



Instalar Trident Protect

Trident

NetApp
October 30, 2026

This PDF was generated from <https://docs.netapp.com/pt-br/trident-2606/trident-protect/trident-protect-requirements.html> on October 30, 2026. Always check docs.netapp.com for the latest.

Índice

Instalar Trident Protect	1
Requisitos do Trident Protect	1
Compatibilidade do cluster Kubernetes com o Trident Protect	1
Compatibilidade do backend de armazenamento Trident Protect	1
Requisitos para volumes nas-economy	2
Protegendo dados com KubeVirt VMs	2
Requisitos para SnapMirror replication	3
Instalar e configurar Trident Protect	5
Instalar Trident Protect	5
Instale o plugin Trident Protect CLI	7
Instale o plugin Trident Protect CLI	8
Veja a ajuda do plugin Trident CLI	9
Ativar o preenchimento automático de comandos	9
Personalizar a instalação do Trident Protect	10
Especifique os limites de recursos do contêiner Trident Protect	10
Personalizar restrições de contexto de segurança	12
Configurar configurações adicionais do helm chart do Trident Protect	13
Restringir os pods do Trident Protect a nós específicos	15

Instalar Trident Protect

Requisitos do Trident Protect

Comece verificando a prontidão do seu ambiente operacional, clusters de aplicativos, aplicativos e licenças. Certifique-se de que seu ambiente atenda a esses requisitos para implantar e operar Trident Protect.

Compatibilidade do cluster Kubernetes com o Trident Protect

Trident Protect é compatível com uma ampla gama de ofertas de Kubernetes totalmente gerenciadas e autogerenciadas, incluindo:

- Amazon Elastic Kubernetes Service (EKS)
- Google Kubernetes Engine (GKE)
- Microsoft Azure Kubernetes Service (AKS)
- Red Hat OpenShift
- SUSE Harvester 1.7.0 (ONTAP iSCSI)
- SUSE Rancher
- VMware Tanzu Portfolio
- Kubernetes upstream



- Os backups do Trident Protect são suportados apenas em nós de computação Linux. Nós de computação Windows não são suportados para operações de backup.
- Certifique-se de que o cluster no qual você instala Trident Protect esteja configurado com um controlador de snapshot em execução e os CRDs relacionados. Para instalar um controlador de snapshot, consulte "[estas instruções](#)".
- Certifique-se de que exista pelo menos um VolumeSnapshotClass. Para obter mais informações, consulte "[VolumeSnapshotClass](#)".
- É necessário o Helm 4.x ou posterior para instalar Trident Protect.

Compatibilidade do backend de armazenamento Trident Protect

Trident Protect é compatível com os seguintes storage backends:

- Amazon FSx for NetApp ONTAP
- Cloud Volumes ONTAP
- Matrices de armazenamento ONTAP
- Google Cloud NetApp Volumes
- Azure NetApp Files

Certifique-se de que seu storage backend atenda aos seguintes requisitos:

- Certifique-se de que o armazenamento NetApp conectado ao cluster esteja usando Trident 24.02 ou mais recente (Trident 24.10 é recomendado).

- Certifique-se de ter um backend de storage NetApp ONTAP.
- Certifique-se de ter configurado um storage de objetos para armazenar backups.
- Crie todos os namespaces de aplicativos que você planeja usar para aplicativos ou operações de gerenciamento de dados de aplicativos. Trident Protect não cria esses namespaces para você; se você especificar um namespace inexistente em um recurso personalizado, a operação falhará.

Requisitos para volumes nas-economy

Trident Protect oferece suporte a operações de backup e restauração em volumes nas-economy. Snapshots, clones e SnapMirror replicação para volumes nas-economy não são suportados atualmente. Você precisa habilitar um diretório de snapshot para cada volume nas-economy que planeja usar com Trident Protect.



Algumas aplicações não são compatíveis com volumes que utilizam um diretório de snapshot. Para essas aplicações, você precisa ocultar o diretório de snapshot executando o seguinte comando no sistema de storage ONTAP:

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

Você pode habilitar o diretório de snapshots executando o seguinte comando para cada volume nas-economy, substituindo <volume-UUID> pelo UUID do volume que deseja alterar:

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



Você pode habilitar diretórios de snapshots por padrão para novos volumes definindo a opção de configuração do backend Trident `snapshotDir` para `true`. Volumes existentes não são afetados.

