Instale Trident Protect

Trident

NetApp
November 14, 2025

Tabla de contenidos

Instale Trident Protect

Los requisitos de Trident protegen

Comience verificando la preparación de su entorno operativo, clústeres de aplicaciones, aplicaciones y licencias. Asegúrese de que su entorno cumpla los siguientes requisitos para poner en marcha y operar Trident Protect.

La compatibilidad de clústeres de Kubernetes de Trident protege

Trident Protect es compatible con una amplia gama de ofertas de Kubernetes totalmente gestionadas y autogestionadas, entre las que se incluyen:

- Amazon Elastic Kubernetes Service (EKS)
- Google Kubernetes Engine (GKE)
- Microsoft Azure Kubernetes Service (AKS)
- Red Hat OpenShift
- SUSE Rancher
- Cartera de VMware Tanzania
- Subida de Kubernetes

- Las copias de seguridad de Trident Protect solo se admiten en nodos de cómputo de Linux. Los nodos de cómputo de Windows no son compatibles con operaciones de copia de seguridad.
- Asegúrese de que el clúster en el que instala Trident Protect está configurado con un controlador de instantáneas en ejecución y los CRD relacionados. Para instalar un controlador de instantánea, consulte "estas instrucciones" .

La compatibilidad del back-end de almacenamiento con Trident protege

Trident Protect es compatible con los siguientes back-ends de almacenamiento:

- Amazon FSX para ONTAP de NetApp
- Cloud Volumes ONTAP
- Cabinas de almacenamiento ONTAP de NetApp
- NetApp Volumes para Google Cloud
- Azure NetApp Files

Asegúrese de que el back-end de almacenamiento cumple los siguientes requisitos:

- Asegúrese de que el almacenamiento de NetApp conectado al clúster utilice Trident 24.02 o más reciente (se recomienda Trident 24.10).
- Asegúrese de tener un back-end de almacenamiento NetApp ONTAP.
- Asegúrese de haber configurado un depósito de almacenamiento de objetos para almacenar backups.
- Cree los espacios de nombres de aplicaciones que desee utilizar para las aplicaciones o las operaciones

de gestión de datos de aplicaciones. Trident Protect no crea estos espacios de nombres; si especifica un espacio de nombres no existente en un recurso personalizado, se producirá un error en la operación.

Requisitos para volúmenes de economía nas

Trident Protect admite las operaciones de backup y restauración en los volúmenes de economía nas. Actualmente no se admiten copias Snapshot, clones y replicación de SnapMirror en volúmenes económicos de nas. Debe habilitar un directorio snapshot para cada volumen económico nas que vaya a utilizar con Trident Protect.

Algunas aplicaciones no son compatibles con volúmenes que usan un directorio Snapshot. Para estas aplicaciones, debe ocultar el directorio Snapshot mediante la ejecución del siguiente comando en el sistema de almacenamiento de ONTAP:

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

Para habilitar el directorio snapshot, ejecute el siguiente comando para cada volumen nas-económico, sustituyéndolo `<volume-UUID>` por el UUID del volumen que desea cambiar:

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

Es posible habilitar los directorios de snapshots de forma predeterminada para volúmenes nuevos si se configura la opción Trident backend configuration `snapshotDir` en `true`. Los volúmenes existentes no se ven afectados.

Protección de datos con máquinas virtuales de KubeVirt

Trident Protect proporciona capacidades de congelación y descongelación del sistema de archivos para las máquinas virtuales KubeVirt durante las operaciones de protección de datos para garantizar la coherencia de los datos. El método de configuración y el comportamiento predeterminado para las operaciones de congelación de máquinas virtuales varían entre las versiones de Trident Protect; las versiones más recientes ofrecen una configuración simplificada mediante parámetros del gráfico Helm.

Durante las operaciones de restauración, cualquier `VirtualMachineSnapshots` Los datos creados para una máquina virtual (VM) no se restauran.

