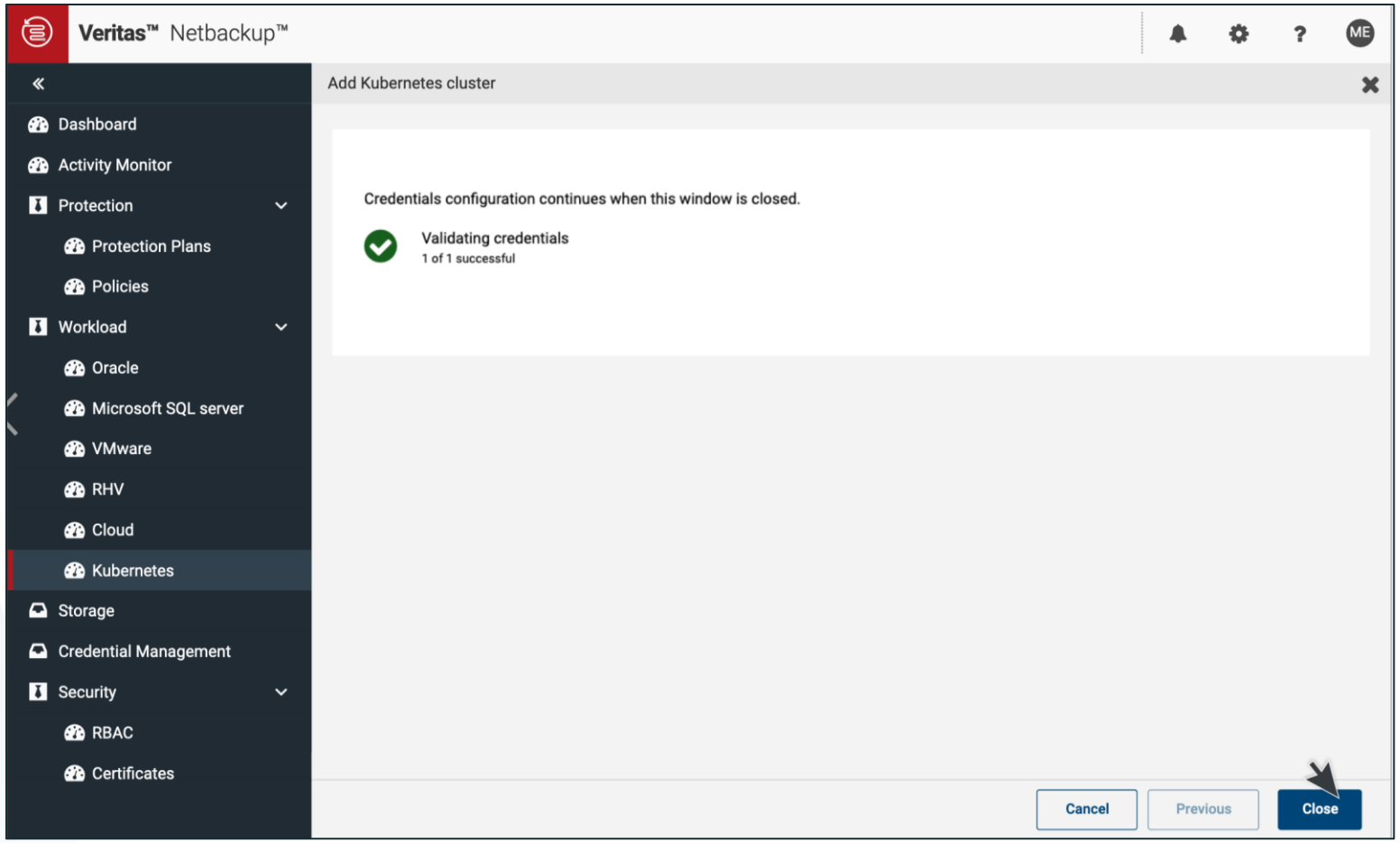
Add Kubernetes cluster



- The user role who created the credential will be assigned access by default.
- More roles can be assigned permissions using **Add** button.

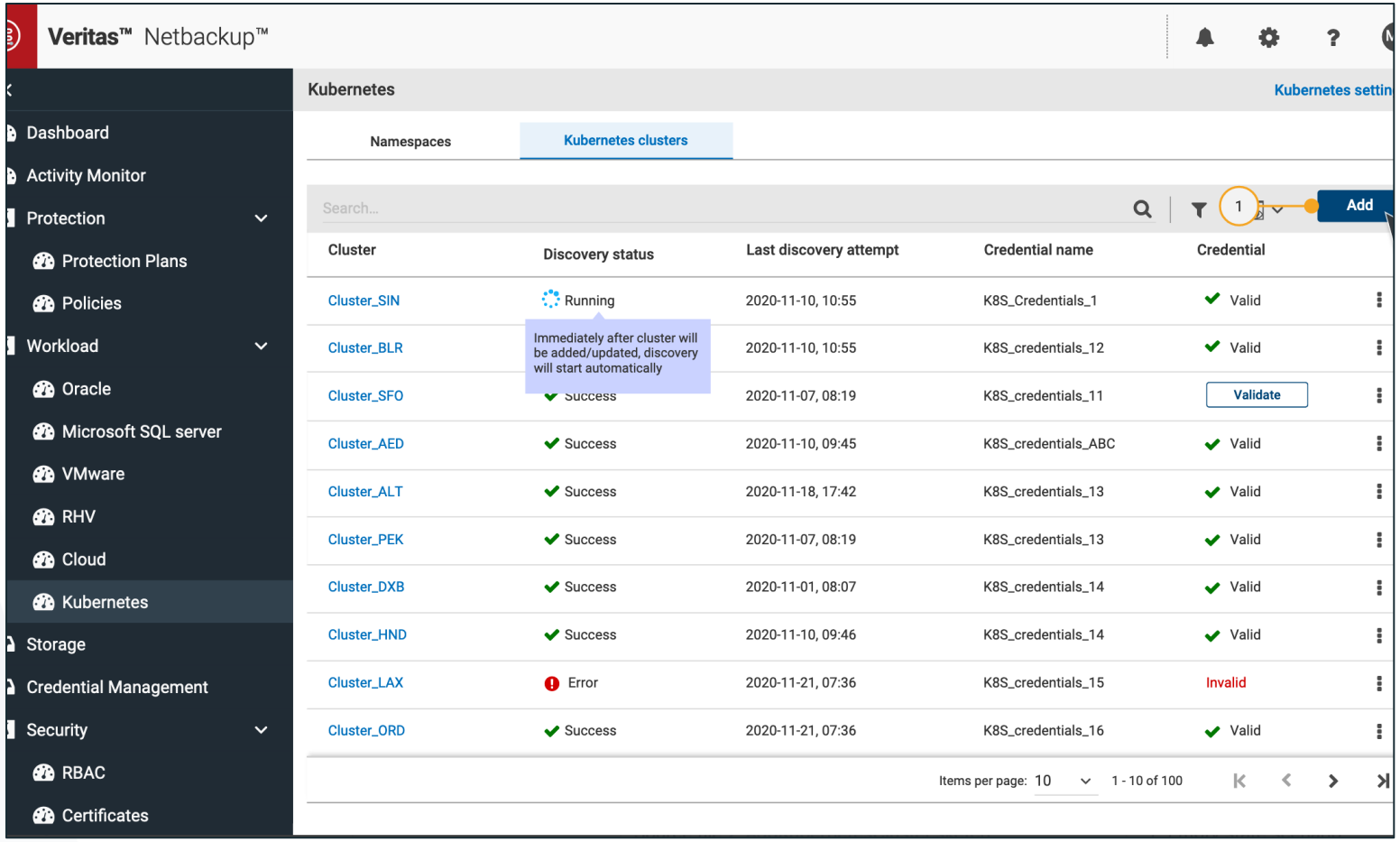
9. Click **Next.**

Add Kubernetes cluster



10. Click **Close.**

Add Kubernetes cluster



- Cluster credentials will show as Valid. If cluster is added the discovery will start automatically.

Add Kubernetes cluster

Note

Permission for the Kubernetes cluster

- The user who initiated the “Add Kubernetes cluster” will have a permission to access this cluster by default.
- If more permissions are required.
  Add permissions to the cluster can be done in the following 2 ways.

  1. From cluster details page
     Workload > Kubernetes > Cluster > Details > Permissions (tab) > Add Permissions

  2. From RBAC
     Security > RBAC > Add Role > Assign workloads.

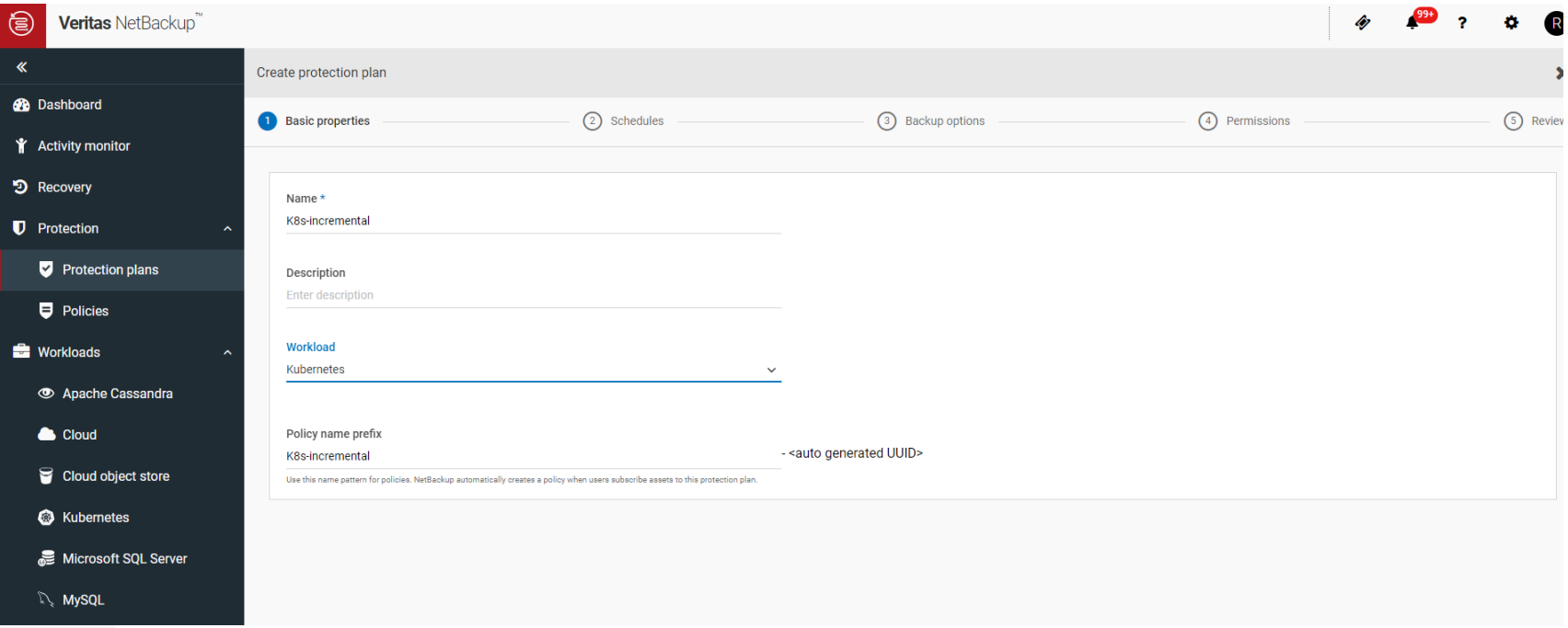
Incremental backup support

Kubernetes incremental backup

- You can choose to create incremental backups via protection plan for Kubernetes workload.
- NetBackup Kubernetes supports two types of incremental backup jobs:
  - Cumulative incremental
  - Differential incremental
- The type of backup needs to be set in the respective schedule while creating a Kubernetes protection plan.
- There is also an ‘Automatic’ backup type supported in order to ease creation of full + incremental backups.
- Apart from snapshot copy, backup from snapshot, duplicate copies have incremental support.

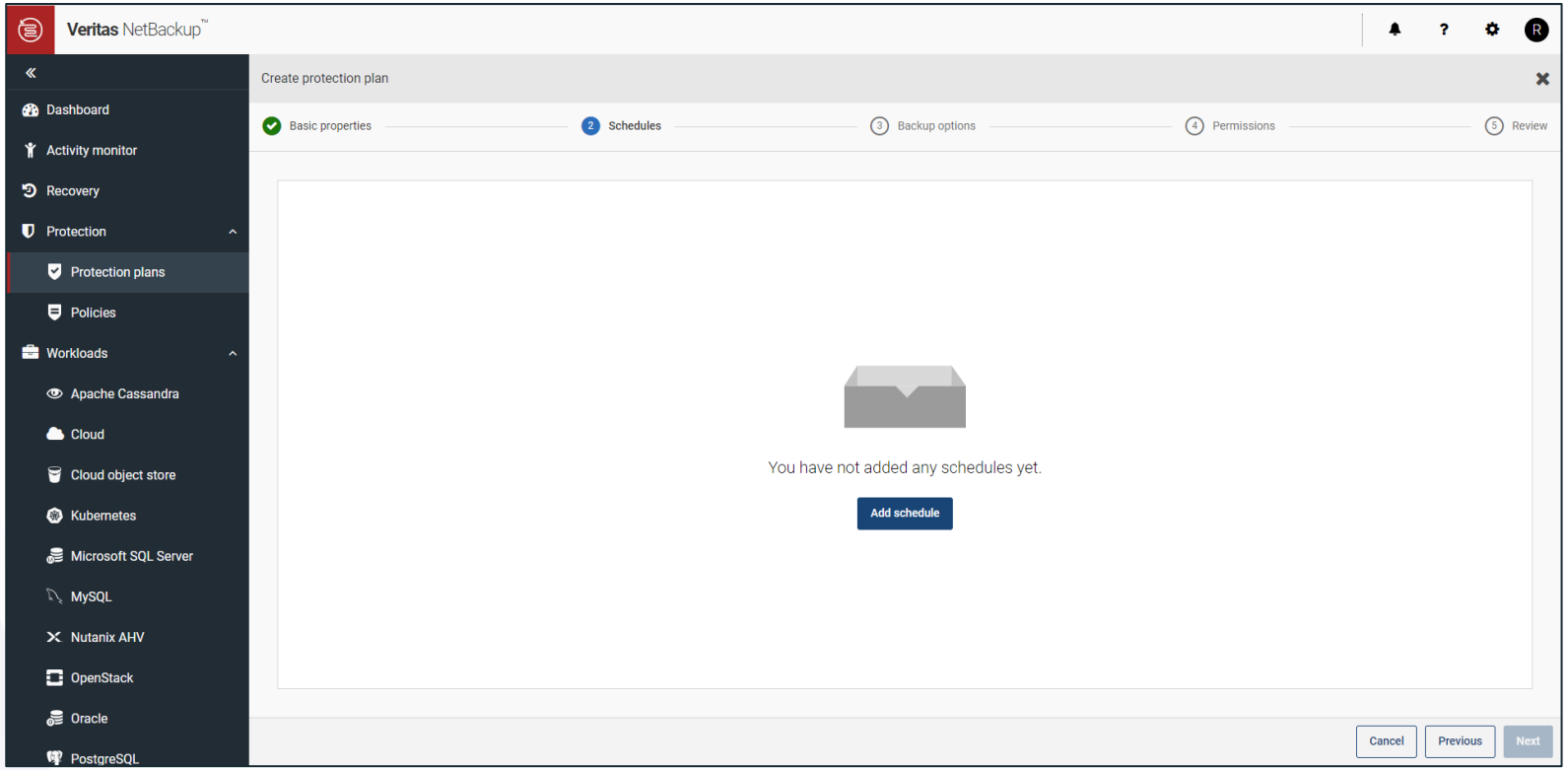
Create a protection plan

Basic properties



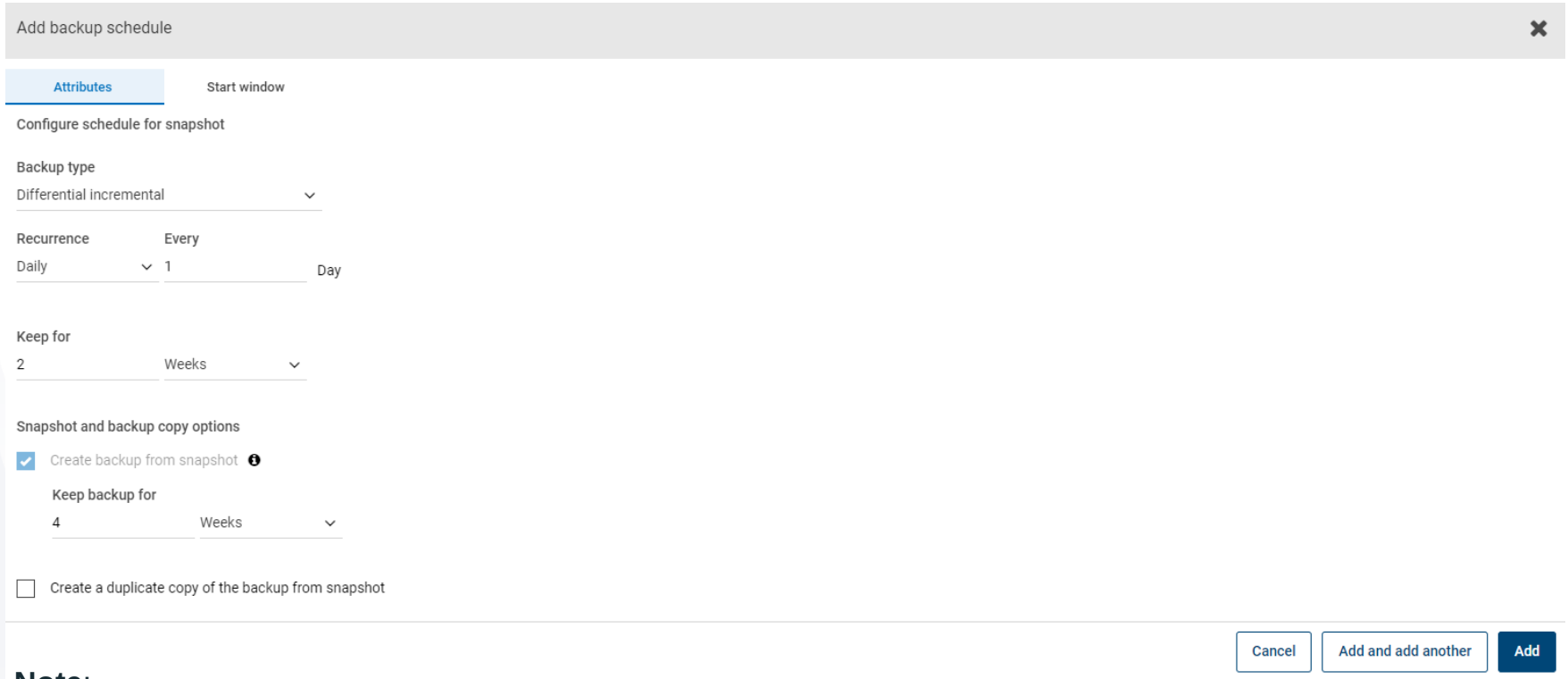
1. Enter name and description.
2. Select a Kubernetes from drop down list.
3. Click **Next**.

Create a protection plan
Schedules



4. Click **Add schedule**.

Steps to set a differential incremental backup job in protection plan



Note:

- Backup from Snapshot option will always be enabled by default for Incremental backup types and cannot be changed.
- AIR (Auto Image Replication) option is not applicable to Incremental backup types.
- It is recommended to add full backup schedule along with differential incremental schedule to have shorter recovery chain .