Protegendo dados com KubeVirt VMs

Trident Protect oferece recursos de congelamento e descongelamento do sistema de arquivos para máquinas virtuais KubeVirt durante operações de proteção de dados para garantir a consistência de dados. O método de configuração e o comportamento padrão para operações de congelamento de máquinas virtuais variam entre as versões do Trident Protect, sendo que as versões mais recentes oferecem configuração simplificada por meio de parâmetros do Helm chart.



Durante as operações de restauração, quaisquer `VirtualMachineSnapshots` criados para uma máquina virtual (VM) não são restaurados.

▼ Trident Protect 25.10 e mais recentes

Trident Protect congela e descongela automaticamente os sistemas de arquivos KubeVirt durante as operações de proteção de dados para garantir a consistência. A partir do Trident Protect 25.10, você pode desativar esse comportamento usando o parâmetro `vm.freeze` durante a instalação do Helm chart. O parâmetro está ativado por padrão.

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

▼ Trident Protect 24.10.1 a 25.06

A partir do Trident Protect 24.10.1, o Trident Protect congela e descongela automaticamente os sistemas de arquivos KubeVirt durante as operações de proteção de dados. Opcionalmente, você pode desativar esse comportamento automático usando o seguinte comando:

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

▼ Trident Protect 24.10

Trident Protect 24.10 não garante automaticamente um estado consistente para os sistemas de arquivos de KubeVirt VM durante as operações de proteção de dados. Se você deseja proteger os dados da sua KubeVirt VM usando Trident Protect 24.10, precisa habilitar manualmente a funcionalidade de congelamento/descongelamento dos sistemas de arquivos antes da operação de proteção de dados. Isso garante que os sistemas de arquivos estejam em um estado consistente.

Você pode configurar Trident Protect 24.10 para gerenciar o congelamento e descongelamento do sistema de arquivos da VM durante as operações de proteção de dados "[configurando virtualização](#)" e, em seguida, usar o seguinte comando:

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

Para aprender como definir aplicativos e restaurar VMs com Trident Protect, consulte "[Proteja máquinas virtuais KubeVirt com Trident Protect](#)".

Requisitos para SnapMirror replication

NetApp SnapMirror a replicação está disponível para uso com Trident Protect para as seguintes soluções ONTAP:

- Sistemas NetApp FAS, AFF e ASA locais. A replicação SnapMirror com Trident protect não é atualmente suportada para sistemas ASA r2.
- NetApp ONTAP Select
- NetApp Cloud Volumes ONTAP
- Amazon FSx for NetApp ONTAP

Requisitos do cluster ONTAP para replicação SnapMirror

Certifique-se de que seu cluster ONTAP atenda aos seguintes requisitos caso planeje usar a replicação SnapMirror:

- **NetApp Trident:** NetApp Trident deve existir em ambos os clusters Kubernetes de origem e destino que utilizam ONTAP como backend. Trident Protect oferece suporte à replicação com a tecnologia NetApp SnapMirror usando classes de armazenamento suportadas pelos seguintes drivers:

- `ontap-nas`: NFS
- `ontap-san`: iSCSI
- `ontap-san`: FC
- `ontap-san`: NVMe/TCP (requer versão mínima do ONTAP 9.15.1)

- **Licenças:** As licenças assíncronas do ONTAP SnapMirror usando o pacote Data Protection devem estar habilitadas tanto nos clusters ONTAP de origem quanto de destino. Consulte "[SnapMirror visão geral do licenciamento no ONTAP](#)" para obter mais informações.

A partir do ONTAP 9.10.1, todas as licenças são fornecidas como um arquivo de licença NetApp (NLF), que é um único arquivo que habilita vários recursos. Consulte "[Licenças incluídas com o ONTAP One](#)" para obter mais informações.



Apenas a proteção assíncrona SnapMirror é suportada.