Trident Protect 25.10 y versiones posteriores

Trident Protect congela y descongela automáticamente los sistemas de archivos de KubeVirt durante las operaciones de protección de datos para garantizar la coherencia. A partir de Trident Protect 25.10, puede deshabilitar este comportamiento mediante la opción `vm.freeze` parámetro durante la instalación del gráfico Helm. El parámetro está habilitado por defecto.

```
helm install ... --set vm.freeze=false ...
```

Trident protege del 24.10.1 al 25.06

A partir de Trident Protect 24.10.1, Trident Protect congela y descongela automáticamente los sistemas de archivos KubeVirt durante las operaciones de protección de datos. De manera opcional, puede deshabilitar este comportamiento automático mediante el siguiente comando:

```
kubectl set env deployment/trident-protect-controller-manager
NEPTUNE_VM_FREEZE=false -n trident-protect
```

Trident Protect 24,10

Trident Protect 24,10 no garantiza automáticamente un estado coherente para los sistemas de archivos de máquinas virtuales KubeVirt durante las operaciones de protección de datos. Si desea proteger los datos de las máquinas virtuales KubeVirt con Trident Protect 24,10, debe habilitar manualmente la funcionalidad de congelación/descongelación para los sistemas de archivos antes de la operación de protección de datos. Esto garantiza que los sistemas de archivos estén en un estado consistente.

Puede configurar Trident Protect 24,10 para gestionar la congelación y descongelación del sistema de archivos de la máquina virtual durante las operaciones de protección de datos, mediante "configurar la virtualización"el siguiente comando:

```
kubectl set env deployment/trident-protect-controller-manager
NEPTUNE_VM_FREEZE=true -n trident-protect
```

Requisitos para la replicación de SnapMirror

La replicación de NetApp SnapMirror está disponible para usar con Trident Protect para las siguientes soluciones de ONTAP:

- Clústeres de NetApp FAS, AFF y ASA en las instalaciones
- ONTAP Select de NetApp
- Cloud Volumes ONTAP de NetApp
- Amazon FSX para ONTAP de NetApp

Requisitos de clústeres de ONTAP para la replicación de SnapMirror

Asegúrese de que el clúster de ONTAP cumple los siguientes requisitos si tiene pensado utilizar la replicación de SnapMirror:

- **NetApp Trident**: NetApp Trident debe existir en los clústeres de Kubernetes de origen y destino que utilizan ONTAP como back-end. Trident Protect admite la replicación con tecnología de NetApp SnapMirror mediante clases de almacenamiento respaldadas por los controladores siguientes:
  - `ontap-nas` : NFS
  - `ontap-san` : iSCSI
  - `ontap-san` :FC

- `ontap-san` :NVMe/TCP (requiere la versión mínima de ONTAP 9.15.1)

- **Licencias**: Las licencias asíncronas de SnapMirror de ONTAP que utilizan el paquete de protección de datos deben estar habilitadas en los clústeres de ONTAP de origen y de destino. Consulte "Información general sobre las licencias de SnapMirror en ONTAP" si desea obtener más información.

  A partir de ONTAP 9.10.1, todas las licencias se proporcionan como archivo de licencia de NetApp (NLF), que es un solo archivo que admite varias funciones. Consulte "Licencias incluidas con ONTAP One" si desea obtener más información.

  > Sólo se admite la protección asincrónica SnapMirror.

Consideraciones sobre la relación de paridad para la replicación de SnapMirror

Compruebe que el entorno cumple los siguientes requisitos si piensa utilizar la paridad de back-end de almacenamiento:

- **Cluster y SVM**: Los back-ends de almacenamiento ONTAP deben ser peered. Consulte "Información general sobre relaciones entre iguales de clústeres y SVM" si desea obtener más información.

  > Compruebe que los nombres de las SVM utilizados en la relación de replicación entre dos clústeres de ONTAP sean únicos.

- **NetApp Trident y SVM**: las SVM remotas emparejadas deben estar disponibles para NetApp Trident en el clúster de destino.
- **Backends administrados**: Necesitas agregar y administrar backends de almacenamiento ONTAP en Trident Protect para crear una relación de replicación.

Configuración de Trident/ONTAP para la replicación de SnapMirror

Trident Protect requiere que configure al menos un back-end de almacenamiento que admita la replicación para los clústeres de origen y destino. Si los clústeres de origen y destino son los mismos, la aplicación de destino debe usar un back-end de almacenamiento diferente al de la aplicación de origen para obtener la mejor resiliencia.