5. Select **Backup type -> Differential Incremental**

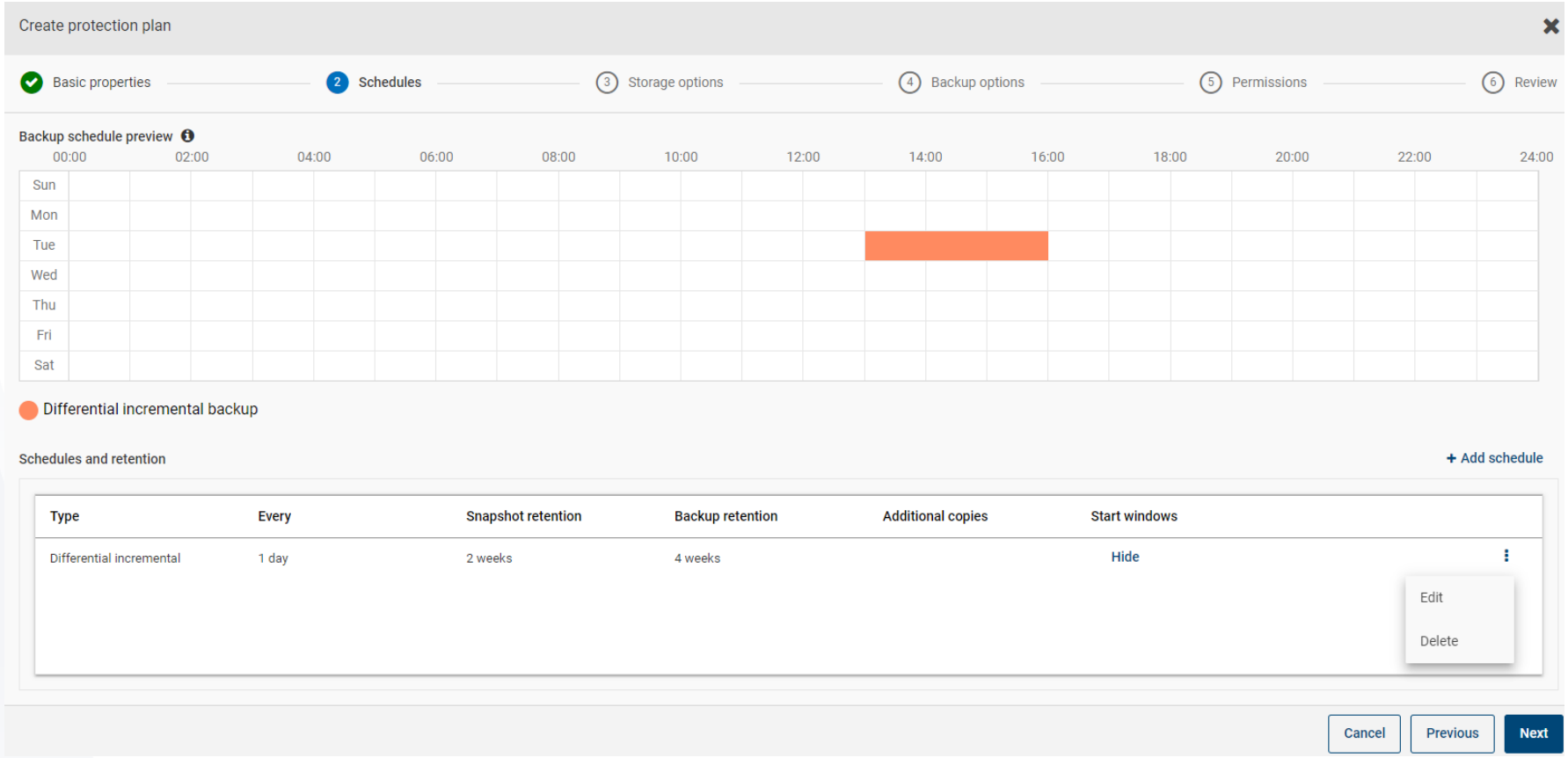
6. Under **Configure schedule for snapshot** set values for **Recurrence** and **Keep for** attribute

7. Under **snapshot and backup copy options,** enter value in the **Keep backup for** text field.

8. Click **Start Window** and provide backup window

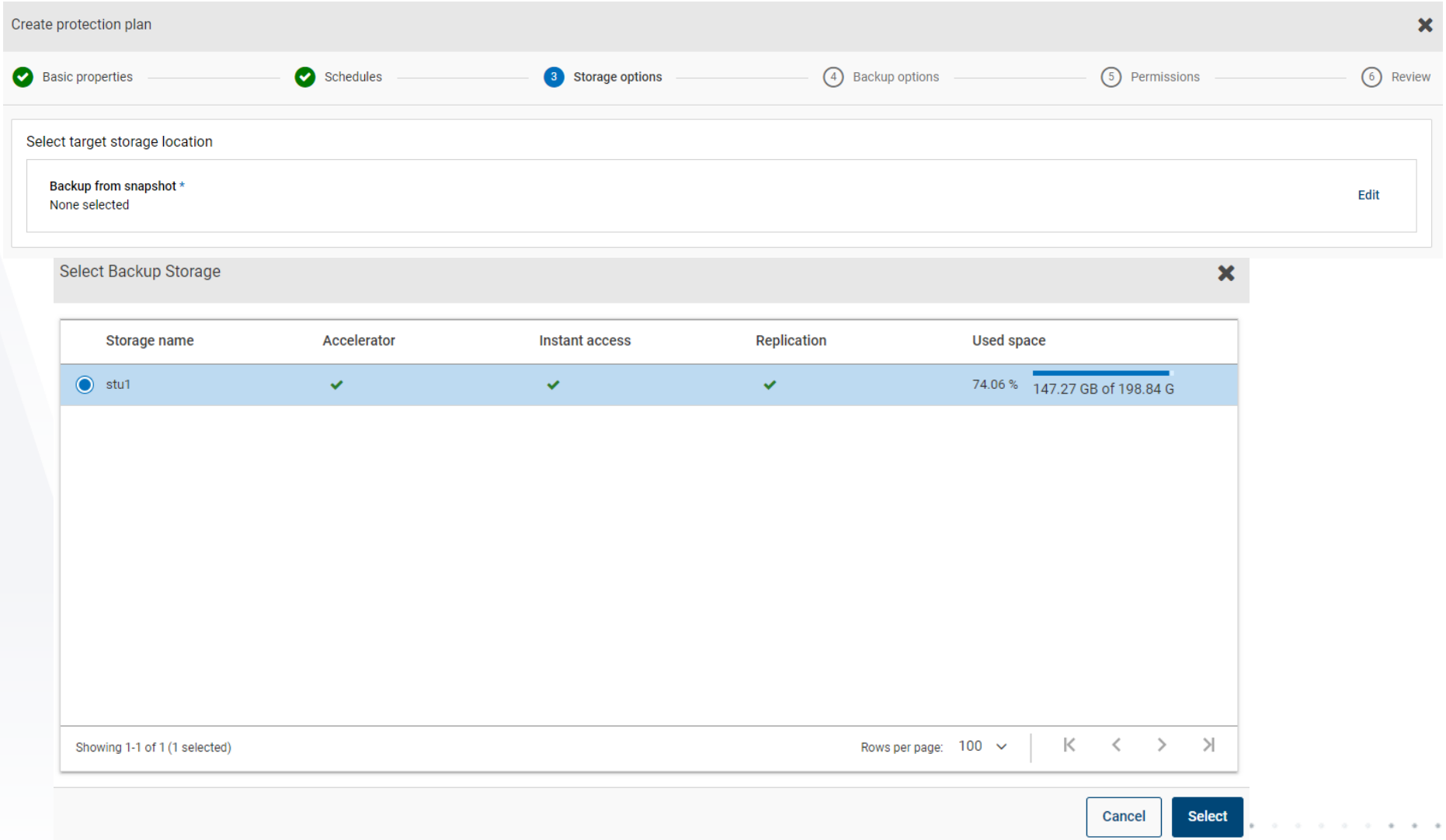
9. Click **Add**

Steps to set a differential incremental backup job in protection plan



- Differential schedule gets added in **Schedules and retention** section.
- Click **Edit** to change values set in the schedule.
- Click **Delete** to remove the schedule.

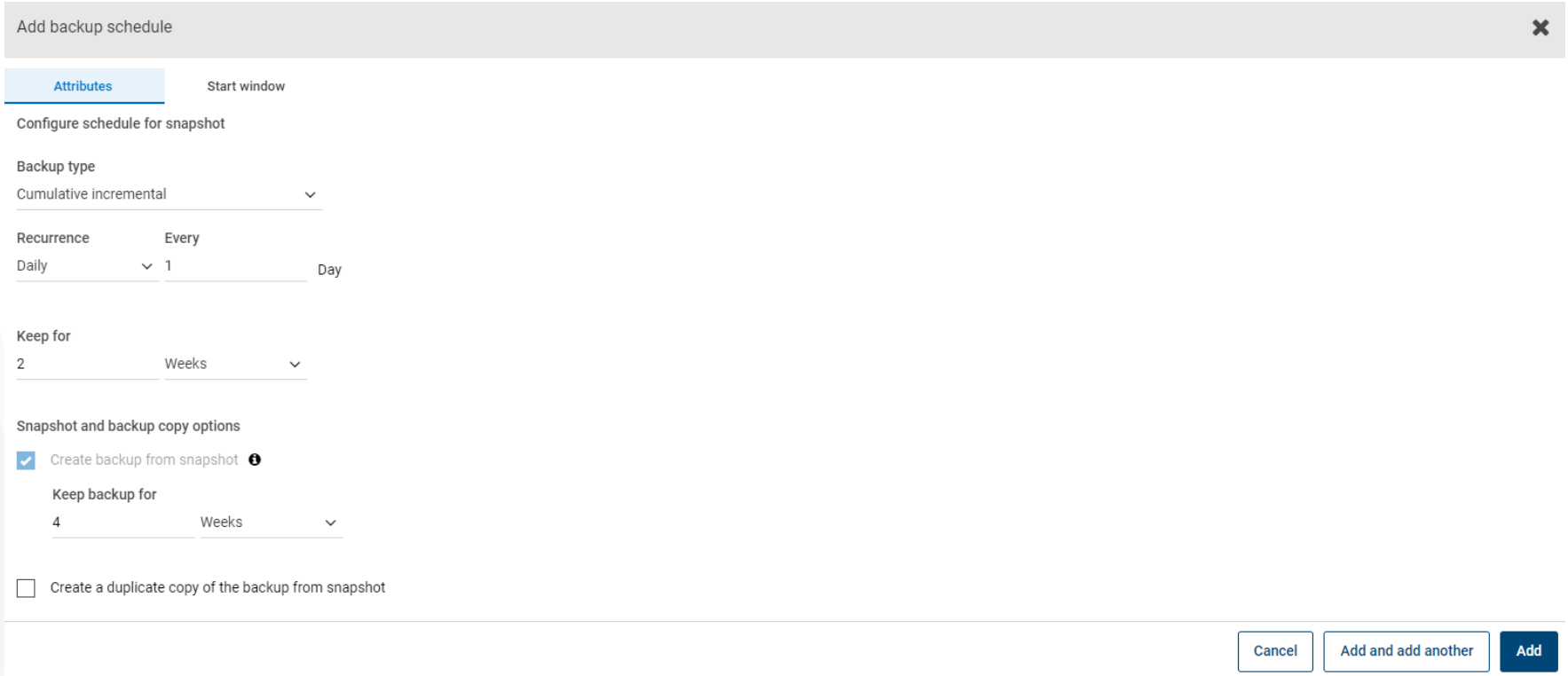
Steps to set a differential incremental backup job in protection plan



10. On **Storage options** page, click **Edit** to provide target storage location for **Backup from Snapshot.**

11. Select the desired storage unit.

Steps to set Cumulative incremental backup in Protection Plan



5. Select **Backup type** -> **Cumulative Incremental**

6. Under **Configure schedule for snapshot** set values for **Recurrence** and **Keep for** attribute

7. Under **snapshot and backup copy options** Mention value under **Keep backup for** attribute

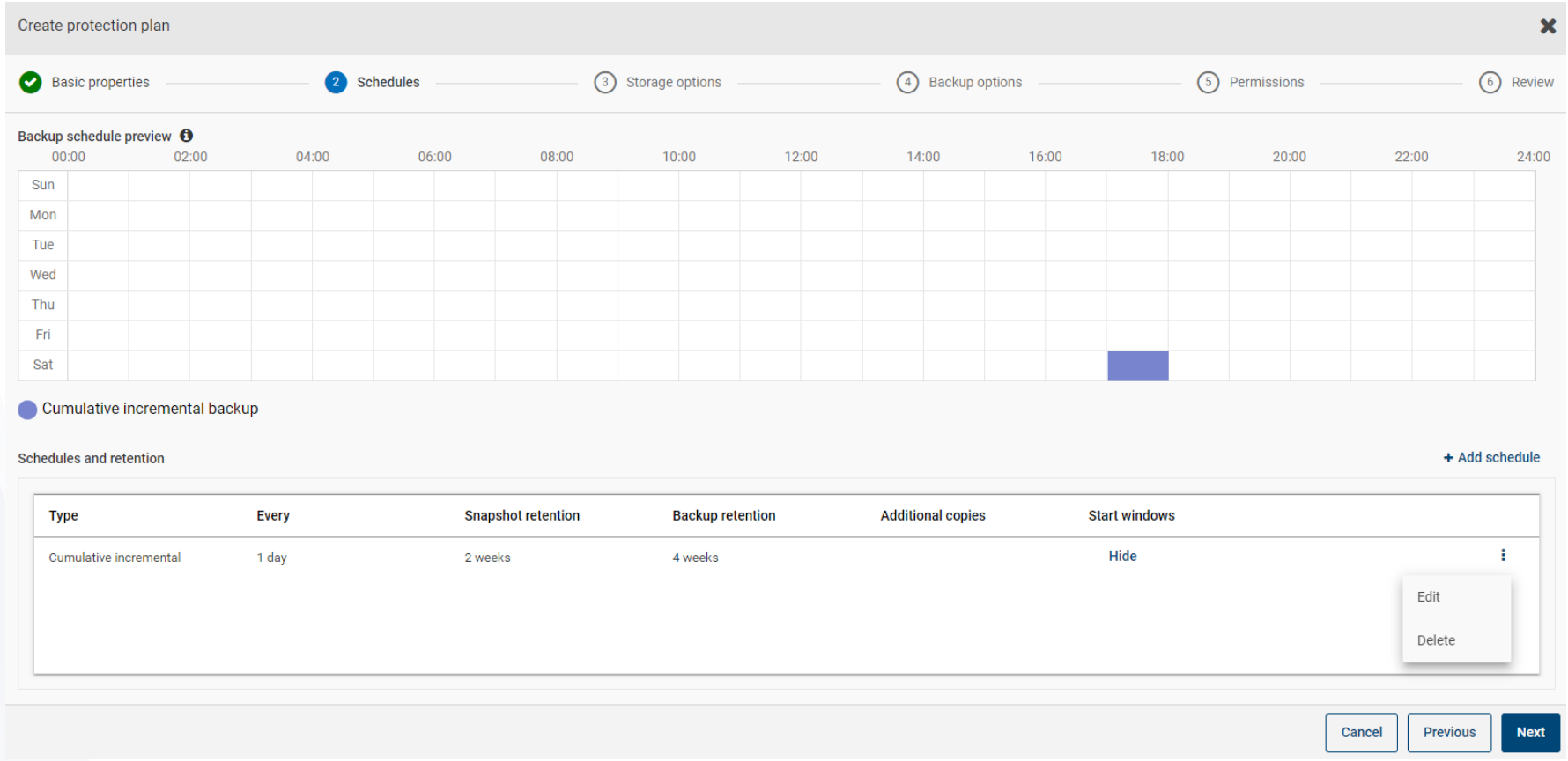
8. Click **Start Window**-> Provide backup window

9. Click **Add.**

Note:

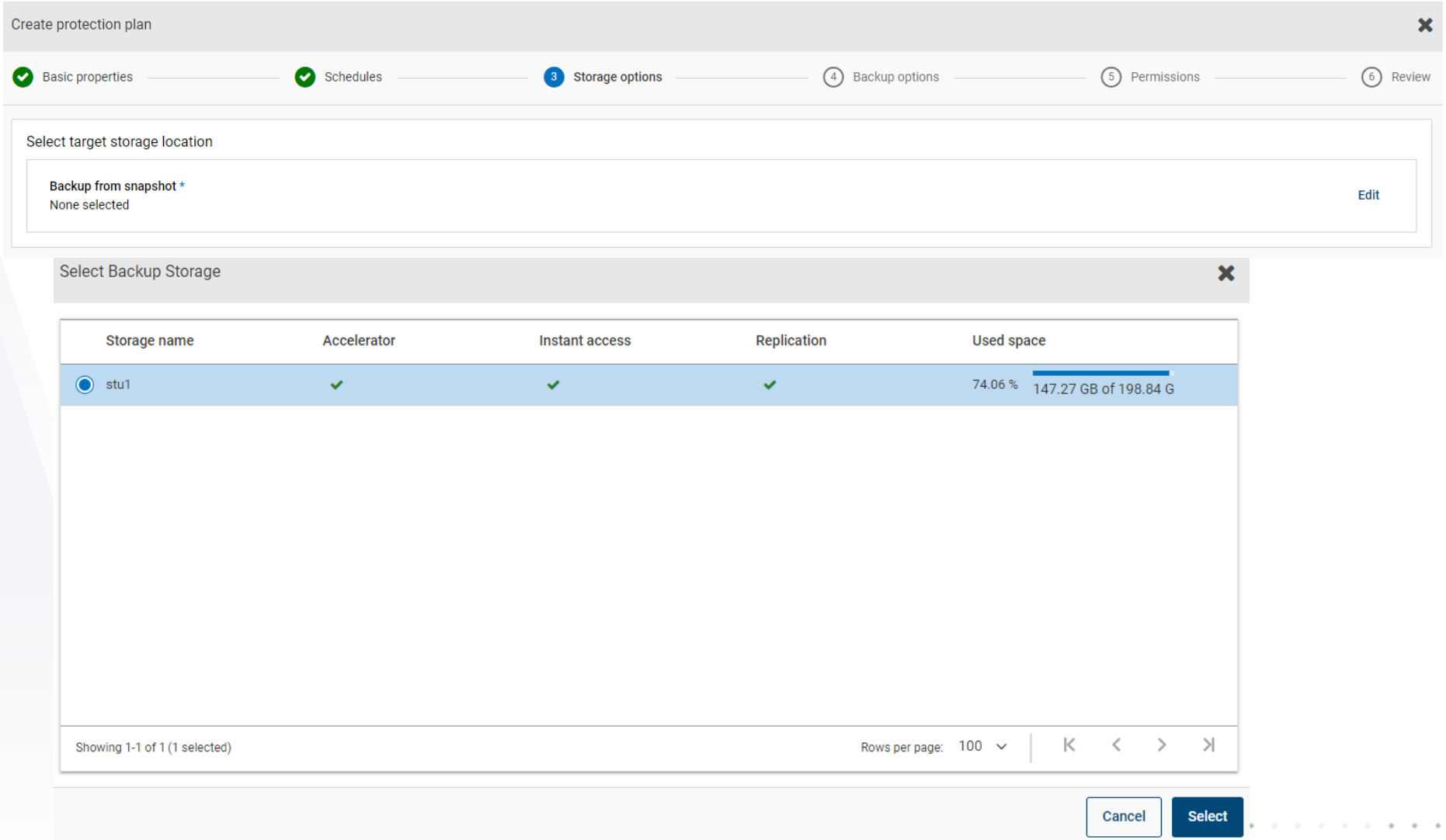
- Backup from Snapshot option will always be enabled by default for Incremental backup types and cannot be changed
- AIR (Auto Image Replication) option is not applicable to Incremental backup types
- Always add full backup schedule along with cumulative backup schedule otherwise every Cumulative backup is Full backup.

Steps to set Cumulative incremental backup in Protection Plan



- Cumulative schedule gets added in **Schedules and retention** section.
- Click **Edit** to change values set in the schedule.
- Click **Delete** to remove the schedule.