Considerações de peering para replicação SnapMirror

Certifique-se de que seu ambiente atenda aos seguintes requisitos caso planeje usar o peering de storage backend:

- **Cluster e SVM:** Os backends de armazenamento ONTAP devem estar emparelhados. Consulte "[Visão geral do peering de cluster e SVM](#)" para obter mais informações.



Certifique-se de que os nomes SVM usados na relação de replicação entre dois clusters ONTAP sejam únicos.

- **NetApp Trident e SVM:** Os SVMs remotos emparelhados devem estar disponíveis para NetApp Trident no cluster de destino.
- **Backends gerenciados:** Você precisa adicionar e gerenciar backends de armazenamento ONTAP no Trident Protect para criar uma relação de replicação.

Configuração do Trident / ONTAP para replicação SnapMirror

Trident Protect exige que você configure pelo menos um storage backend que suporte replicação para ambos os clusters de origem e destino. Se os clusters de origem e destino forem os mesmos, o aplicativo de destino deve usar um storage backend diferente do aplicativo de origem para a melhor resiliência.

Requisitos do cluster Kubernetes para SnapMirror replication

Certifique-se de que seus clusters Kubernetes atendam aos seguintes requisitos:

- **AppVault acessibilidade:** Tanto o cluster de origem quanto o cluster de destino devem ter acesso à rede para ler e gravar na AppVault para replicação de objetos da aplicação.
- **Conectividade de rede:** Configure regras de firewall, permissões de bucket e listas de permissão de IP para habilitar a comunicação entre ambos os clusters e o AppVault através das WANs.



Muitos ambientes corporativos implementam políticas de firewall rigorosas em conexões WAN. Verifique esses requisitos de rede com sua equipe de infraestrutura antes de configurar a replicação.

Instalar e configurar Trident Protect

Se o seu ambiente atender aos requisitos do Trident Protect, você pode seguir estas etapas para instalar o Trident Protect no seu cluster. Você pode obter o Trident Protect da NetApp ou instalá-lo a partir do seu próprio registro privado. Instalar a partir de um registro privado é útil se o seu cluster não puder acessar a Internet.

Instalar Trident Protect

Instale Trident Protect a partir de NetApp

Passos

1. Adicione o repositório Trident:

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

2. Use o Helm para instalar Trident Protect. Substitua <name-of-cluster> pelo nome do cluster, que será atribuído ao cluster e usado para identificar os backups e snapshots do cluster:

```
helm install trident-protect netapp-trident-protect/trident-protect  
--set clusterName=<name-of-cluster> --version 100.2606.0 --create  
--namespace --namespace trident-protect
```

3. Opcionalmente, para ativar o registro de depuração (recomendado para solução de problemas), use:

```
helm install trident-protect netapp-trident-protect/trident-protect  
--set clusterName=<name-of-cluster> --set logLevel=debug --version  
100.2606.0 --create-namespace --namespace trident-protect
```

O registro de depuração ajuda NetApp support a solucionar problemas sem exigir alterações no nível de registro ou reprodução do problema.

Instale Trident Protect a partir de um registro privado

Você pode instalar Trident Protect a partir de um registro de imagens privado caso seu cluster Kubernetes não tenha acesso à Internet. Nestes exemplos, substitua os valores entre colchetes pelas informações do seu ambiente:

Passos

1. Baixe as seguintes imagens para sua máquina local, atualize as tags e, em seguida, envie-as para seu registro privado:

```
docker.io/netapp/controller:26.06.0  
docker.io/netapp/restic:26.06.0  
docker.io/netapp/kopia:26.06.0
```

```
docker.io/netapp/kopiablockrestore:26.06.0
docker.io/netapp/trident-autosupport:26.06.0
docker.io/netapp/execheek:26.06.0
docker.io/netapp/resourcebackup:26.06.0
docker.io/netapp/resourcerestore:26.06.0
docker.io/netapp/resourcedelete:26.06.0
docker.io/netapp/trident-protect-utils:v1.0.0
```

Por exemplo:

```
docker pull docker.io/netapp/controller:26.06.0
```

```
docker tag docker.io/netapp/controller:26.06.0 <private-registry-
url>/controller:26.06.0
```

```
docker push <private-registry-url>/controller:26.06.0
```



Para obter o Helm chart, primeiro baixe o Helm chart em uma máquina com acesso à internet usando `helm pull trident-protect --version 100.2606.0 --repo https://netapp.github.io/trident-protect-helm-chart`, depois copie o arquivo resultante `trident-protect-100.2606.0.tgz` para seu ambiente offline e instale usando `helm install trident-protect ./trident-protect-100.2606.0.tgz` em vez da referência ao repositório na etapa final.