Requisitos del clúster de Kubernetes para la replicación de SnapMirror

Asegúrese de que sus clústeres de Kubernetes cumplan con los siguientes requisitos:

- **Accesibilidad de AppVault**: tanto los clústeres de origen como de destino deben tener acceso a la red para leer y escribir en AppVault para la replicación de objetos de la aplicación.
- **Conectividad de red**: configure reglas de firewall, permisos de bucket y listas de IP permitidas para habilitar la comunicación entre ambos clústeres y AppVault a través de WAN.

> Muchos entornos empresariales implementan políticas de firewall estrictas en las conexiones WAN. Verifique estos requisitos de red con su equipo de infraestructura antes de configurar la replicación.

Instalar y configurar Trident Protect

Si su entorno cumple los requisitos de protección Trident, puede seguir estos pasos para

instalar Trident Protect en el clúster. Puede obtener Trident Protect de NetApp o instalarlo desde su propio registro privado. La instalación desde un registro privado es útil si su clúster no puede acceder a Internet.

Instale Trident Protect

Instale Trident Protect de NetApp

Pasos

1. Añada el repositorio Helm de Trident:

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

2. Utilice Helm para instalar Trident Protect. Sustituya `<name-of-cluster>` por un nombre de clúster, que se asignará al clúster y se utilizará para identificar los backups y las snapshots del clúster:

```
helm install trident-protect netapp-trident-protect/trident-protect
--set clusterName=<name-of-cluster> --version 100.2510.0 --create
-namespace --namespace trident-protect
```

3. Opcionalmente, para habilitar el registro de depuración (recomendado para la resolución de problemas), utilice:

```
helm install trident-protect netapp-trident-protect/trident-protect
--set clusterName=<name-of-cluster> --set logLevel=debug --version
100.2510.0 --create-namespace --namespace trident-protect
```

   El registro de depuración ayuda al soporte de NetApp a solucionar problemas sin necesidad de cambiar el nivel de registro ni reproducir el problema.

Instale Trident Protect desde un registro privado

Puede instalar Trident Protect desde un registro de imágenes privado si su clúster de Kubernetes no puede acceder a Internet. En estos ejemplos, reemplace los valores entre paréntesis por información de su entorno:

Pasos

1. Tire de las siguientes imágenes a su máquina local, actualice las etiquetas y, a continuación, empújelas en su registro privado:

```
docker.io/netapp/controller:25.10.0
docker.io/netapp/restic:25.10.0
docker.io/netapp/kopia:25.10.0
docker.io/netapp/kopiablockrestore:25.10.0
docker.io/netapp/trident-autosupport:25.10.0
docker.io/netapp/exechook:25.10.0
docker.io/netapp/resourcebackup:25.10.0
docker.io/netapp/resourcerestore:25.10.0
docker.io/netapp/resourcedelete:25.10.0
docker.io/netapp/trident-protect-utils:v1.0.0
```

Por ejemplo:

```
docker pull docker.io/netapp/controller:25.10.0
```

```
docker tag docker.io/netapp/controller:25.10.0 <private-registry-url>/controller:25.10.0
```

```
docker push <private-registry-url>/controller:25.10.0
```

> Para obtener el gráfico de Helm, primero descárguelo en una máquina con acceso a Internet usando `helm pull trident-protect --version 100.2510.0 --repo https://netapp.github.io/trident-protect-helm-chart`, luego copie el resultado `trident-protect-100.2510.0.tgz` transfiera el archivo a su entorno sin conexión e instálelo usando `helm install trident-protect ./trident-protect-100.2510.0.tgz` en lugar de la referencia al repositorio en el paso final.