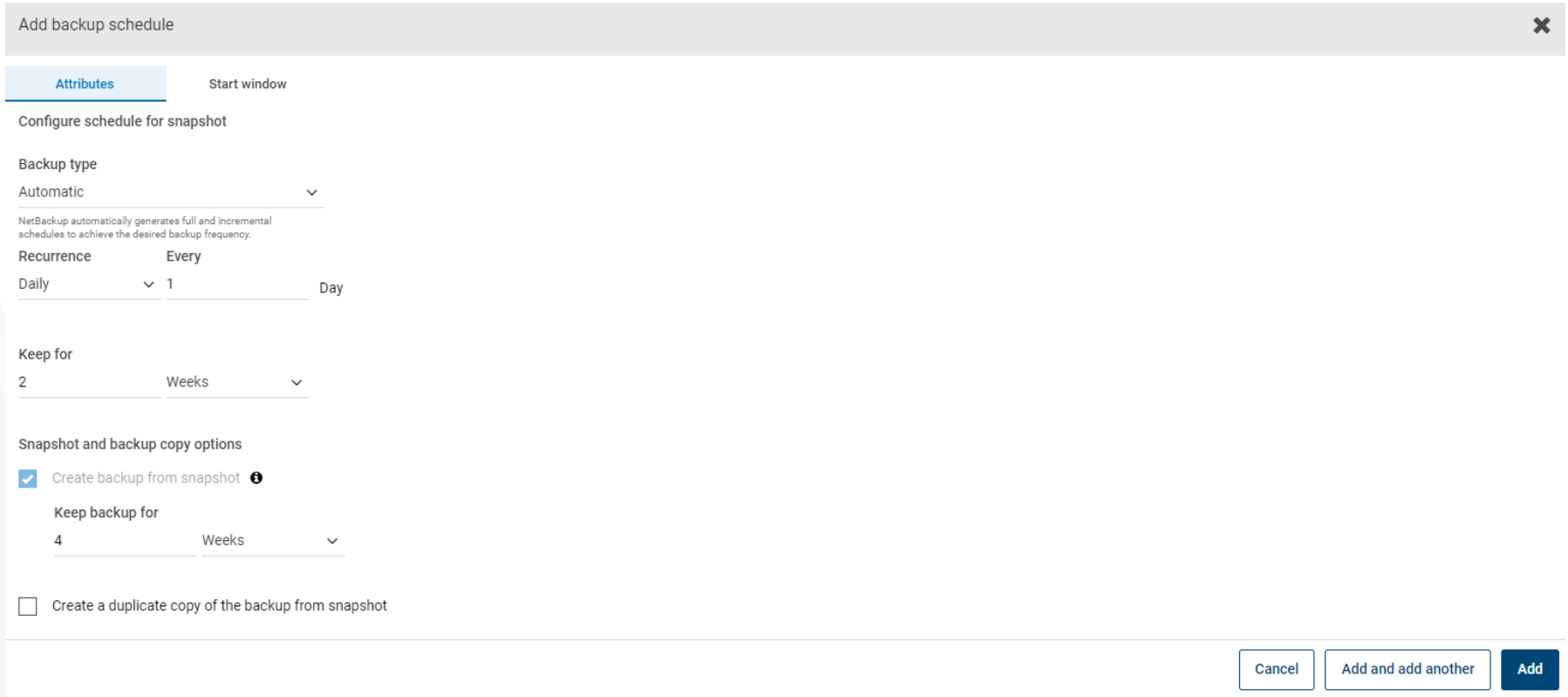
Steps to set Cumulative incremental backup in Protection Plan



10. On **Storage options** page, click **Edit** to provide target storage location for **Backup from Snapshot**

11. Select the desired storage unit.

Steps to set Automatic schedule in Protection Plan



- Backup from Snapshot option will always be enabled by default for Automatic backup type and cannot be changed
- AIR (Auto Image Replication) option is not applicable to Automatic backup type

5. Select **Backup type -> Automatic**

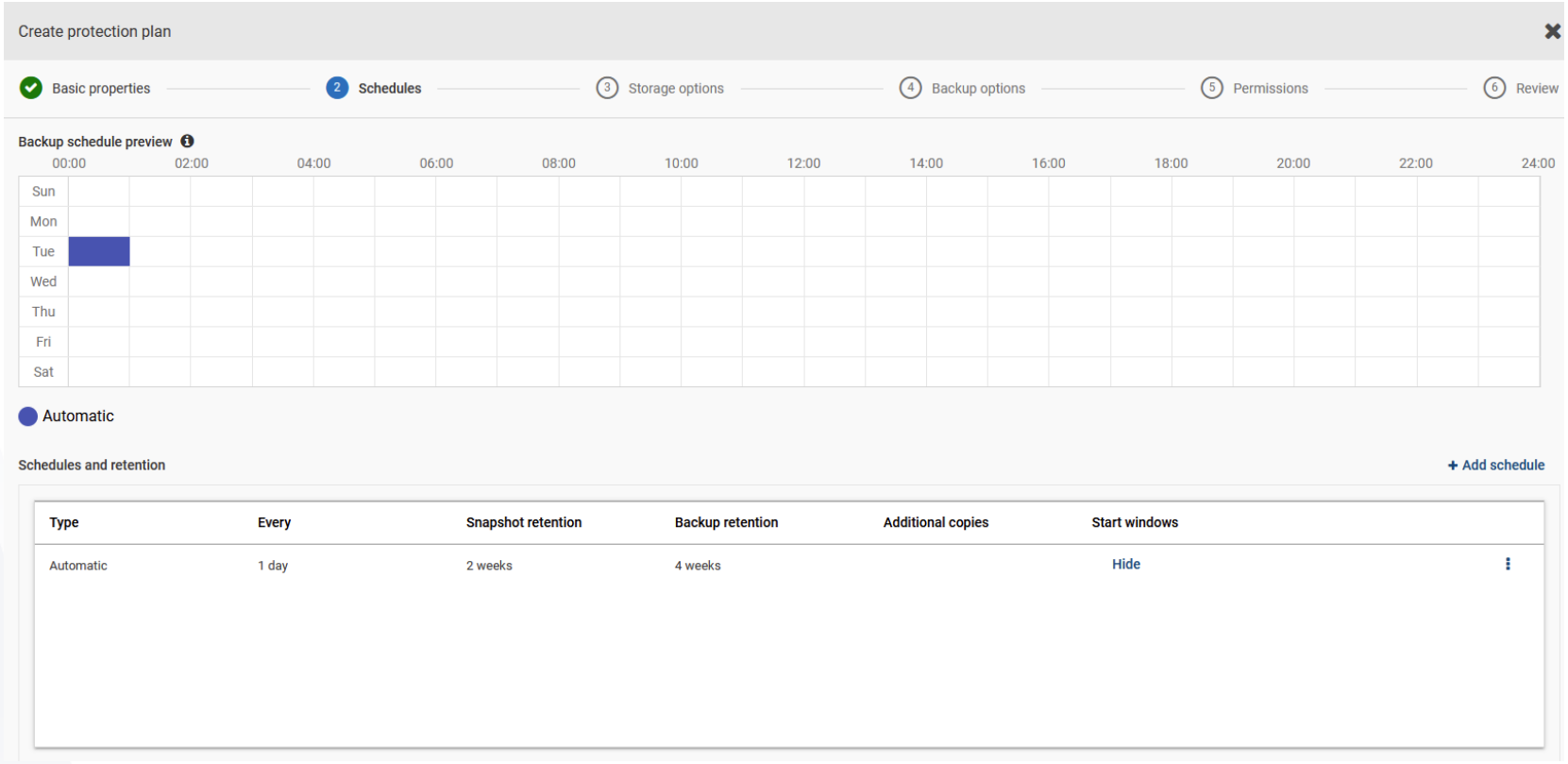
6. Under **Configure schedule for snapshot** set values for **Recurrence** and **Keep for** attribute

7. Under **snapshot and backup copy options** Mention value under **Keep backup for** attribute

8. Click **Start Window** to provide backup window

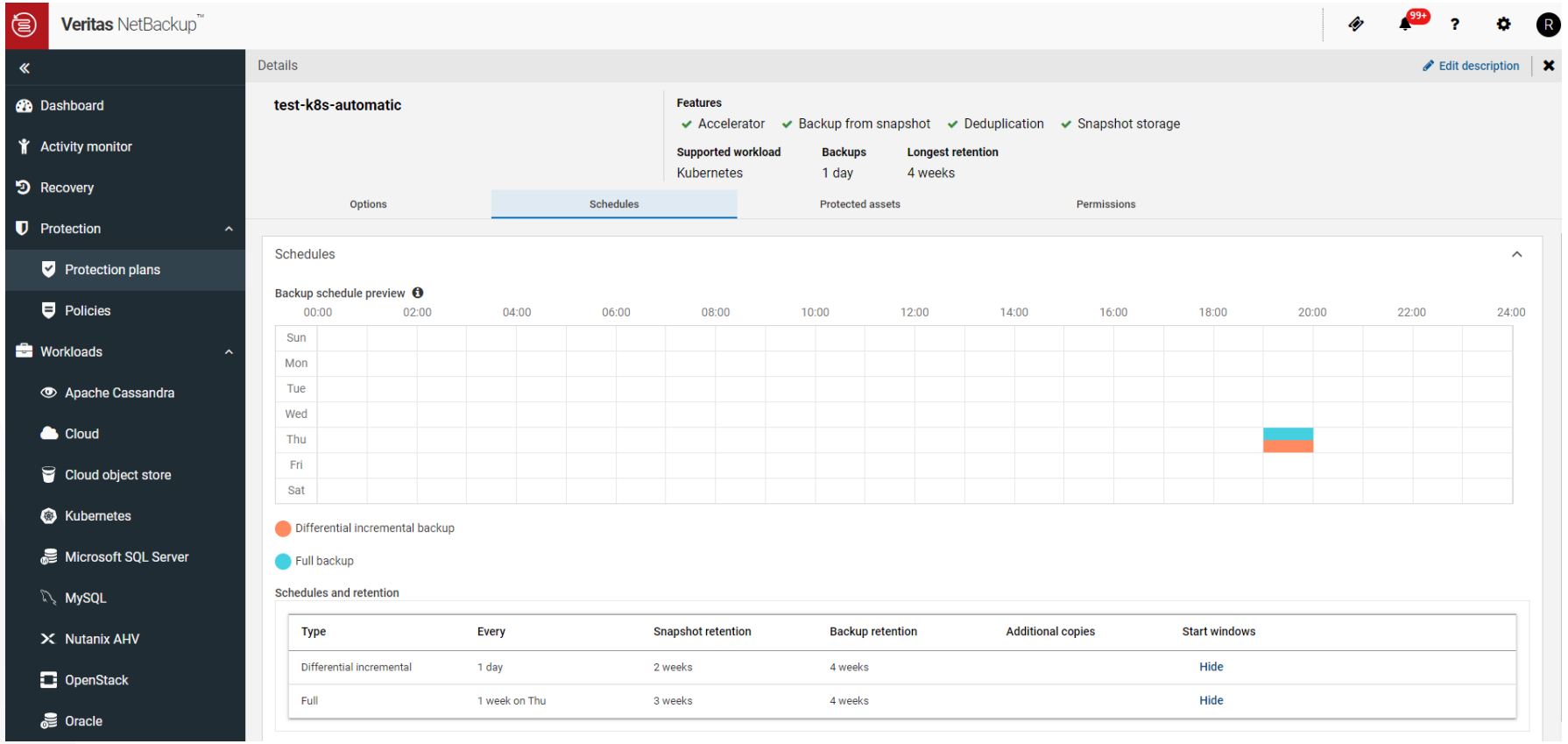
9. Click **Add.**

Steps to set Automatic schedule in Protection Plan



- Automatic schedule gets added in **Schedules and retention** section.
- Click **Edit** to change values set in this schedule.
- Click **Delete** to remove this schedule.

Steps to set Automatic schedule in Protection Plan



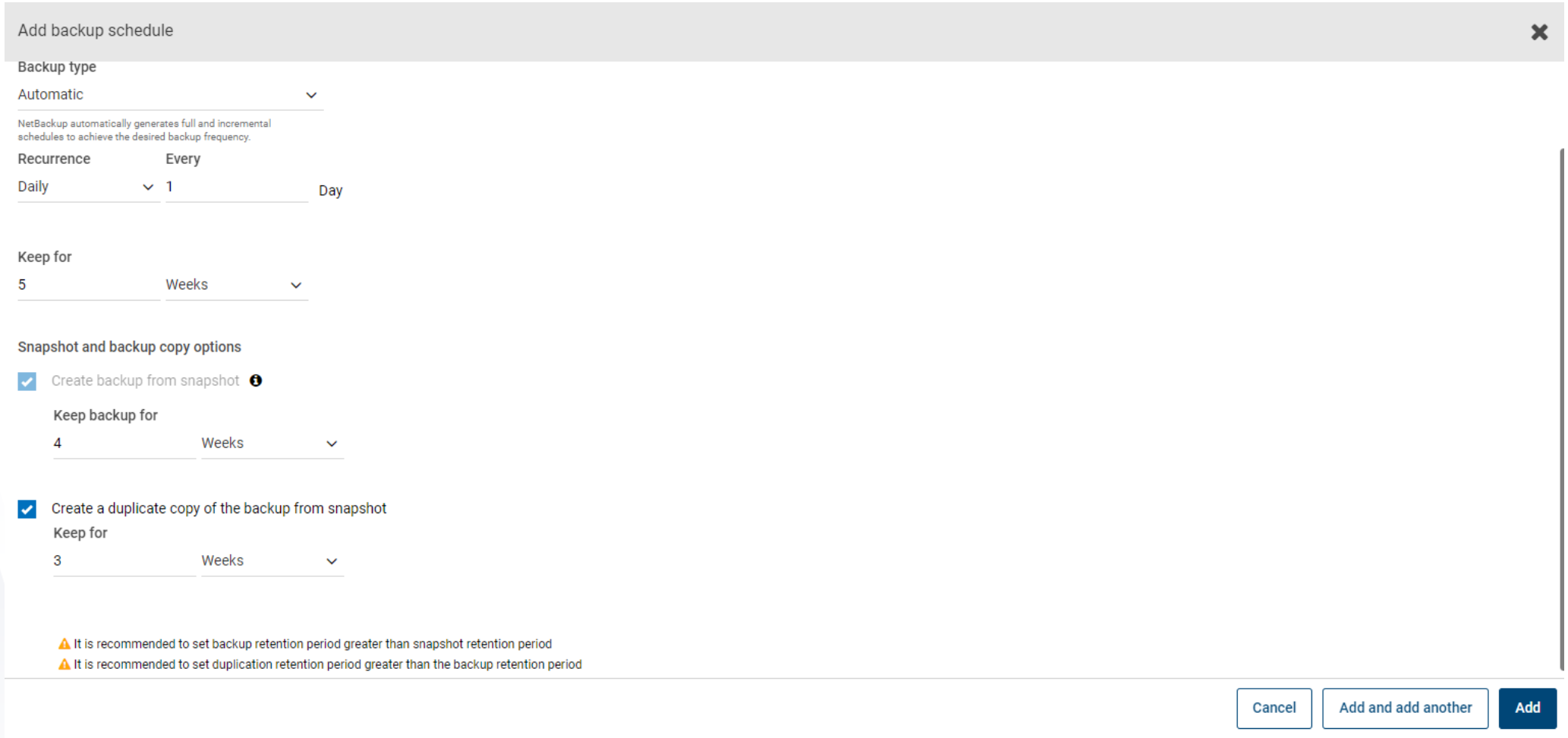
After a protection plan with an Automatic Schedule is created, there will be two schedules created

- If recurrence is less than 1 week, one differential schedule and one full schedule will be created
- If recurrence is >= 1 week, only full schedule will be created

Recommendations for Retention values

- To perform a restore from recovery point from any copies (snapshot, backup from snapshot, duplicate), it is recommended to keep retention of that copy longer.
- For example, to restore from a backup copy, retention of backup from snapshot > snapshot, otherwise backup copy will get expired, and recovery point will be marked as COMPLETE = NO
- Warnings for the same will be shown in web UI as follows.
- It is recommended to set backup retention period greater than snapshot retention period.
- It is recommended to set duplication retention period greater than the backup retention period.

Recommendations for retention values

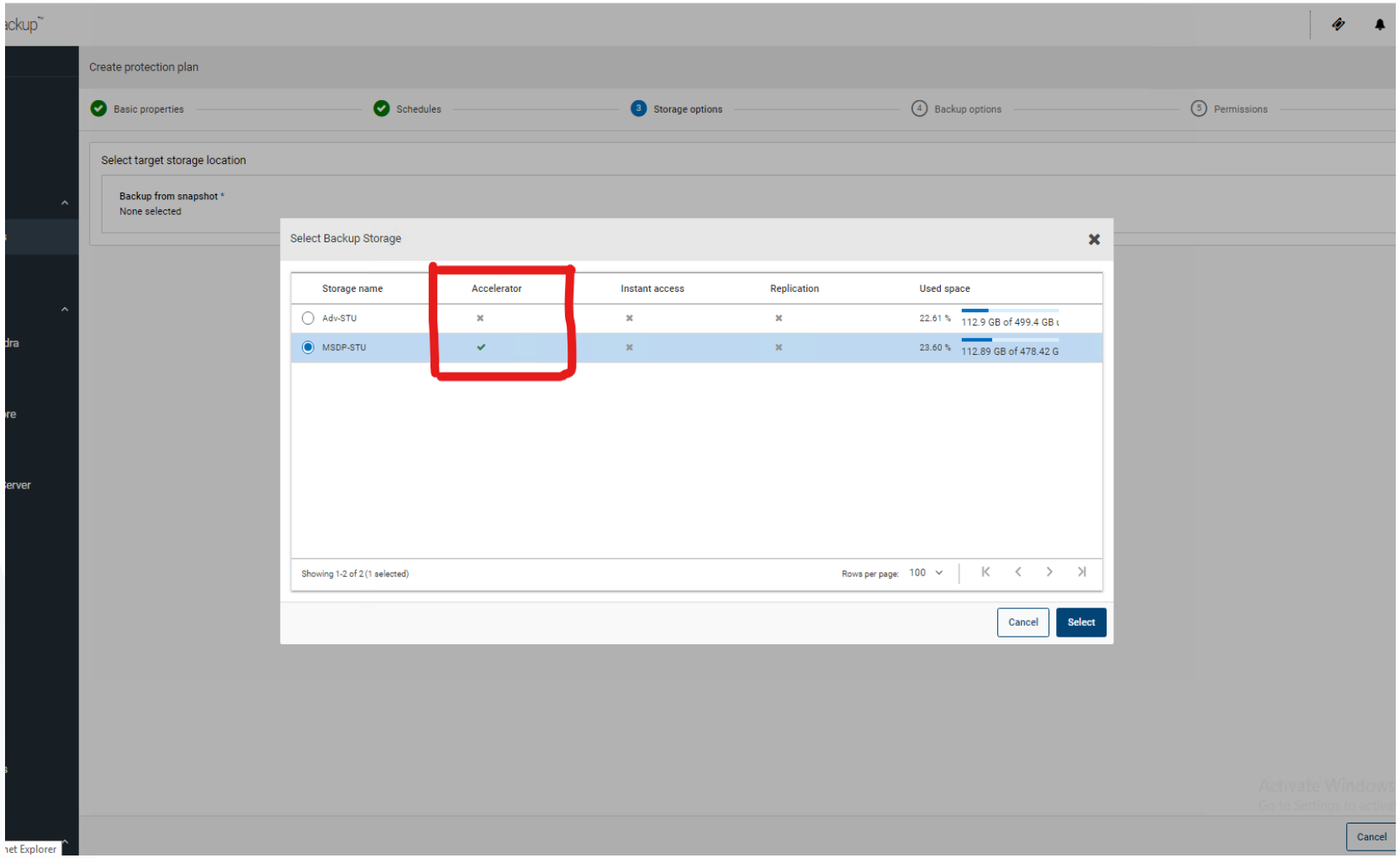


Accelerator backup support

Prerequisites for Accelerator backups

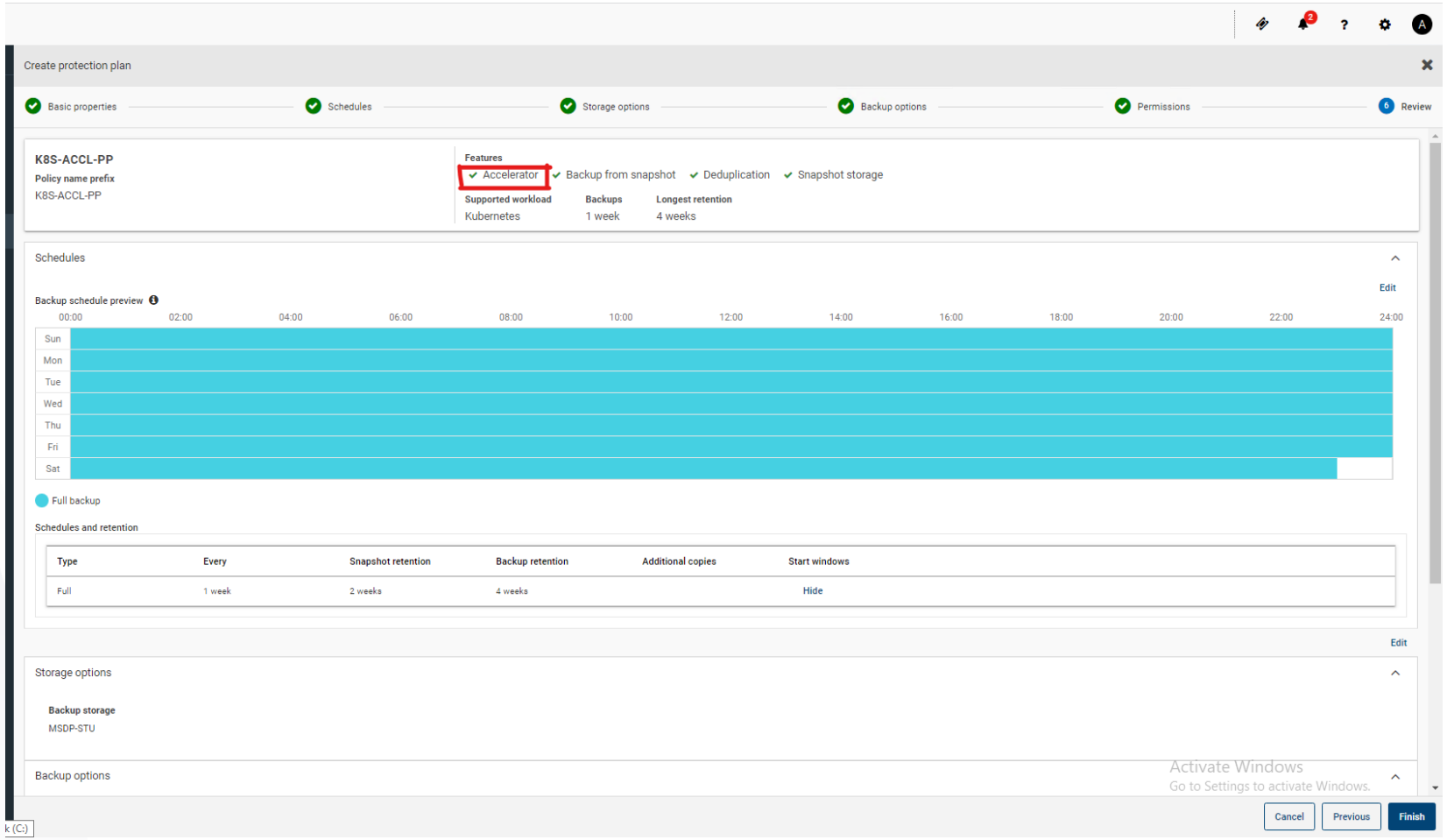
- Accelerator-supported NetBackup storage targets like MSDP, MSDP-C, OpenStorage Technology (OST) should be created before creating protection plan
- Storage class for provisioning Tracklog PVC

Steps to configure accelerator backups



1. Select **Accelerator supported Storage** while creating a Protection plan

Steps to configure accelerator backups



2. The Protection plan creation Summary page will list the Accelerator feature.
3. Before you click Finish, ensure Accelerator is listed there.

Steps to configure accelerator backups

```
netbackupkops:
  containers:
    manager:
      image: nbk8splugin.nbartifactory.rsv.ven.veritas.com/10.3.1/nbk8splugin:netbackupkops_10.3.1_0037
      resources:
        limits:
          cpu: 150m
          memory: 600Mi
        requests:
          cpu: 100m
          memory: 500Mi
  kopsPvcStorageClass:
  kopsPvcSize: 10Gi
  pvMountPath: /usr/openv
  imagePullSecrets: []
  # Example to add imagePullSecrets
  # imagePullSecrets:
  # - name: netbackupkops-docker-cred
  fipsMode: DISABLE
  acceleratorTracklogPvcStorageClass: ocs-storagecluster-ceph-rbd
nbsetup:
  replicas: 1
  containers:
    netbackup_config_pod:
      nbprimaryserver: r7515-117-v56.vxindia.veritas.com
      nbsha256fingerprint: E2:16:C6:2D:28:80:24:29:A7:2D:C3:50:3D:23:02:F5:6F:5F:42:43:F0:77:B2:BE:BC:45:C4
      k8sCluster: api.lab.ocp.lan
      k8sPort: 6443
      datamoverimage: k8s-nb-support.nbartifactory.rsv.ven.veritas.com/qa/datamover:10.3.1_0108
      storageclassblock: ocs-storagecluster-cephfs
      volumesnapshotclassblock: ocs-storagecluster-cephfsplugin-snapclass
      storageclassfilesystem: ocs-storagecluster-cephfs
      volumesnapshotclassfilesystem: ocs-storagecluster-cephfsplugin-snapclass
~
```

4. Storage for Tracklog can be provisioned in two ways.
5. While deploying NBUKOPs, update **"acceleratorTracklogPvcStorageClass"** in values.yaml file

VERITAS

Steps to configure accelerator backups

```
# Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving this file will be
# reopened with the relevant failures.
#
apiVersion: v1
data:
  acceleratorTracklogPvcStorageClass: ocs-storagecluster-cephfs
  checkNbcertdaemonStatusDurationMinutes: "1440"
  cleanStaleCRDurationMinutes: "1440"
  cleanStaleCertFilesDurationMinutes: "1440"
  clusterName: api.lab.ocp.lan
  collectDataMoverLogs: Failed
  fipsMode: DISABLE
  livenessProbePeriodSeconds: "180"
  livenessProbeTimeoutInSeconds: "1"
  livesnessProbeInitialDelay: "60"
  maxRetentionDataMoverInHours: "24"
  maxRetentionDataMoverLogsInHours: "72"
  maxRetentionInDiscoveryCacheHours: "48"
  maxRetentionRestoreJobMetaInHours: "72"
  namespace: shakti-nbukops-ns
  nbcertPrerequisteDirectoryAndFiles: |
    vxssDirPath=/usr/openv/var/vxss
    bp.conf=/usr/openv/netbackup/bp.conf
    atLibPath=/nbcertcmdtool
    fingerprintDirPath=/usr/openv/fingerprint-dir
  pollingFrequencyInSecs: "5"
  pollingTimeoutInMinutes: "15"
```

Or

6. Updating Configmap - "backup-operator-configuration" after NetBackup Kubernetes operators' deployment.
7. Run the command: kubectl edit cm <ns>-backup-operator-configuration –n <ns>
8. Change **"acceleratorTracklogPvcStorageClass"**

Kubernetes Operators install and upgrade changes
Validate accelerator tracklog storage class

- The storage class should be such that it allows creation of file mode PVC.
- During install and upgrade, if accelerator is enabled on Kubernetes cluster, ('acceleratorTracklogPvcStorageClass' is provided in values.yaml) NetBackup Kubernetes Operator creates a file mode PVC and a pod to check if the given storage class is valid.
- If volume binding mode of the storage class is 'Immediate', it only creates a PVC, and installation is successful if the PVC is in Bound state.
- If volume binding mode of the storage class is 'WaitForFirstConsumer' then, it creates a datamover pod with a PVC. Installation is successful if PVC is in Bound state and pod is in running state.

Kubernetes Operators install and upgrade changes
Validate minimum size provided for the creation of accelerator tracklog PVC

- In NetBackup release 10.4.1 and later, if a user provides a value less than the minimum, the installation or upgrade of Netbackupkops will fail, and the user must adjust the value accordingly.
- If the "acceleratorTracklogPvcStorageClass" attribute in values.yaml is other than "None" during installation, a PVC of the size specified by "minSizeForAcceleratorTracklogPvc" will be created.
- During accelerator backups, the size required for the tracklog PVC is calculated based on the size of data available in PVC being backed up.

Kubernetes Operators install and upgrade changes
Validate minimum size provided for the creation of accelerator tracklog PVC

- If the calculated size is less the minimum specified by the "minSizeForAcceleratorTracklogPvc" attribute in the configmap <kops-namespace>-backup-operator-configuration, the size from the configmap is utilized instead.
- If the calculated size is the minimum specified by the "minSizeForAcceleratorTracklogPvc" attribute in the configmap <kops-namespace>-backup-operator-configuration, the size from the configmap is utilized instead.

Upgrade notes for Accelerator Backups

- The Accelerator Backups feature in Kubernetes FileSystem mode is compatible with NetBackup version 10.4.
- Protection plans created before the 10.4 release do not qualify for Accelerator benefits. New protection plans must be created to take advantage of this feature.

  Note: This applies to all Protection Plans, even if older plans have NetBackup storage targets that support Accelerator. NetBackup does not automatically upgrade protection plans.
- All hosts including Primary, Media, Storage, and NBUKOps must be upgraded to version 10.4 to fully utilize the accelerator advantages.

Forced Rescan

- The Accelerator forced rescan feature safeguards against corrupt backup image issues by allowing manual execution of the ForcedRescan command. This functionality provides protection against potential issues such as checksum verification failures.
- NetBackup creates a schedule named 'ForcedRescan' for every protection plan that contains FULL schedule.
- To initiate the backup manually with a forced rescan, run the following command in the command prompt or Linux terminal:
  - bpbackup -i -p <policy_name> -s ForcedRescan
- You can obtain the policy name from web UI from the relevant protection plan.

Forced Rescan

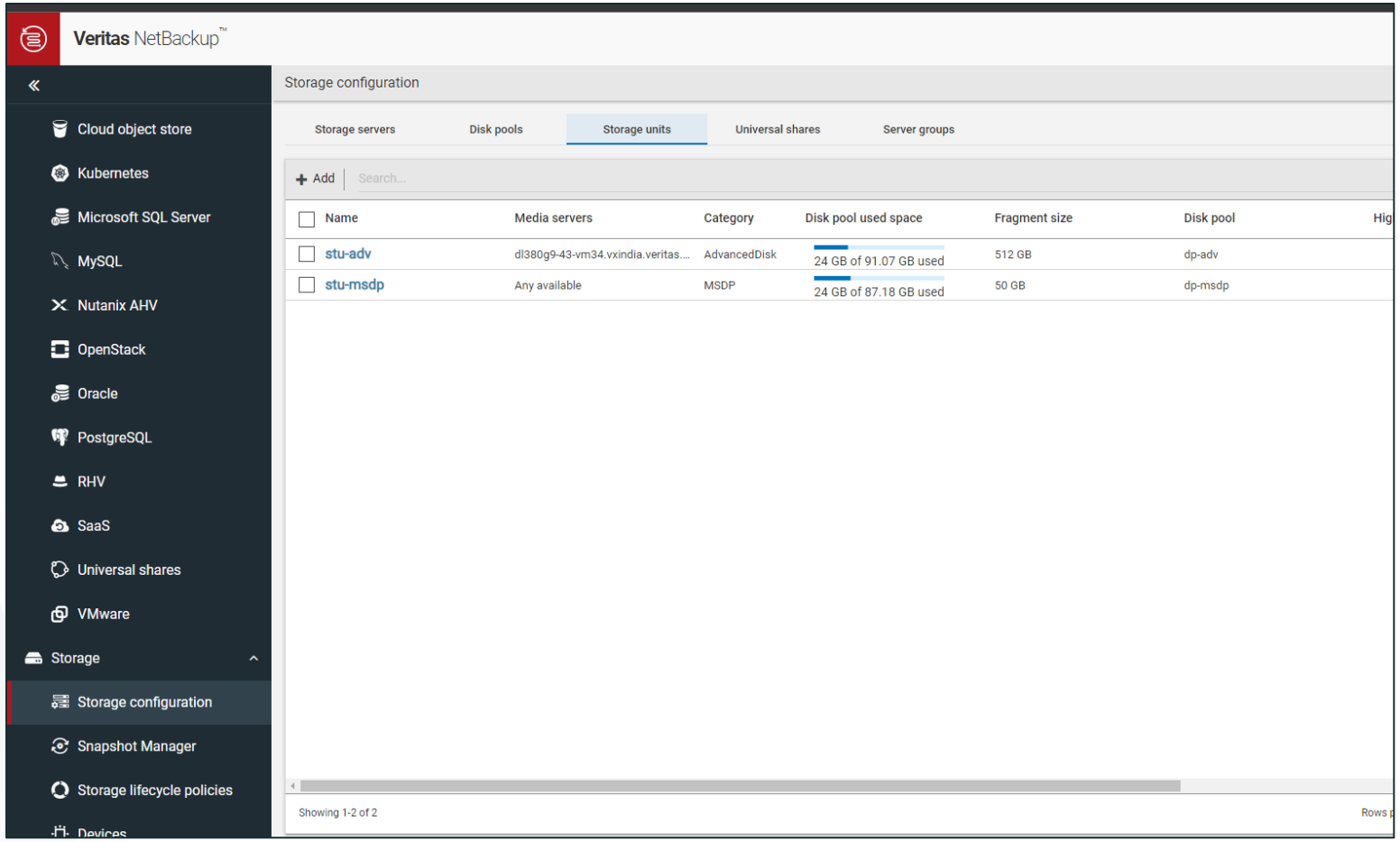
- Example,

  bpbackup -i -p msdp_10mins_FRS+5d990ab5-f791-474f-885a-ae0c30f31c98 -s ForcedRescan

- Likewise, the following API can achieve the same outcome.
  - POST admin/manual-backup
- You can obtain the policy name from the NetBackup web UI from the relevant protection plan.

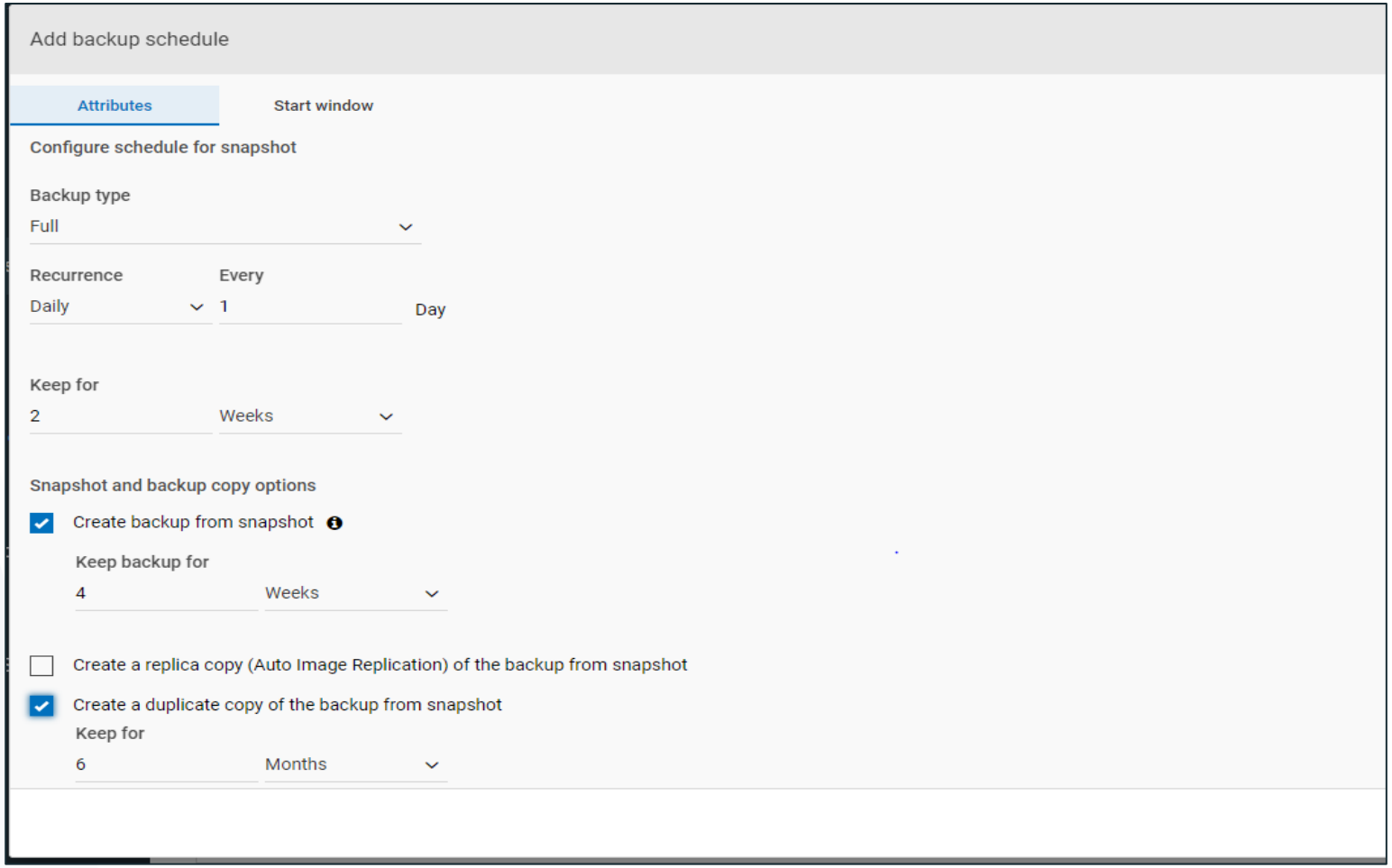
Configure Duplication

Configure Duplication



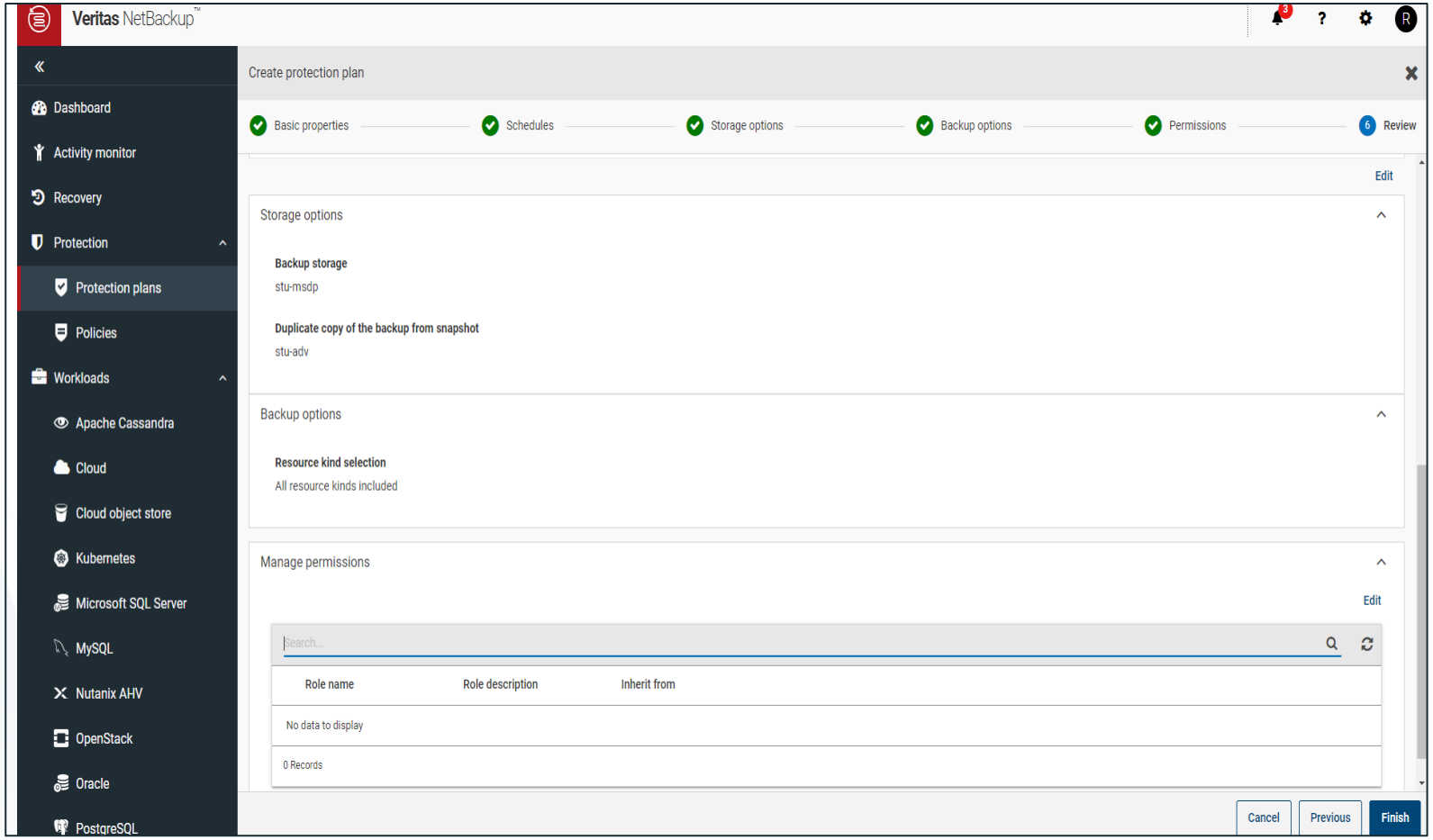
1. Create Storage Unit to keep duplicate image copies.