2. Cree el espacio de nombres del sistema Trident Protect:

```
kubectl create ns trident-protect
```

3. Inicie sesión en el Registro:

```
helm registry login <private-registry-url> -u <account-id> -p <api-token>
```

4. Cree un secreto de extracción para utilizarlo en la autenticación del registro privado:

```
kubectl create secret docker-registry regcred --docker
-username=<registry-username> --docker-password=<api-token> -n
trident-protect --docker-server=<private-registry-url>
```

5. Añada el repositorio Helm de Trident:

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

6. Crear un archivo llamado `protectValues.yaml`. Asegúrese de que contiene las siguientes configuraciones de Trident Protect:

```
---
imageRegistry: <private-registry-url>
imagePullSecrets:
  - name: regcred
```

> El `imageRegistry` y `imagePullSecrets` Los valores se aplican a todas las imágenes de componentes, incluyendo `resourcebackup` y `resourcerestore` . Si envías imágenes a una ruta de repositorio específica dentro de tu registro (por ejemplo, `example.com:443/my-repo`), incluya la ruta completa en el campo del registro. Esto garantizará que todas las imágenes se extraigan de `<private-registry-url>/<image-name>:<tag>`.

7. Utilice Helm para instalar Trident Protect. Sustituya `<name_of_cluster>` por un nombre de clúster, que se asignará al clúster y se utilizará para identificar los backups y las snapshots del clúster:

```
helm install trident-protect netapp-trident-protect/trident-protect
--set clusterName=<name_of_cluster> --version 100.2510.0 --create
-namespace --namespace trident-protect -f protectValues.yaml
```

8. Opcionalmente, para habilitar el registro de depuración (recomendado para la resolución de problemas), utilice:

```
helm install trident-protect netapp-trident-protect/trident-protect
--set clusterName=<name-of-cluster> --set logLevel=debug --version
100.2510.0 --create-namespace --namespace trident-protect -f
protectValues.yaml
```

El registro de depuración ayuda al soporte de NetApp a solucionar problemas sin necesidad de cambiar el nivel de registro ni reproducir el problema.

Para obtener opciones de configuración adicionales para el gráfico de Helm, incluidos los ajustes de AutoSupport y el filtrado de espacios de nombres, consulte "Personalice la instalación de Trident Protect".

Instale el complemento de la CLI Trident Protect

Puede utilizar el plugin de línea de comandos Trident Protect, que es una extensión de la utilidad Trident `tridentctl`, para crear e interactuar con los recursos personalizados de Trident Protect (CRS).

Instale el complemento de la CLI Trident Protect

Antes de utilizar la utilidad de línea de comandos, debe instalarla en la máquina que utiliza para acceder al clúster. Siga estos pasos, dependiendo de si su máquina utiliza una CPU x64 o ARM.

Descargar plugin para CPU Linux AMD64

Pasos

1. Descargue el complemento de la CLI de Trident Protect:

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.10.0/tridentctl-protect-linux-amd64
```

Descargar plugin para CPU Linux ARM64

Pasos

1. Descargue el complemento de la CLI de Trident Protect:

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.10.0/tridentctl-protect-linux-arm64
```

Descargar plugin para CPU Mac AMD64

Pasos

1. Descargue el complemento de la CLI de Trident Protect:

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.10.0/tridentctl-protect-macos-amd64
```

Descargar plugin para CPU Mac ARM64

Pasos

1. Descargue el complemento de la CLI de Trident Protect:

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/25.10.0/tridentctl-protect-macos-arm64
```

1. Active los permisos de ejecución para el binario del plugin:

```
chmod +x tridentctl-protect
```

2. Copie el binario del plugin a una ubicación definida en su variable PATH. Por ejemplo, `/usr/bin` o `/usr/local/bin` (puede que necesite Privileges elevado):

```
cp ./tridentctl-protect /usr/local/bin/
```

3. Opcionalmente, puede copiar el binario del plugin a una ubicación en su directorio principal. En este caso, se recomienda asegurarse de que la ubicación forma parte de la variable PATH:

```
cp ./tridentctl-protect ~/bin/
```

Copiar el plugin a una ubicación en su variable PATH le permite usar el plugin escribiendo `tridentctl-protect` o `tridentctl protect` desde cualquier ubicación.

Consulte la ayuda del complemento de la CLI de Trident

Puede utilizar las funciones de ayuda del plugin incorporado para obtener ayuda detallada sobre las capacidades del plugin:

Pasos

1. Utilice la función de ayuda para ver la guía de uso:

```
tridentctl-protect help
```

Habilite el autocompletado de comandos

Después de instalar el complemento de CLI Trident Protect, puede habilitar la finalización automática para ciertos comandos.

Active la finalización automática del shell Bash

Pasos

1. Crea el script de finalización:

```
tridentctl-protect completion bash > tridentctl-completion.bash
```

2. Cree un nuevo directorio en el directorio principal para que contenga el script:

```
mkdir -p ~/.bash/completions
```

3. Mueva el script descargado al `~/.bash/completions` directorio:

```
mv tridentctl-completion.bash ~/.bash/completions/
```

4. Añada la siguiente línea al `~/.bashrc` archivo en su directorio principal:

```
source ~/.bash/completions/tridentctl-completion.bash
```

Active la finalización automática del shell Z

Pasos

1. Crea el script de finalización:

```
tridentctl-protect completion zsh > tridentctl-completion.zsh
```

2. Cree un nuevo directorio en el directorio principal para que contenga el script:

```
mkdir -p ~/.zsh/completions
```

3. Mueva el script descargado al `~/.zsh/completions` directorio:

```
mv tridentctl-completion.zsh ~/.zsh/completions/
```

4. Añada la siguiente línea al `~/.zprofile` archivo en su directorio principal:

```
source ~/.zsh/completions/tridentctl-completion.zsh
```

Resultado

En su próximo inicio de sesión en el shell, puede utilizar el comando auto-completado con el plugin tridentctl-Protect.

Personalice la instalación de Trident Protect

Es posible personalizar la configuración predeterminada de Trident Protect para cumplir con los requisitos específicos del entorno.

Especifique los límites de recursos del contenedor Trident Protect

Puede utilizar un archivo de configuración para especificar límites de recursos para contenedores Trident Protect después de instalar Trident Protect. La configuración de límites de recursos permite controlar cuántos recursos del clúster consumen las operaciones de Trident Protect.

Pasos

1. Crear un archivo llamado `resourceLimits.yaml`.
2. Rellene el archivo con opciones de límite de recursos para contenedores Trident Protect según las necesidades de su entorno.

   El siguiente archivo de configuración de ejemplo muestra la configuración disponible y contiene los valores predeterminados para cada límite de recursos:

```
---
jobResources:
  defaults:
    limits:
      cpu: 8000m
      memory: 10000Mi
      ephemeralStorage: ""
    requests:
      cpu: 100m
      memory: 100Mi
      ephemeralStorage: ""
  resticVolumeBackup:
    limits:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
    requests:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
  resticVolumeRestore:
    limits:
      cpu: ""
      memory: ""
```

```
      ephemeralStorage: ""
    requests:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
  kopiaVolumeBackup:
    limits:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
    requests:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
  kopiaVolumeRestore:
    limits:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
    requests:
      cpu: ""
      memory: ""
      ephemeralStorage: ""
```

3. Aplique los valores del `resourceLimits.yaml` archivo:

```
helm upgrade trident-protect -n trident-protect netapp-trident-
protect/trident-protect -f resourceLimits.yaml --reuse-values
```

Personalizar restricciones de contexto de seguridad

Puede utilizar un archivo de configuración para modificar la restricción de contexto de seguridad de OpenShift (SCCs) para los contenedores de Trident Protect después de instalar Trident Protect. Estas restricciones definen restricciones de seguridad para los pods en un clúster de Red Hat OpenShift.

Pasos

1. Crear un archivo llamado `sccconfig.yaml`.
2. Agregue la opción SCC al archivo y modifique los parámetros según las necesidades de su entorno.

   El siguiente ejemplo muestra los valores predeterminados de los parámetros para la opción SCC:

```
scc:
  create: true
  name: trident-protect-job
  priority: 1
```

En esta tabla se describen los parámetros de la opción SCC:

Parámetro	Descripción	Predeterminado
cree	Determina si se puede crear un recurso SCC. Un recurso de SCC se creará sólo si `scc.create` se establece en `true` y el proceso de instalación de Helm identifica un entorno de OpenShift. Si no funciona en OpenShift, o si `scc.create` está establecido en `false`, no se creará ningún recurso SCC.	verdadero
nombre	Especifica el nombre del SCC.	Trident-protect-job
prioridad	Define la prioridad del SCC. Los SCCS con valores de prioridad más altos se evalúan antes que aquellos con valores más bajos.	1

3. Aplique los valores del `sccconfig.yaml` archivo:

```
helm upgrade trident-protect netapp-trident-protect/trident-protect -f
sccconfig.yaml --reuse-values
```

Esto reemplazará los valores predeterminados por los especificados en el `sccconfig.yaml` archivo.