Configure Duplication



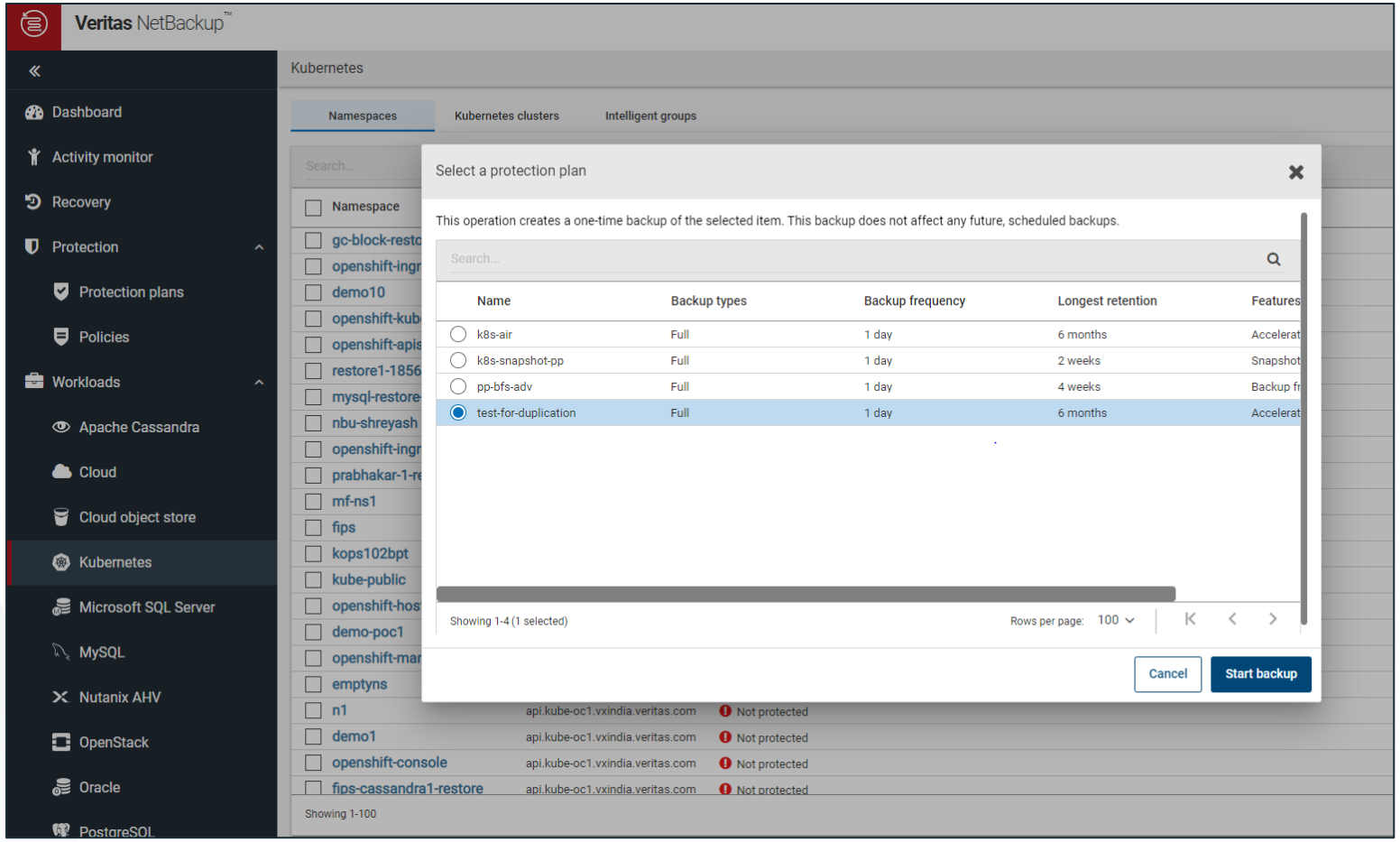
2. Create Protection plan with duplication.
3. Enter retention period of each copies.

Configure Duplication



1. Enter storage for each copies during protection plan creation step.
2. Click **Finish**.

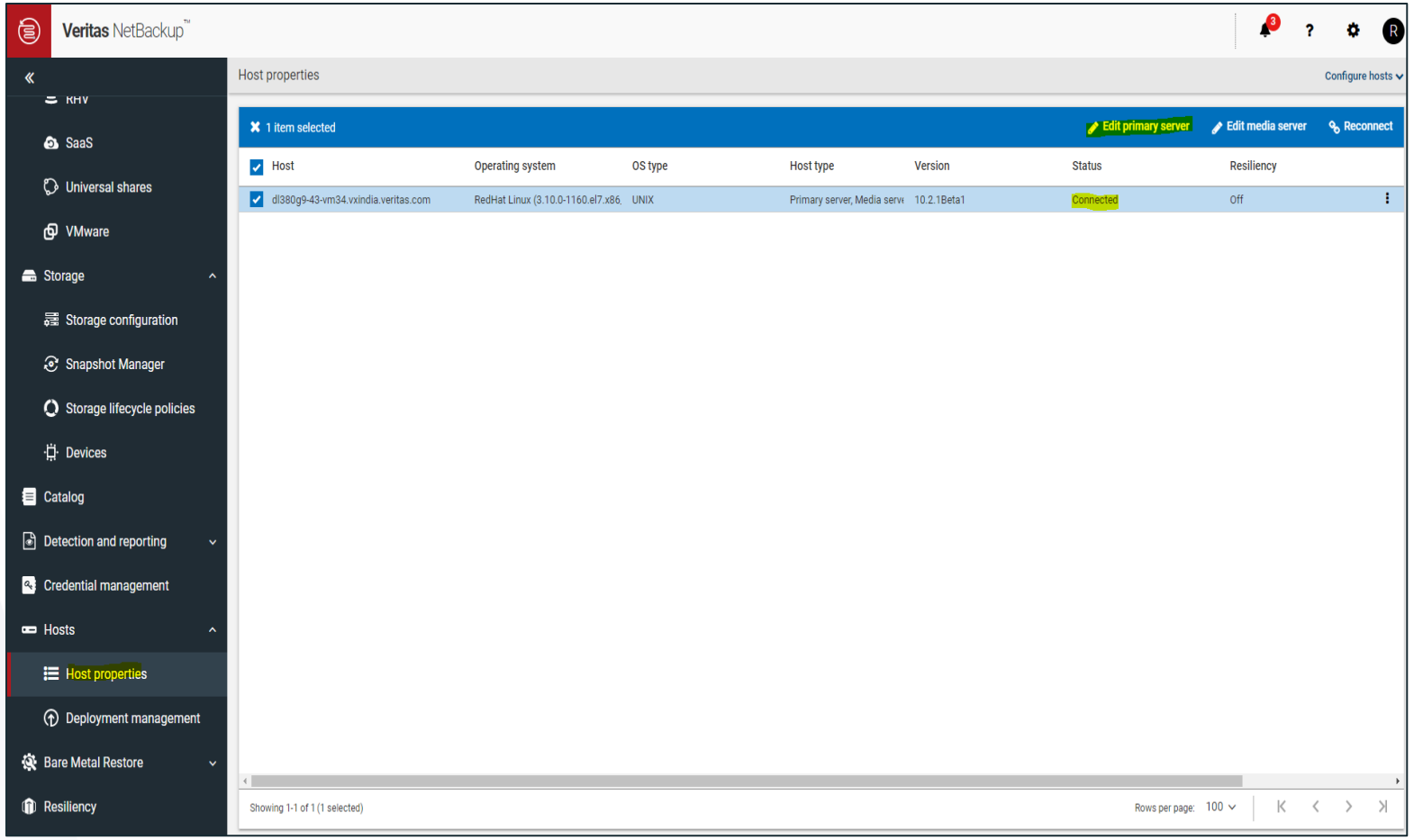
Configure Duplication



3. Start backup of Kubernetes asset using the protection plan.
4. Duplication jobs gets triggered only after backup completed.
5. Duplication happen using the backup from snapshot copy

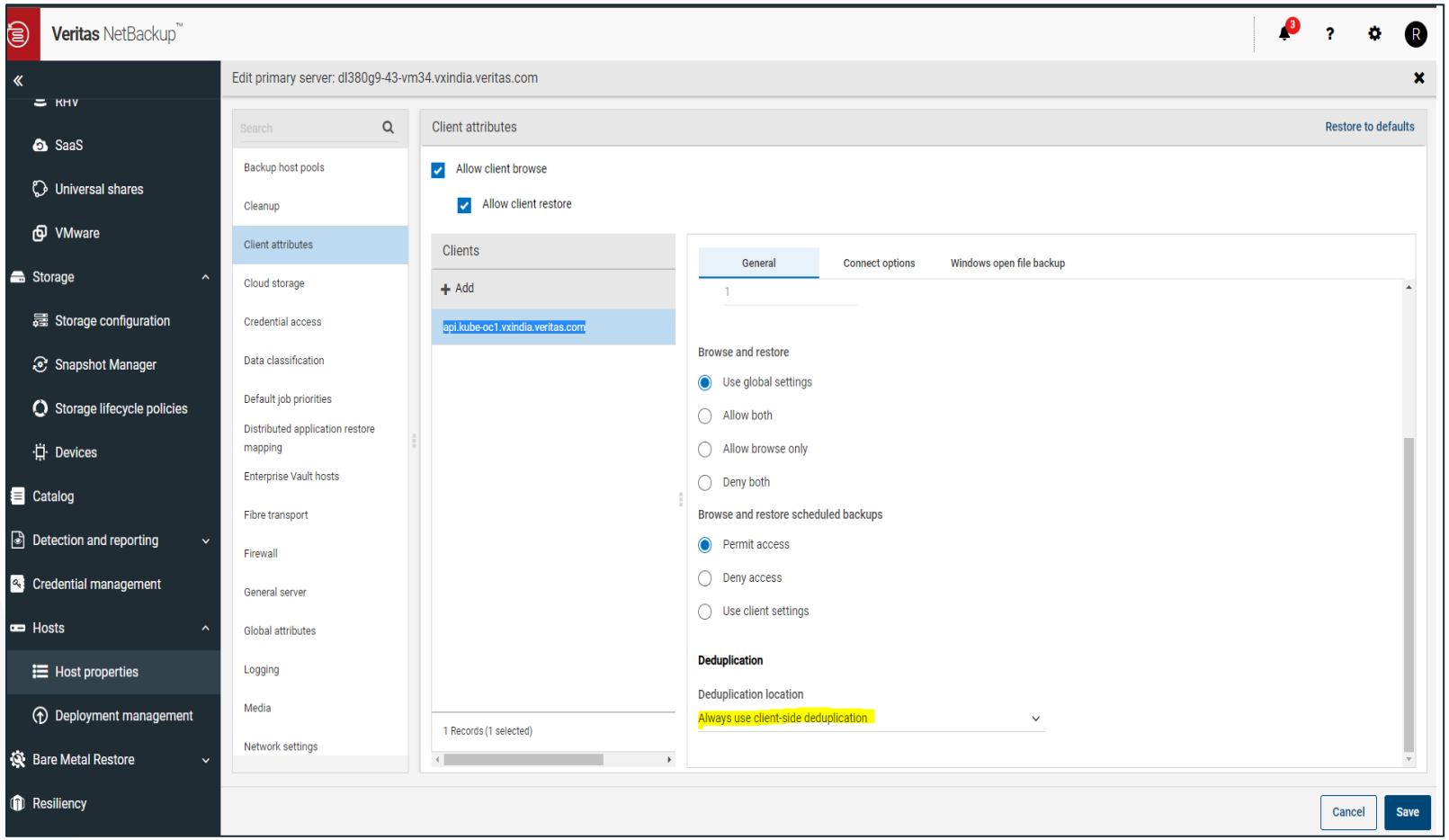
Configure Client-side Deduplication

Configure Client-side Deduplication



Click **Host Properties** to connect to the primary server.

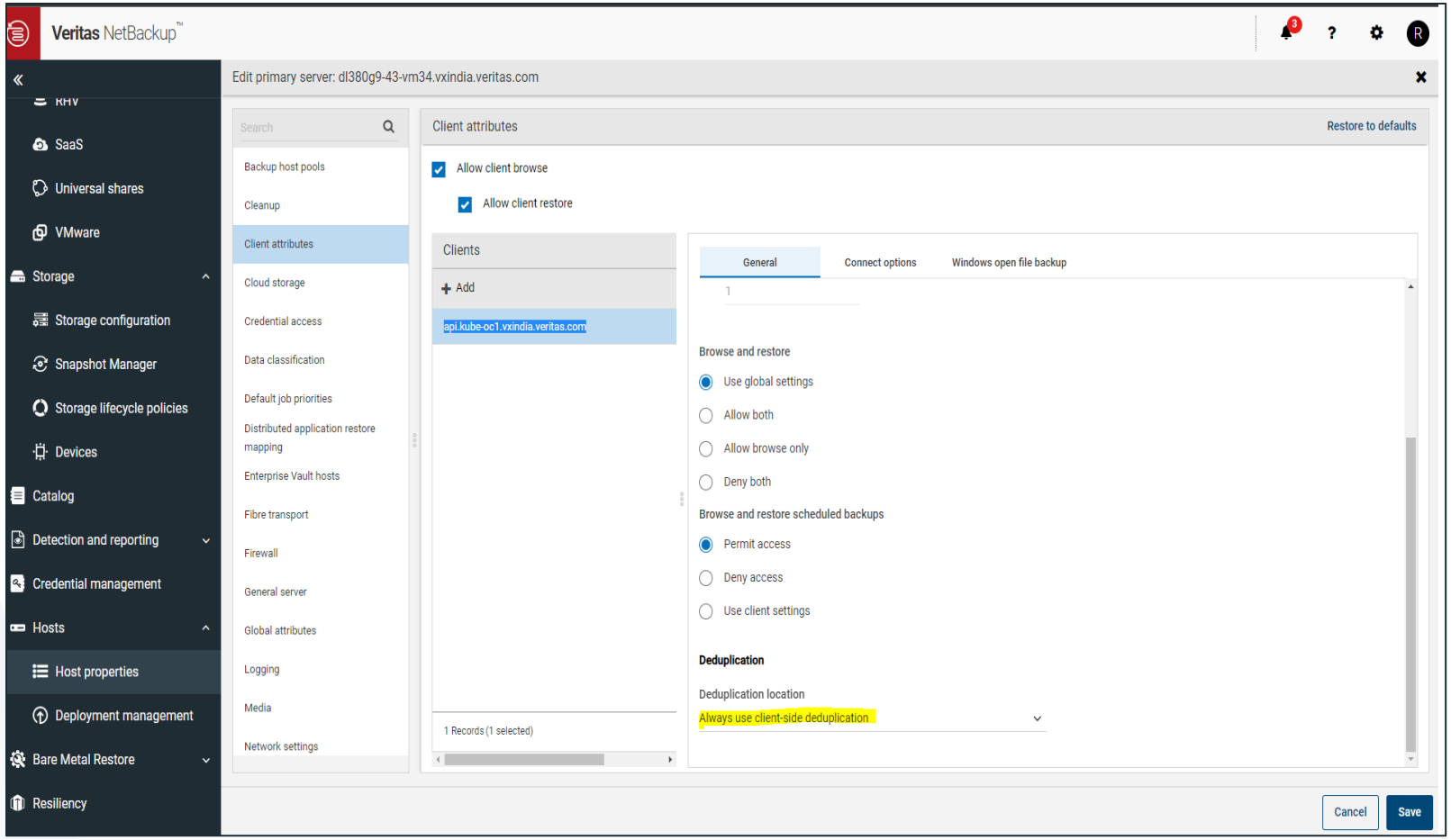
Configure Client-side Deduplication



- Edit Primary server **Client attributes**, add Kubernetes cluster name under **Clients** tab.
- You can retrieve the Kubernetes cluster name running the following command on the cluster.

  kubectl cluster-info
- Cluster name can also be found on the NetBackup web UI.
- Workloads > **Kubernetes** > **Kubernetes clusters** > **Name** column.

Configure Client-side Deduplication



- Select **Always use client side deduplication** from deduplication location dropdown list.
- Click **Save**.

Note: Storage unit configured in Protection Plan for Backup from snapshot should be of MSDP type.

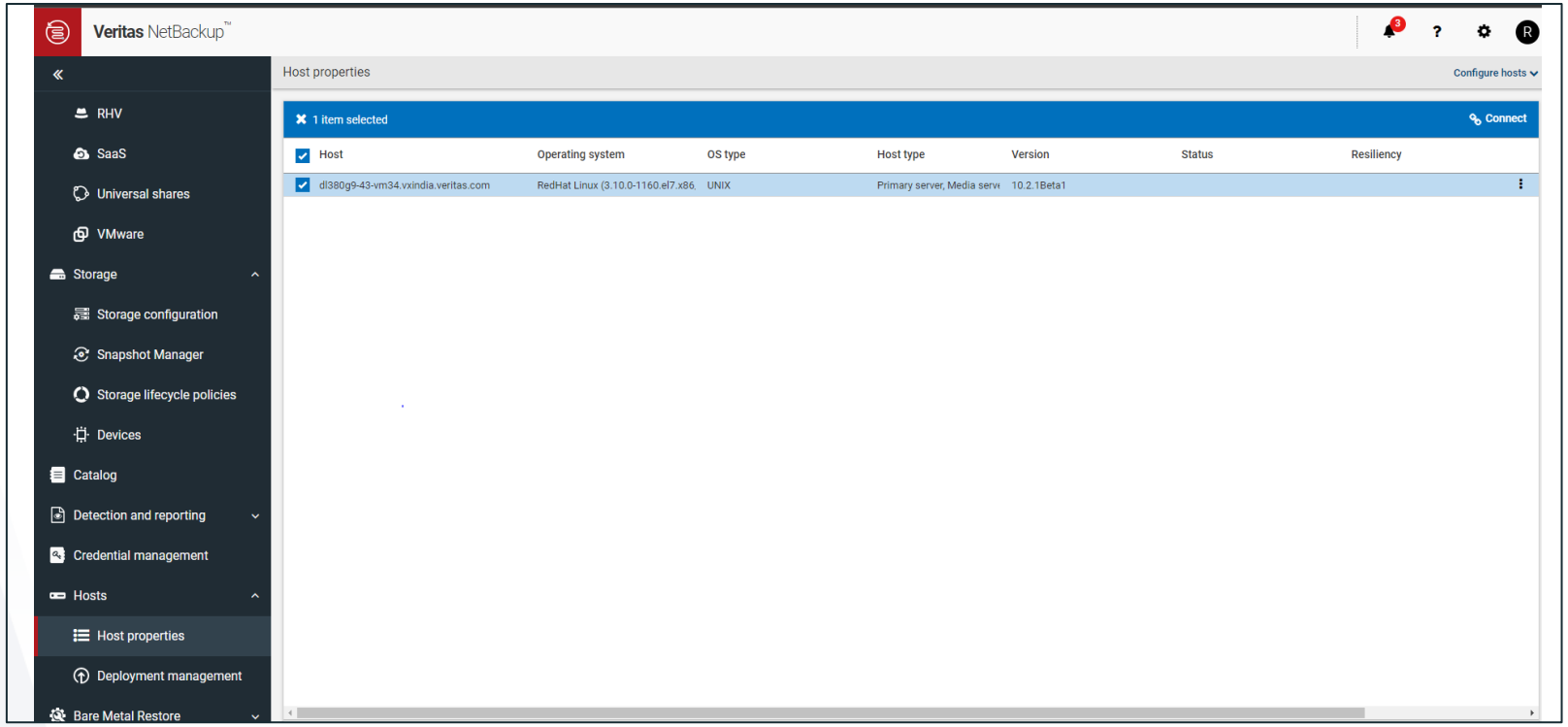
Configure Auto Image Replication (AIR)

Configure Auto Image Replication

Prerequisite for AIR:

- Source and target NetBackup primary servers must have MSDP storage configured as storage unit.
- Source and target primary servers must be reachable to each other.

Configure Auto Image Replication

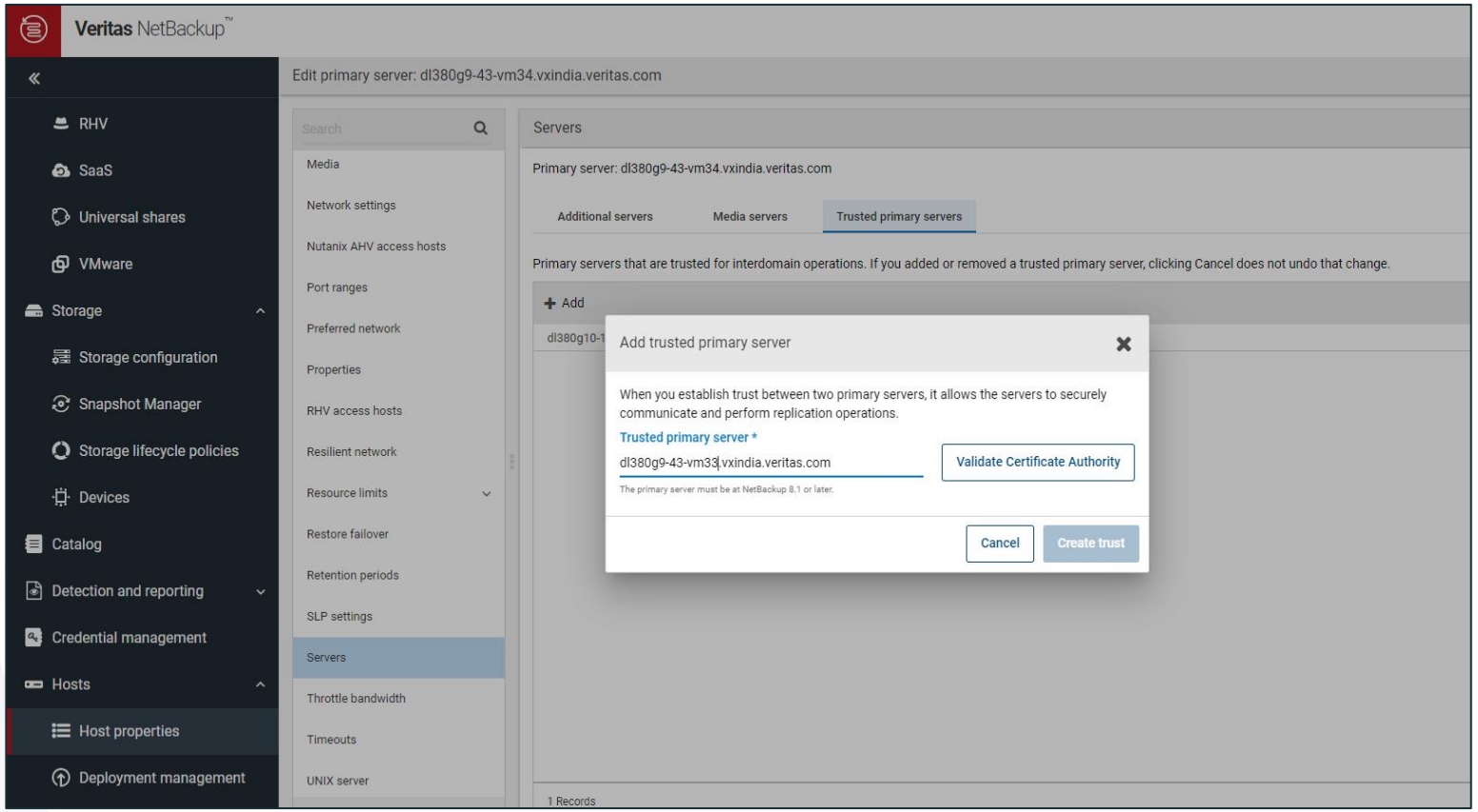


Configuration on the source primary server.

Step 1

- Add target primary server entry under trusted servers.
- In the NetBackup web UI navigate to the Host properties
  - Select primary server in host entries, and connect.

Configure Auto Image Replication

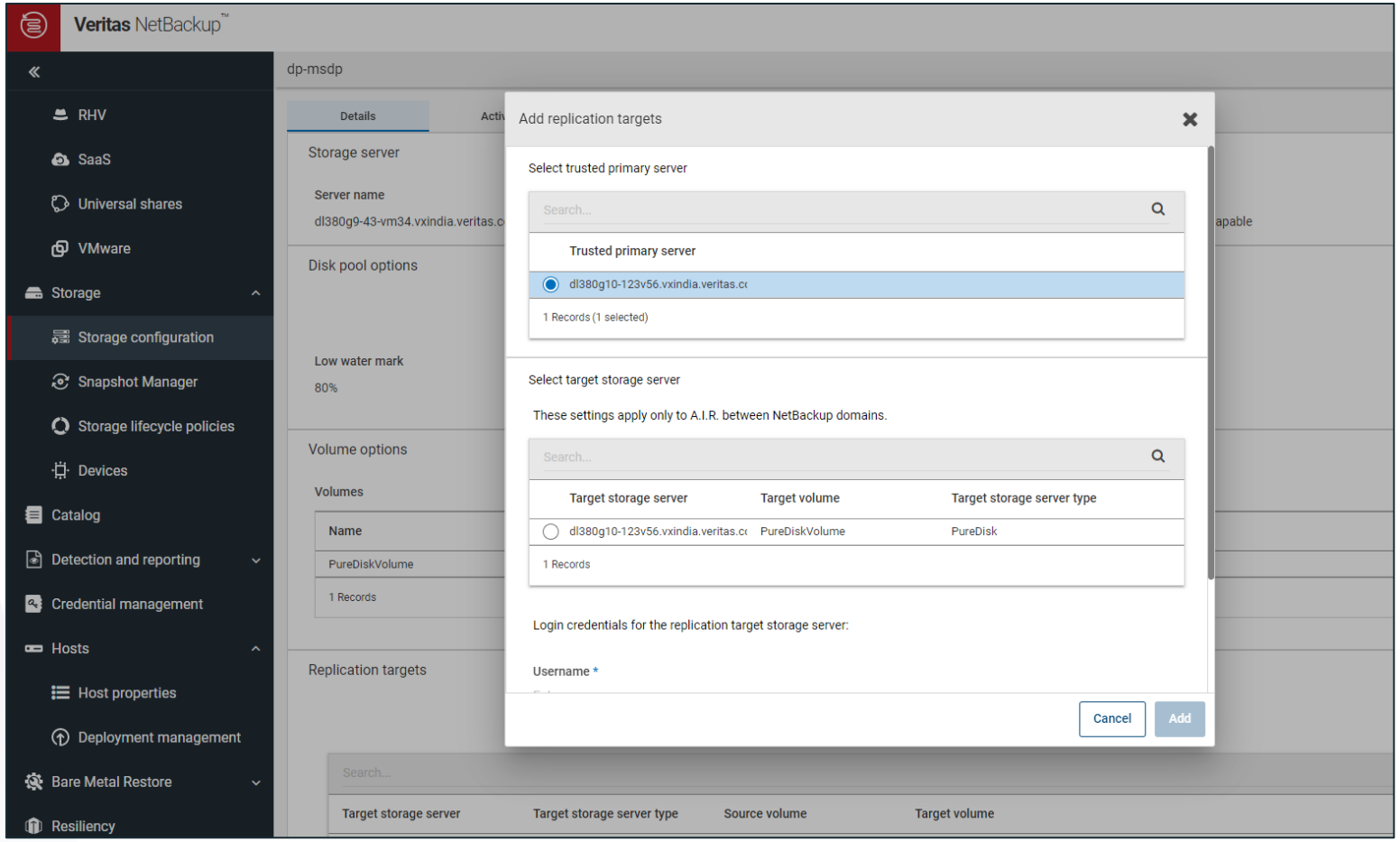


Configuration on source primary server.

Step 2

- Navigate to **Servers** in **Edit Primary server** page
- Under **Trusted primary server** tab **Add** entry of target primary server.
- Click **Validate Certificate Authority**
- Enter **Target primary server credentials** and click **Create Trust.**

Configure Auto Image Replication

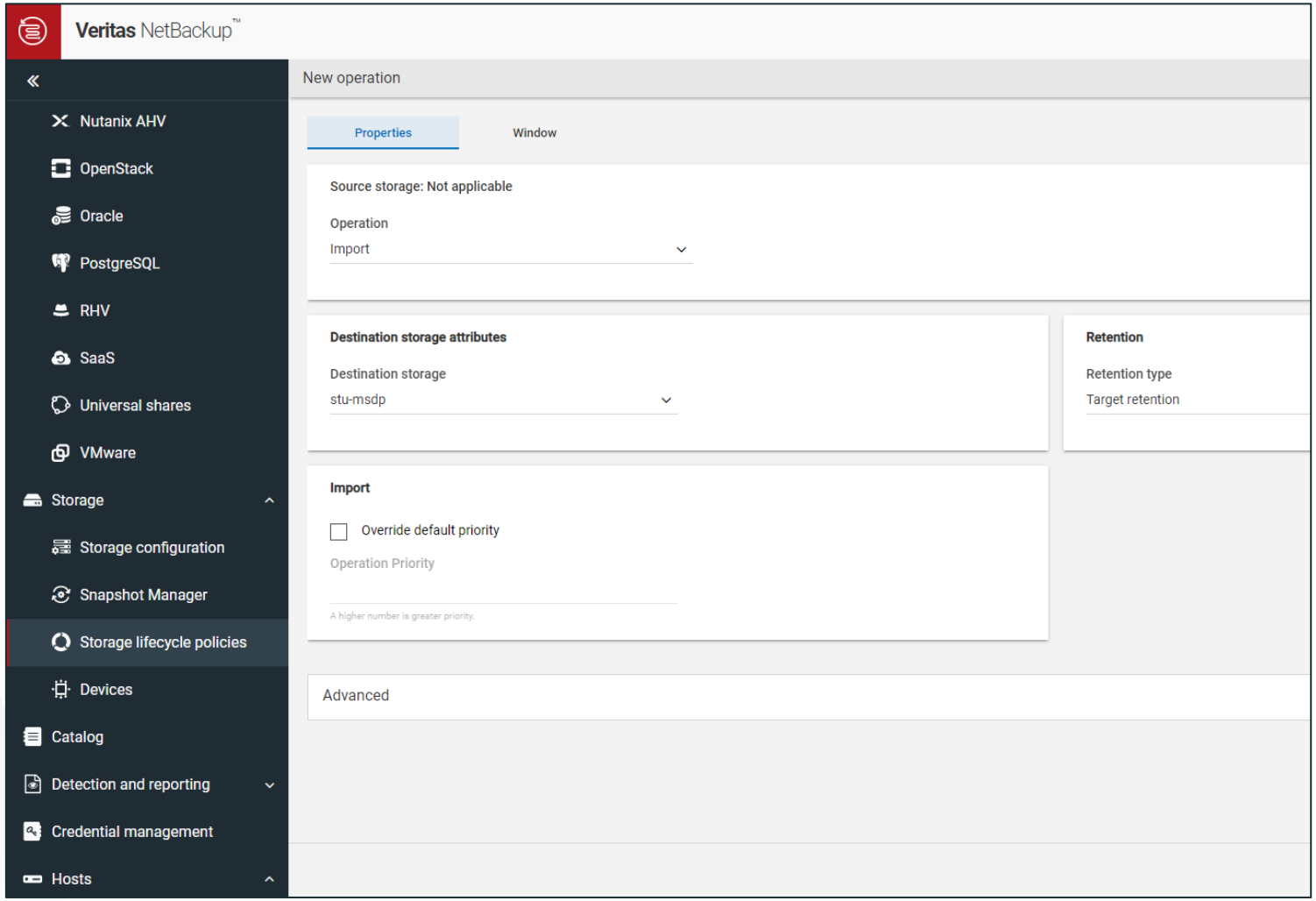


Configuration on source primary server.

Step 3

- Navigate to **Storage** > **Storage Configuration** in the NetBackup web UI
- Select **Disk-pools** tab.
- Select disk-pool having MSDP category
- Add entry for Replication targets.
  - Select trusted primary server
  - Enter the primary server credentials.
  - Click **Add**.

Configure Auto Image Replication



Configuration on Target primary server.

Step 1

Follow the Configuration of source primary server Step 1 and 2 to add trusted primary servers.

Step 2

- Create Storage lifecycle policies
- Navigate to Storage > Storage lifecycle policies in Target machine NetBackup web UI.
- Create SLP with operation import.
- Select Destination storage of MSDP type.
- Select Retention type.
- Click Create.
- Same SLP name is visible during configuration of source primary server step 5

Configure Auto Image Replication

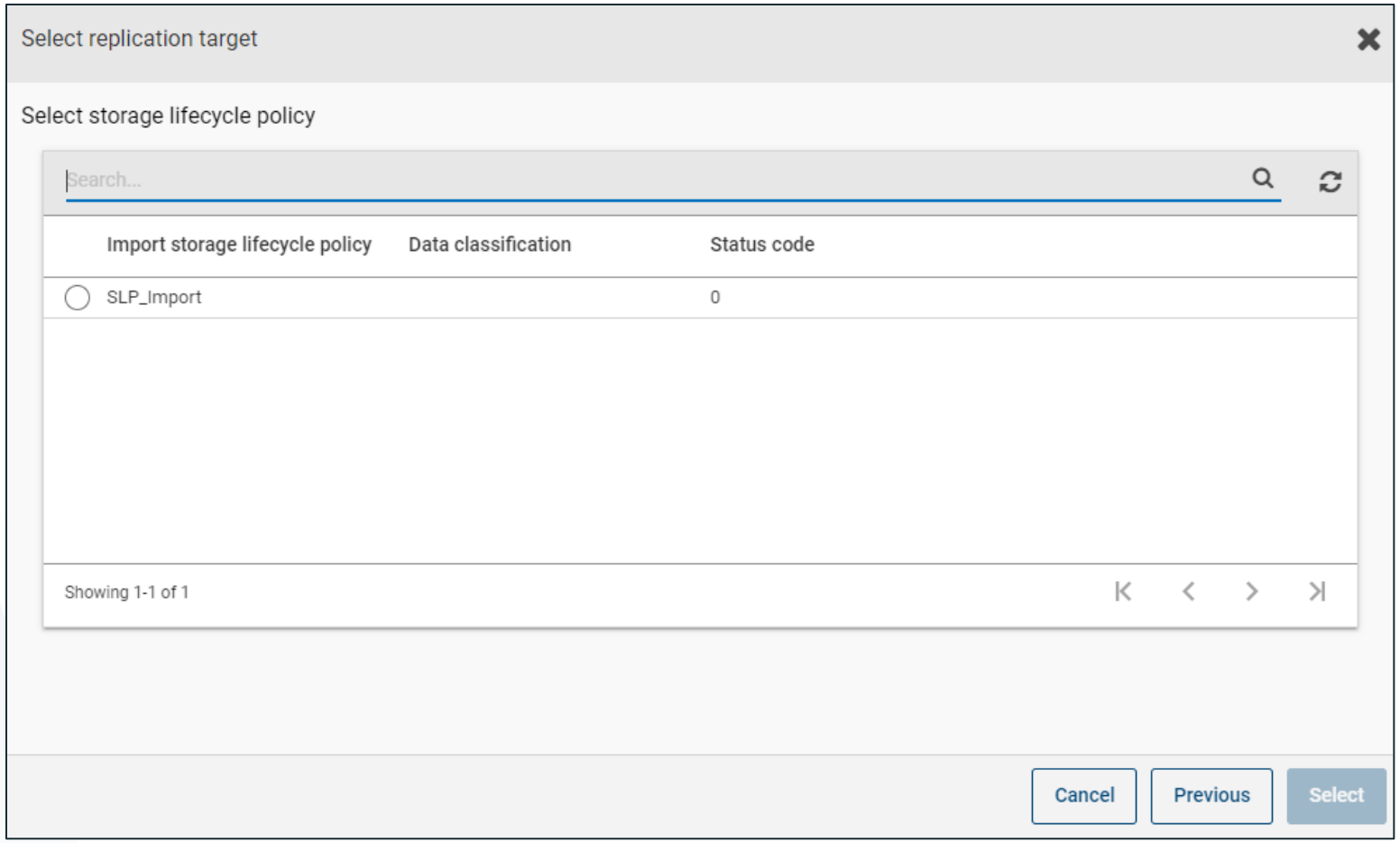
Configuration on source primary server.

Step 4

- Create protection plan for Kubernetes workload, with **Create a replica copy (Auto Image Replication) of the backup from snapshot** option selected.
- Add Schedules of protection plan.

VERITAS

Configure Auto Image Replication



Configuration on source primary server.

Step 5

- In **Create protection plan storage** options, select storage unit for backup from snapshot as local MSDP storage unit.
- For Replica copy of the backup from snapshot
  - Select target primary server.
  - Click **Next**.
  - Select storage lifecycle policy, this is created in “Configuration of target primary server step 2”
- Click **Next.**
- Select **Resource for Protection,** and click **Next.**
- Select the roles you want to have access to this protection plan.
- Click **Finish.**

FIPS Enablement

Configure FIPS enablement

From NetBackup 10.2.1(Sequoia) release, FIPS support has been introduced for NetBackup Kubernetes workloads in Redhat-based deployments. This requires that all components involved in the Kubernetes workload, NetBackup Kubernetes Operator and NetBackup Data Mover must operate in FIPS mode. Meeting specific requirements across these components is necessary to achieve FIPS support.

VERITAS

Configure FIPS enablement

System Requirement

- NetBackup Primary and Media server
  - Both Primary and Media should be deployed on NetBackup 10.2.1 with underlying RHEL-8 system which is enabled with FIPS.
  - RHEL OS version should be greater than REHL8.
    - You can check version of Redhat machine.
      - cat /etc/Redhat-release
    - You can check if underlying system has FIPS is enabled using below command
      - fips-mode-setup --check
      - For more information, you can check man page entry for command fips-mode-setup

- Kubernetes cluster
  - Kubernetes cluster should be deployed with FIPS enabled mode.
  - The process to deploy Kubernetes cluster in FIPS mode in vendor dependent.
  - For example, deploying Openshift with FIPS Enabled.

VERITAS

Configure FIPS enablement

Configuration

- NetBackup Primary and NetBackup Media
  - Enabling NetBackup process to run in FIPS mode:
    - Update <Netbackup-Installation-Path>/netbackup/bp.conf with below key
    - NB_FIPS_MODE = ENABLE

- NetBackup Kubernetes Operator
  - User can follow any one of the below steps to enable FIPS mode.
    - Update the value of parameter **fipsMode to ENABLE** in values.yaml file from the Helm Chart.

      OR
    - Update the value of parameter **NB_FIPS_MODE to ENABLE** in backup-operator.

Note: Customer must ensure that the system on which Kubernetes workload is running are FIPS compliant.

VERITAS

OpenShift Virtualization support

OpenShift Virtualization support

NetBackup 10.4.1 release and later, provides backup and restore support for namespaces having one or more multiple virtual machines running on Kubernetes clusters. Users can create Kubernetes Protection plan to protect namespace which has virtual machines and its related resources.

Note: To ensure the successful backup and restore of namespaces containing virtual machines, it is essential to have the necessary components required for virtualization is installed and configured properly on Kubernetes cluster.

OpenShift Virtualization support

Intelligent group :

Users can create intelligent groups by filtering namespaces based on the following resource kinds.

- Virtual Machines
- Persistent Volumes
- Persistent Volume Claims

VERITAS

Troubleshooting OpenShift Virtualization

Restore of VM having static MAC defined will fail on the same cluster if MAC is already assigned to any virtual machine.

Error message: Failed to allocate requested mac.

Recommended action: When a user performs virtual machine restore operation in same cluster with alternate namespace then there may be chance existing virtual machine MAC is assigned to source virtual machine, user needs to make sure MAC should not be assigned to any virtual machine.

VERITAS

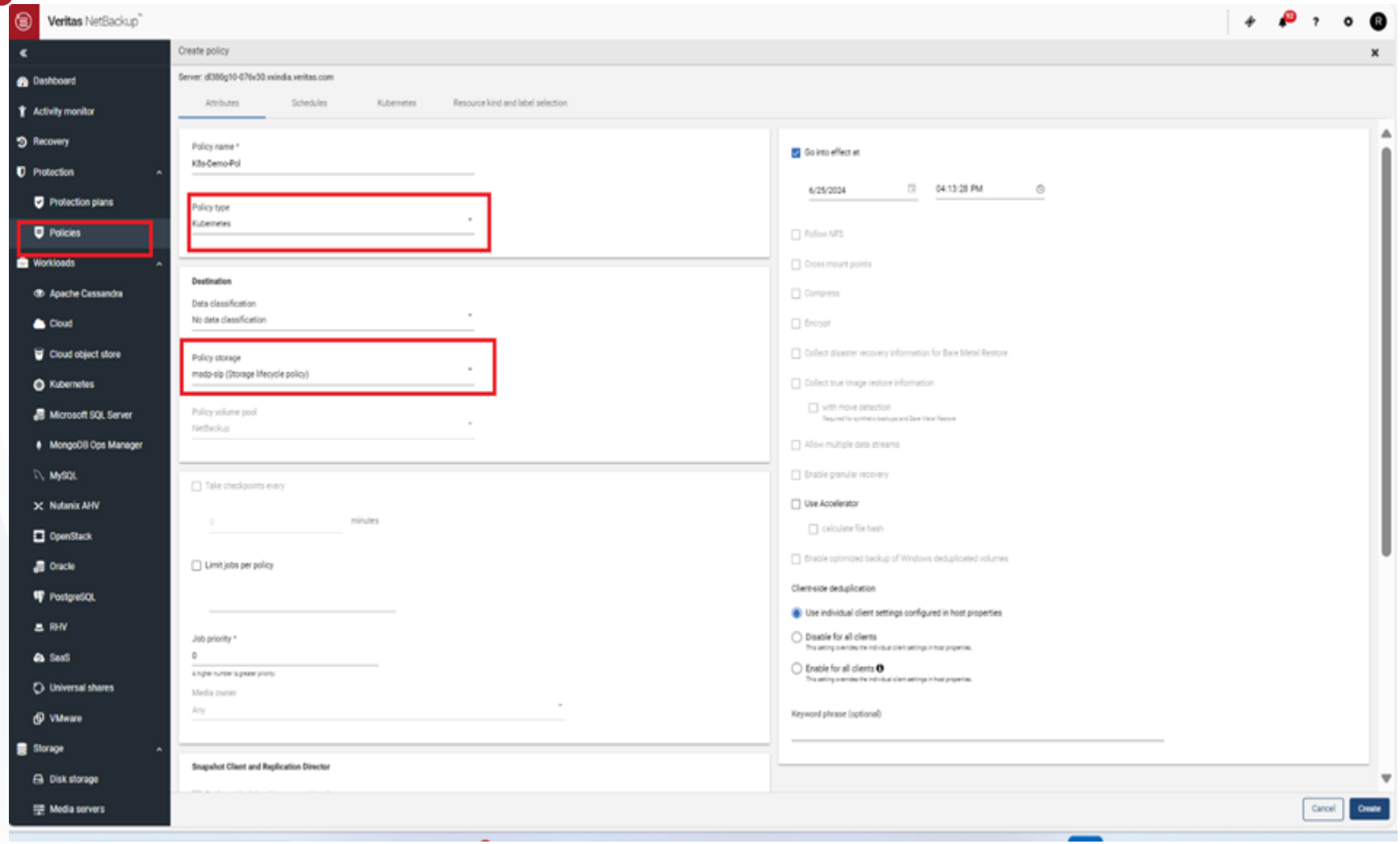
Kubernetes Policy support

Kubernetes Policy support

- Starting with the NetBackup 10.5 release, backup and restore capabilities have been extended to Kubernetes resources via NetBackup policies.
- Furthermore, this includes:
  - Support for tape as a media option for Kubernetes backup and restore, covering all types of PVCs, including File Mode, Block Mode, and File Mode.
  - Support for Expire after copy and other SLP operations for Kubernetes.