Configurar ajustes adicionales del gráfico de protección del timón Trident

Puede personalizar la configuración de AutoSupport y el filtrado de espacios de nombres para satisfacer sus requisitos específicos. La siguiente tabla describe los parámetros de configuración disponibles:

Parámetro	Tipo	Descripción
autoSupport.proxy	cadena	Configura una URL de proxy para conexiones de NetApp AutoSupport . Use esto para enrutar las cargas de paquetes de soporte a través de un servidor proxy. Ejemplo: `http://my.proxy.url` .

Parámetro	Tipo	Descripción
autoSupport.inseguro	booleano	Omite la verificación TLS para las conexiones proxy de AutoSupport cuando se configura en `true` . Úselo sólo para conexiones proxy inseguras. (por defecto: `false`)
autoSupport.enabled	booleano	Habilita o deshabilita las cargas diarias del paquete AutoSupport de Trident Protect. Cuando se establece en `false` Las cargas diarias programadas están deshabilitadas, pero aún puedes generar paquetes de soporte manualmente. (por defecto: `true`)
restaurar anotaciones de SkipNamespace	cadena	Lista separada por comas de anotaciones de espacios de nombres para excluir de las operaciones de copia de seguridad y restauración. Le permite filtrar espacios de nombres según anotaciones.
restaurar etiquetas de espacios de nombres de saltos	cadena	Lista separada por comas de etiquetas de espacios de nombres para excluir de las operaciones de copia de seguridad y restauración. Le permite filtrar espacios de nombres según etiquetas.

Puede configurar estas opciones mediante un archivo de configuración YAML o indicadores de línea de comandos:

Utilice el archivo YAML

Pasos

1. Crea un archivo de configuración y nómbralo `values.yaml` .
2. En el archivo que ha creado, agregue las opciones de configuración que desea personalizar.

```
autoSupport:
  enabled: false
  proxy: http://my.proxy.url
  insecure: true
restoreSkipNamespaceAnnotations: "annotation1,annotation2"
restoreSkipNamespaceLabels: "label1,label2"
```

3. Después de rellenar el `values.yaml` archivo con los valores correctos, aplique el archivo de configuración:

```
helm upgrade trident-protect -n trident-protect netapp-trident-
protect/trident-protect -f values.yaml --reuse-values
```

Usar la bandera CLI

Pasos

1. Utilice el siguiente comando con el `--set` bandera para especificar parámetros individuales:

```
helm upgrade trident-protect -n trident-protect netapp-trident-
protect/trident-protect \
  --set autoSupport.enabled=false \
  --set autoSupport.proxy=http://my.proxy.url \
  --set restoreSkipNamespaceAnnotations="annotation1,annotation2" \
  --set restoreSkipNamespaceLabels="label1,label2" \
  --reuse-values
```

Restrinja los pods de protección de Trident a nodos específicos

Puede usar la restricción de selección de nodos de Kubernetes nodeSelector para controlar qué nodos pueden ejecutar pods de Trident Protect, según las etiquetas de los nodos. De forma predeterminada, Trident Protect está restringido a los nodos que ejecutan Linux. Puede personalizar aún más estas restricciones en función de sus necesidades.

Pasos

1. Crear un archivo llamado `nodeSelectorConfig.yaml`.
2. Agregue la opción nodeSelector al archivo y modifique el archivo para agregar o cambiar etiquetas de nodo para restringir según las necesidades del entorno. Por ejemplo, el siguiente archivo contiene la

restricción predeterminada del sistema operativo, pero también se dirige a una región y un nombre de aplicación específicos:

```
nodeSelector:
  kubernetes.io/os: linux
  region: us-west
  app.kubernetes.io/name: mysql
```

3. Aplique los valores del `nodeSelectorConfig.yaml` archivo:

```
helm upgrade trident-protect -n trident-protect netapp-trident-
protect/trident-protect -f nodeSelectorConfig.yaml --reuse-values
```

Esto reemplaza las restricciones predeterminadas por las especificadas en el `nodeSelectorConfig.yaml` archivo.