VERITAS

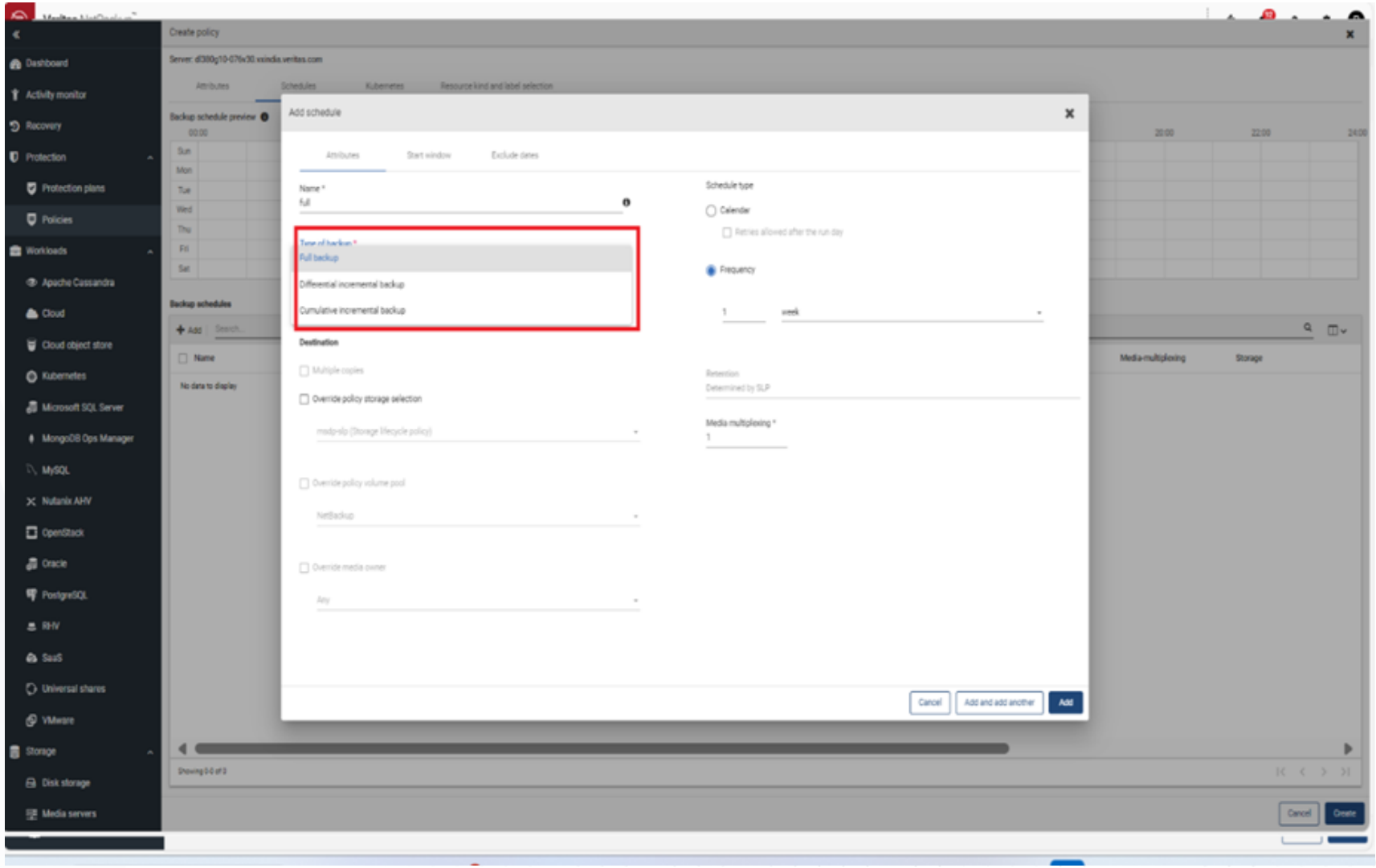
Create policy



Attributes

1. Provide policy name.
2. Select Policy type as Kubernetes from the list.
3. Specify Policy storage as SLP or storage unit.
4. Click **Next**.

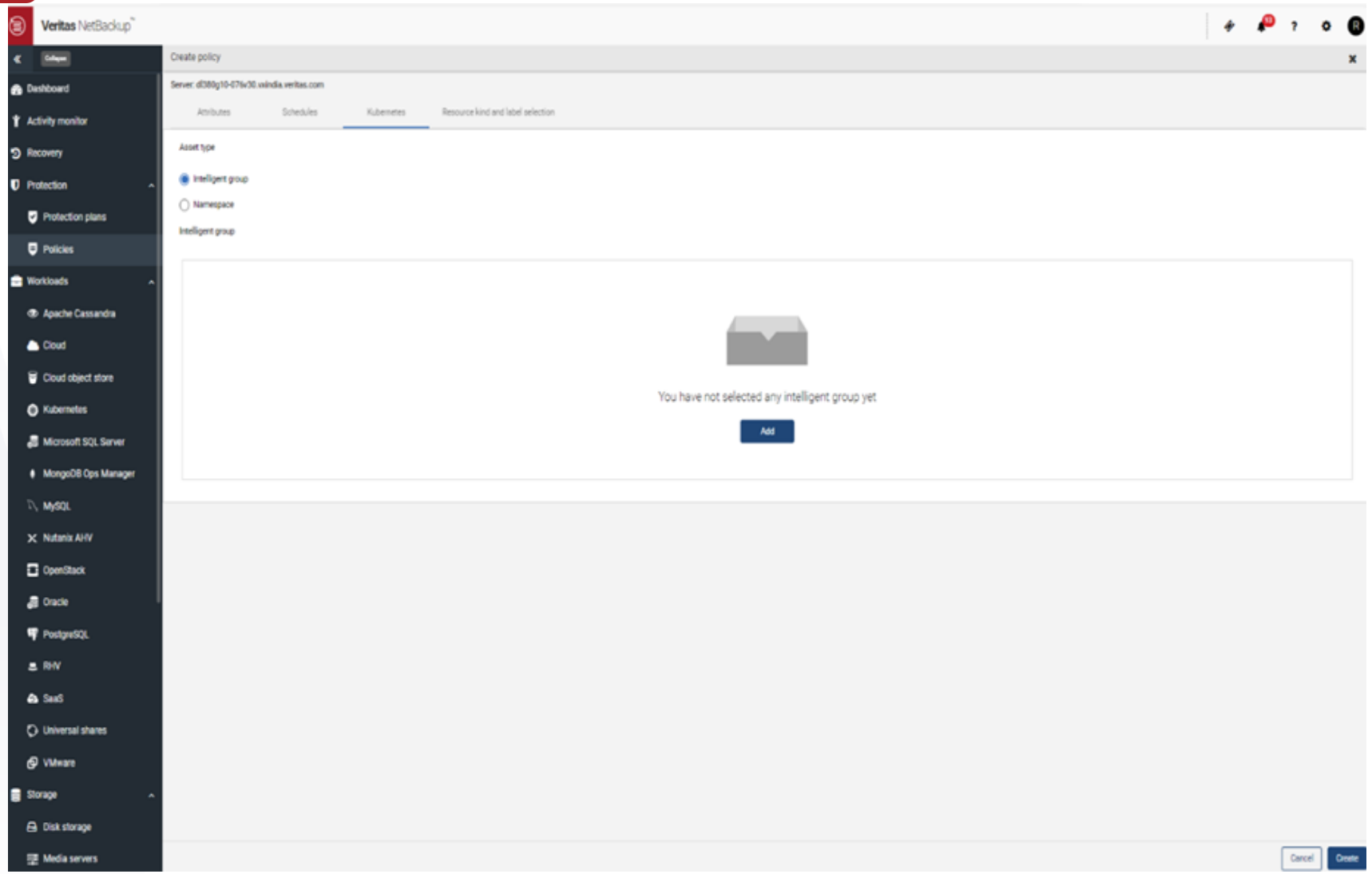
Create policy



Schedule

5. Provide schedule name.
6. Select Schedule type from the list.
7. Click **Add.**

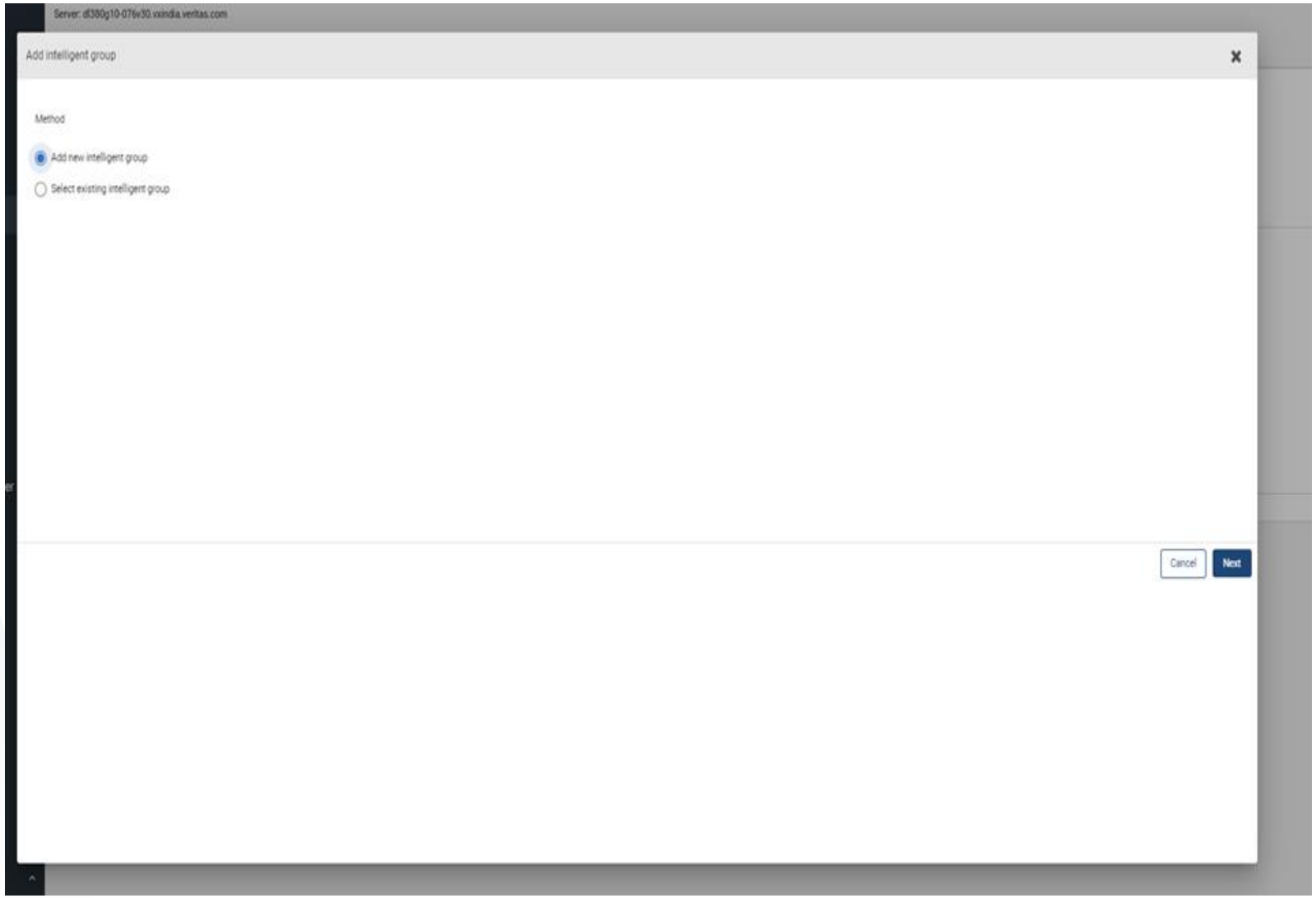
Create policy



Kubernetes

8. Provide Backup Selection
9. Select asset type as an Intelligent group or Namespace.
10. Click **Add.**

Create policy

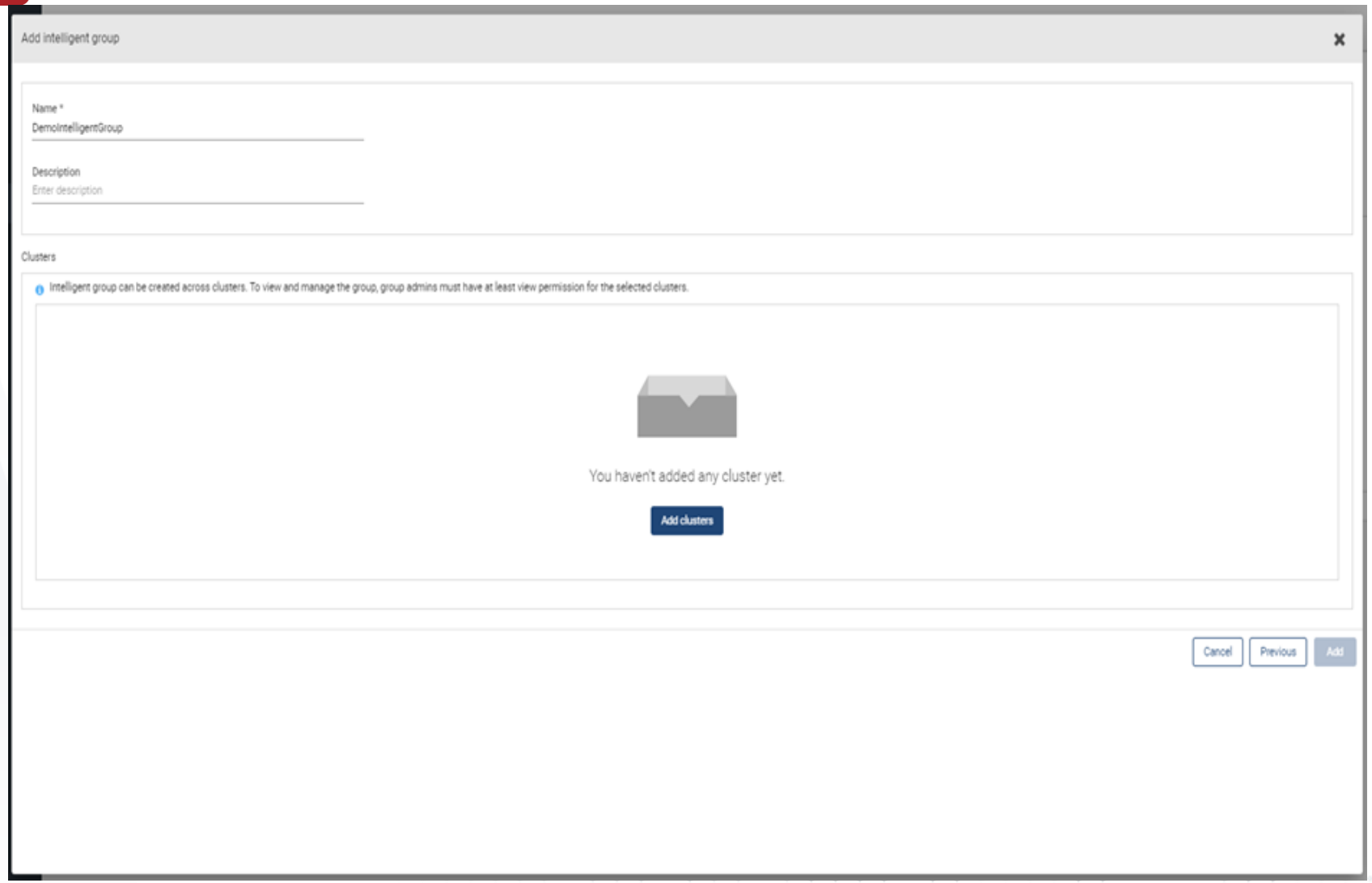


Kubernetes

Add Intelligent group

1. Select a new or existing intelligent group that you want to protect.

Create policy

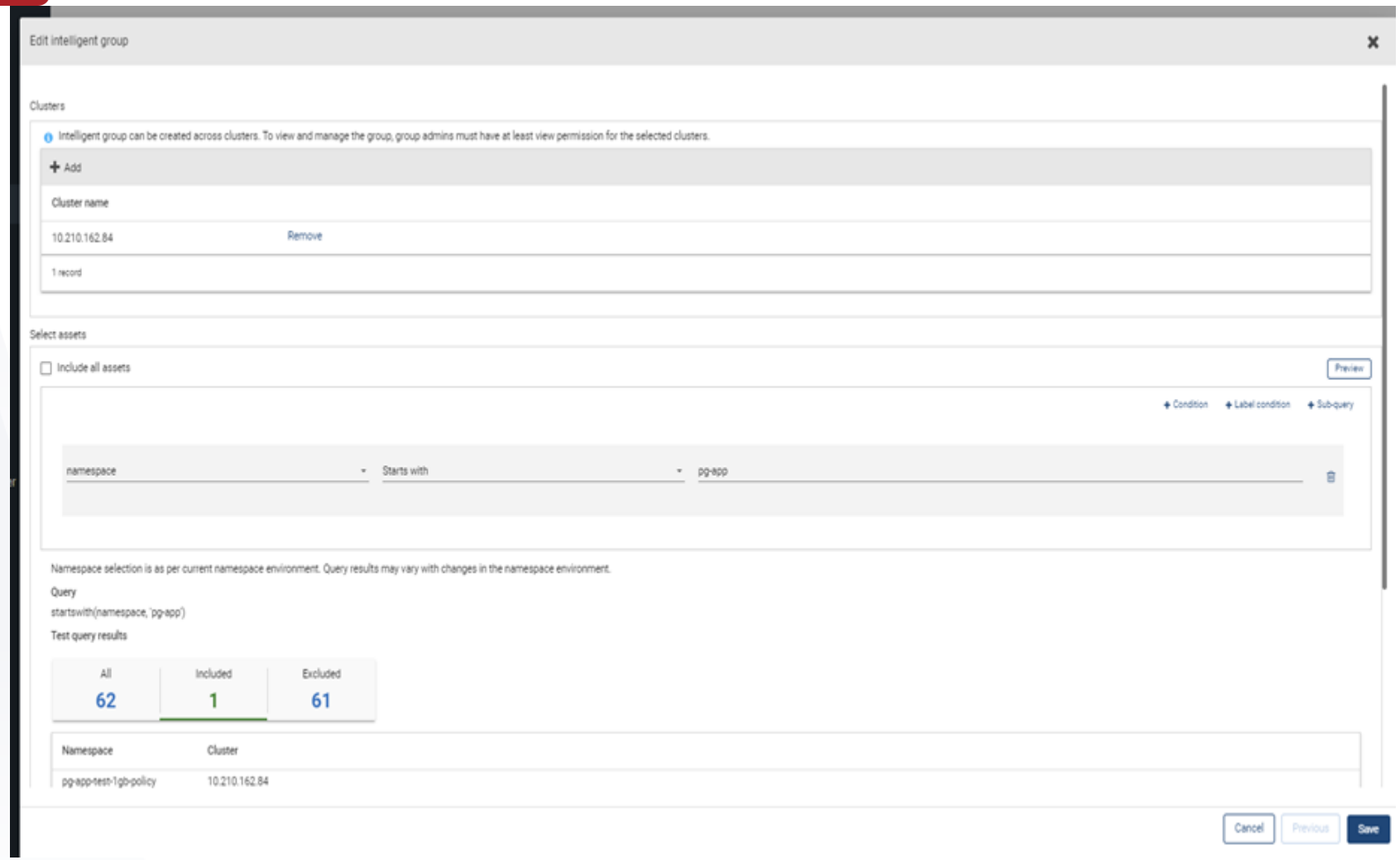


Kubernetes

Add Intelligent Group

2. Enter the Intelligent group name and description.
3. Click **Add Clusters.**

Create policy



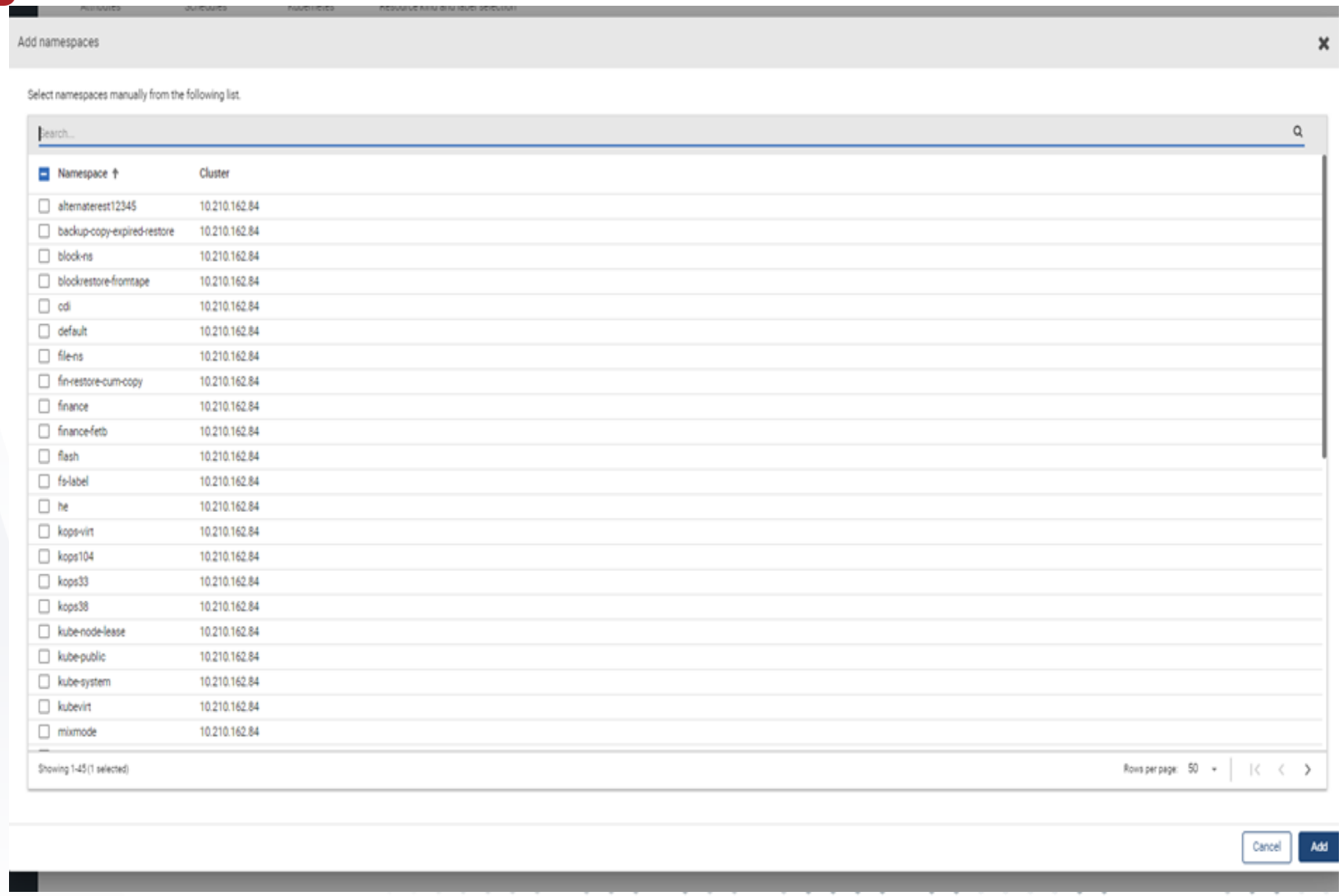
Kubernetes

Add Intelligent Group

4. Specify the query for an intelligent group.
5. Click **Save.**

VERITAS

Create policy

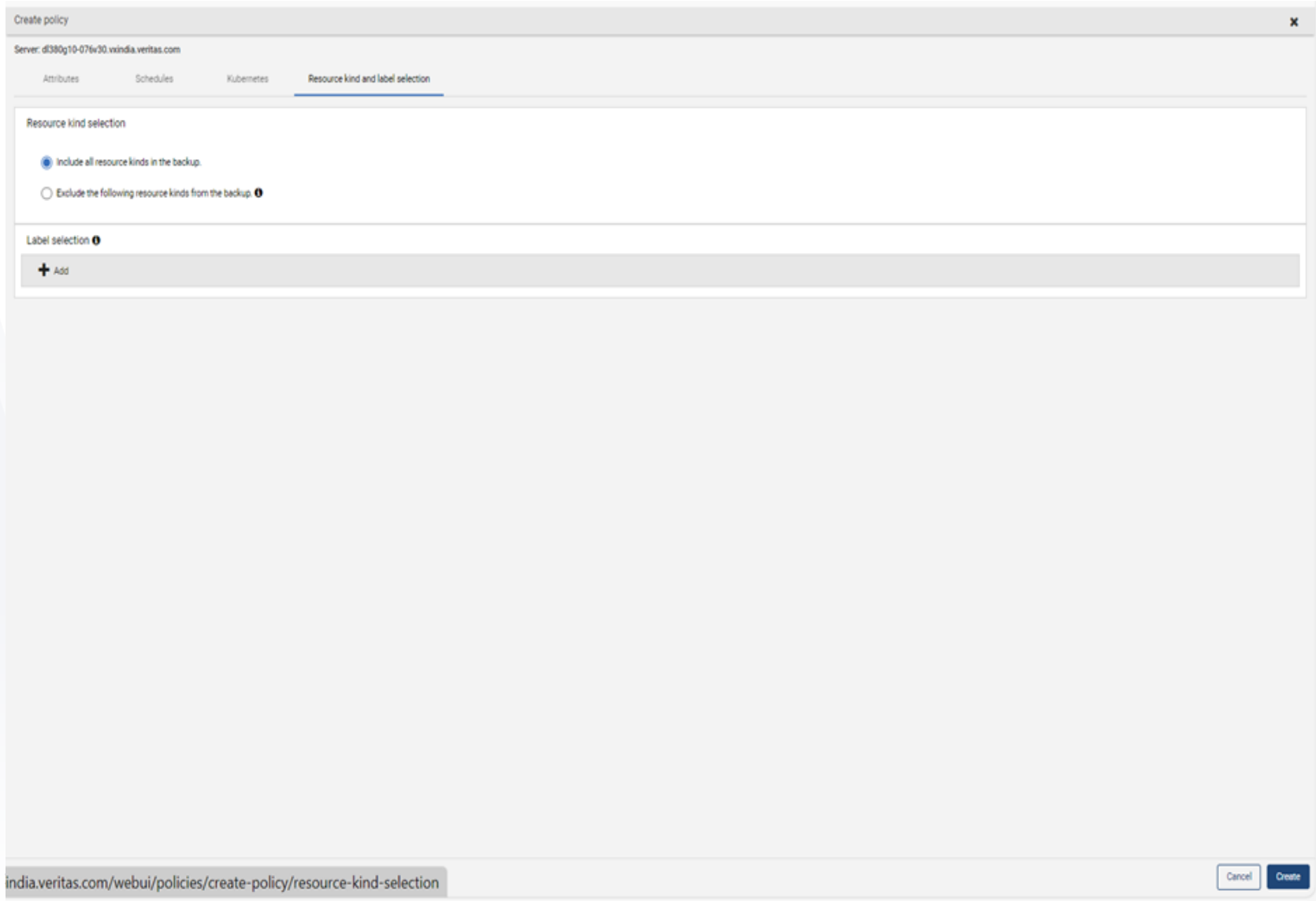


Kubernetes

Add Namespace

6. Select the namespaces from the list you want to protect.

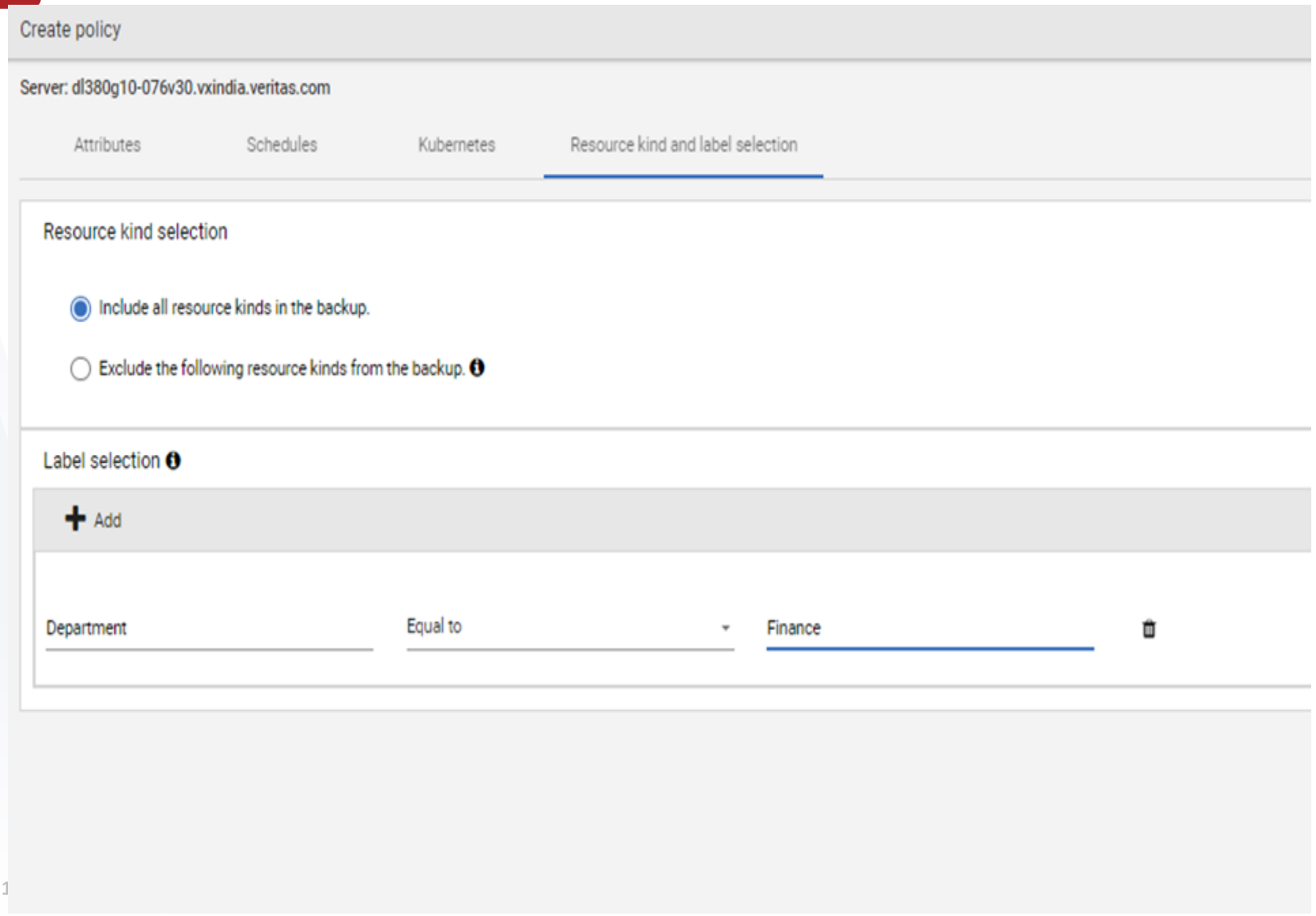
Create policy



Resource Kind and label selection

7. Select the include **all resource kind in the backup** option to include all resource kinds in the backup.
8. Select **exclude the following resource kinds from the backup** option to include all resource kinds in the backup.

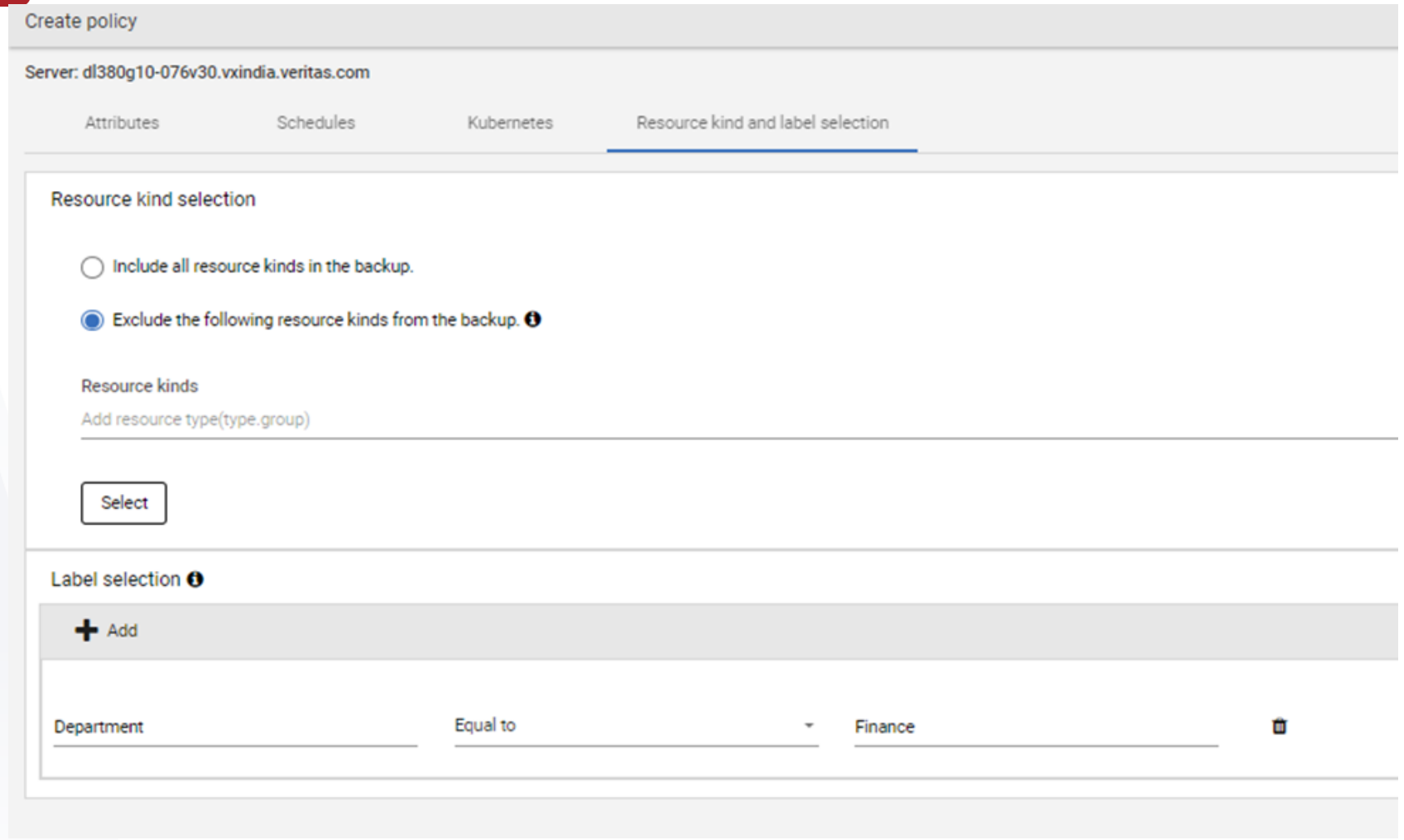
Create Policy



Resource Kind and label selection

- If you have selected the **Include all resource kind in the backup** option to include all resource kinds in the backup.
- Add a query for a label in **Label selection**.

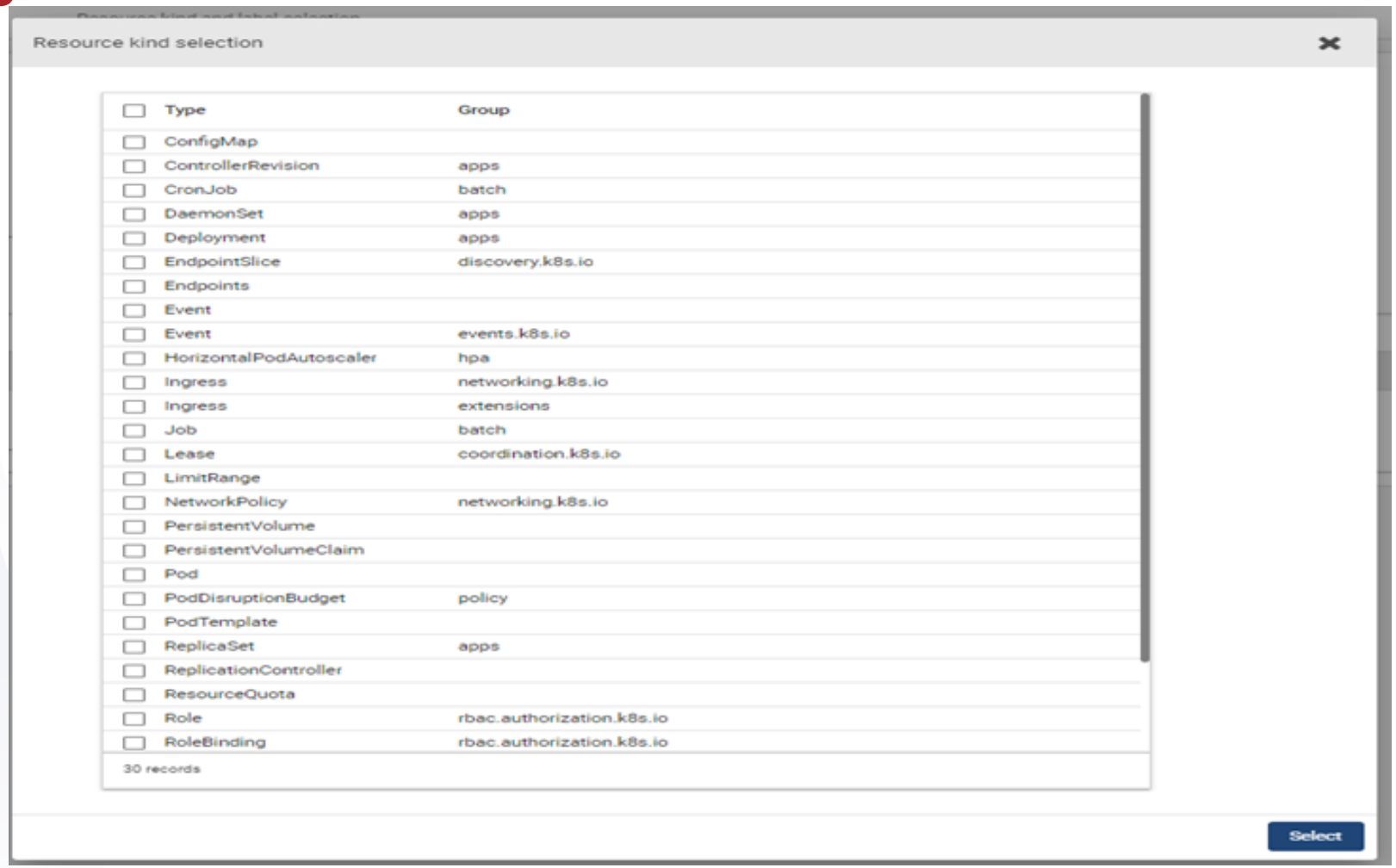
Create policy



Resource Kind and label selection

- Select the **exclude the following resource kinds from the backup** option.
- Enter the **Resource kinds** or **Select**.

Create policy



Resource Kind and label selection

- Select the **exclude the following resource kinds from the backup** option.
- Select the **Resource kinds** you want to exclude from backup from the list.

9. Click **Select.**

Troubleshooting

Troubleshooting

Troubleshooting for known issues

- If the Kubernetes add cluster operation fails in NetBackup with an error message **Failed to validate cluster <cluster-name>**, then the failure could be due to the following reasons:
  - User might have created credentials with incomplete ca.crt value in NetBackup.
  - The ca.crt value was properly copied but the cluster's service account and API endpoint have a different Certifying Authority (CA). User can check CA by extracting CA certificate from the cluster API endpoint.
  - Verify if <kops-namespace>-nb-config-deploy-secret has proper values for k8scacert, k8stoken with the right indentation.

VERITAS

Troubleshooting

Troubleshooting for known issues

- Solution: Refer to the tech note x509 certificate signed by unknown authority error during discovery and backupservercert configuration of NetBackup Kubernetes setup
- In customer environments with restricted access to external repositories, the deployment of NetBackup Kubernetes operators may fail to pull the 'kube-rbac-proxy' image from these repositories.