2. Crie o namespace do sistema Trident Protect:

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

3. Faça login no registro:

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

4. Crie um segredo de pull para usar na autenticação do 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. Adicione o repositório Trident:

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

6. Crie um arquivo chamado `protectValues.yaml`. Certifique-se de que ele contenha as seguintes configurações do Trident Protect:

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



Os `imageRegistry` e `imagePullSecrets` valores se aplicam a todas as imagens de componentes, incluindo `resourcebackup` e `resourcerestore`. Se você enviar imagens para um caminho de repositório específico em seu registro (por exemplo, `example.com:443/my-repo`), inclua o caminho completo no campo de registro. Isso garantirá que todas as imagens sejam obtidas de `<private-registry-url>/<image-name>:<tag>`.

7. Use o Helm para instalar Trident Protect. Substitua `<name_of_cluster>` pelo nome do cluster, que será atribuído ao cluster e usado para identificar os backups e snapshots do cluster:

```
helm install trident-protect netapp-trident-protect/trident-protect
--set clusterName=<name_of_cluster> --version 100.2606.0 --create
--namespace --namespace trident-protect -f protectValues.yaml
```

8. Opcionalmente, para ativar o registro de depuração (recomendado para solução de problemas), use:

```
helm install trident-protect netapp-trident-protect/trident-protect
--set clusterName=<name-of-cluster> --set logLevel=debug --version
100.2606.0 --create-namespace --namespace trident-protect -f
protectValues.yaml
```

O registro de depuração ajuda NetApp support a solucionar problemas sem exigir alterações no nível de registro ou reprodução do problema.



Para opções adicionais de configuração do gráfico Helm, incluindo configurações do AutoSupport e filtragem de namespace, consulte "[Personalizar a instalação do Trident Protect](#)".

Instale o plugin Trident Protect CLI

Você pode usar o plugin de linha de comando Trident Protect, que é uma extensão do utilitário Trident `tridentctl`, para criar e interagir com recursos personalizados (CRs)

do Trident Protect.

Instale o plugin Trident Protect CLI

Antes de usar o utilitário de linha de comando, você precisa instalá-lo na máquina que usa para acessar seu cluster. Siga estas etapas, dependendo se sua máquina usa uma CPU x64 ou ARM.

Baixar plugin para CPUs Linux AMD64

Passos

1. Baixe o plugin Trident Protect CLI:

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/v26.06.0/tridentctl-protect-linux-amd64
```

Baixar plugin para CPUs Linux ARM64

Passos

1. Baixe o plugin Trident Protect CLI:

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/v26.06.0/tridentctl-protect-linux-arm64
```

Baixar plugin para CPUs Mac AMD64

Passos

1. Baixe o plugin Trident Protect CLI:

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/v26.06.0/tridentctl-protect-macos-amd64
```

Baixar plugin para CPUs Mac ARM64

Passos

1. Baixe o plugin Trident Protect CLI:

```
curl -L -o tridentctl-protect https://github.com/NetApp/tridentctl-protect/releases/download/v26.06.0/tridentctl-protect-macos-arm64
```

1. Habilite as permissões de execução para o binário do plugin:

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

2. Copie o binário do plugin para um local definido na sua variável PATH. Por exemplo, `/usr/bin` ou

/usr/local/bin (você pode precisar de privilégios elevados):

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

3. Opcionalmente, você pode copiar o binário do plugin para um local no seu diretório base. Nesse caso, é recomendável garantir que o local faça parte da sua variável PATH:

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



Copiar o plugin para um local na sua variável PATH permite que você utilize o plugin digitando `tridentctl-protect` ou `tridentctl protect` de qualquer local.

Veja a ajuda do plugin Trident CLI

Você pode usar os recursos de ajuda incorporado do plugin para obter ajuda detalhada sobre as funcionalidades do plugin:

Passos

1. Use a função de ajuda para visualizar as instruções de utilização:

```
tridentctl-protect help
```

Ativar o preenchimento automático de comandos

Após instalar o plugin Trident Protect CLI, você pode ativar o recurso de autocompletar para determinados comandos.

Ative o recurso de autocompletar para o shell Bash

Passos

1. Crie o script de conclusão:

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

2. Crie um novo diretório em seu diretório base para conter o script:

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

3. Mova o script baixado para o diretório ~/.bash/completions:

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

4. Adicione a seguinte linha ao arquivo `~/.bashrc` no seu diretório base:

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

Ative o recurso de autocompletar para o Z shell

Passos

1. Crie o script de conclusão:

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

2. Crie um novo diretório em seu diretório base para conter o script:

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

3. Mova o script baixado para o diretório `~/.zsh/completions`:

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

4. Adicione a seguinte linha ao arquivo `~/.zprofile` no seu diretório base:

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

Resultado

Na sua próxima sessão de login no shell, você pode usar o recurso de autocompletar comandos com o plugin `tridentctl-protect`.

Personalizar a instalação do Trident Protect

Você pode personalizar a configuração padrão do Trident Protect para atender aos requisitos específicos do seu ambiente.

Especifique os limites de recursos do contêiner Trident Protect

Você pode usar um arquivo de configuração para especificar limites de recursos para os contêineres do Trident Protect após instalar o Trident Protect. Definir limites de recursos permite controlar quanto dos recursos do cluster são consumidos pelas operações do Trident Protect.

Passos

1. Crie um arquivo chamado `resourceLimits.yaml`.
2. Preencha o arquivo com as opções de limite de recursos para os contêineres do Trident Protect de acordo com as necessidades do seu ambiente.

O seguinte arquivo de configuração mostra as configurações disponíveis e contém os valores padrão para cada limite de recurso:

```
---
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 os valores do arquivo `resourceLimits.yaml`:

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

Personalizar restrições de contexto de segurança

Você pode usar um arquivo de configuração para modificar as restrições de contexto de segurança (OpenShift SCCs) para os contêineres do Trident Protect após instalar o Trident Protect. Essas restrições definem limitações de segurança para pods em um cluster Red Hat OpenShift.

Passos

1. Crie um arquivo chamado `sccconfig.yaml`.
2. Adicione a opção SCC ao arquivo e modifique os parâmetros de acordo com as necessidades do seu ambiente.

O exemplo a seguir mostra os valores padrão dos parâmetros para a opção SCC:

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

Esta tabela descreve os parâmetros para a opção SCC:

Parâmetro	Descrição	Padrão
criar	Determina se um recurso SCC pode ser criado. Um recurso SCC será criado somente se <code>scc.create</code> estiver definido como <code>true</code> e o processo de instalação do Helm identificar um ambiente OpenShift. Se não estiver operando em OpenShift, ou se <code>scc.create</code> estiver definido como <code>false</code> , nenhum recurso SCC será criado.	verdadeiro
nome	Especifica o nome do SCC.	trident-protect-job

Parâmetro	Descrição	Padrão
prioridade	Define a prioridade do SCC. SCCs com valores de prioridade mais altos são avaliados antes daqueles com valores mais baixos.	1

3. Aplique os valores do arquivo `sccconfig.yaml`:

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

Isso substituirá os valores padrão pelos valores especificados no arquivo `sccconfig.yaml`.

Configurar configurações adicionais do helm chart do Trident Protect

Você pode personalizar as configurações do AutoSupport, o filtro de namespace e o gerenciamento de certificados de webhook do OpenShift para atender às suas necessidades específicas. A tabela a seguir descreve os parâmetros de configuração disponíveis:

Parâmetro	Tipo	Descrição
AutoSupport.proxy	string	Configura uma URL de proxy para conexões do NetApp AutoSupport. Use isto para rotear uploads de pacotes de suporte por meio de um servidor proxy. Exemplo: http://my.proxy.url .
AutoSupport.insecure	booleano	Ignora a verificação TLS para conexões proxy do AutoSupport quando definido como <code>true</code> . Use somente para conexões proxy inseguras. (padrão: <code>false</code>)
AutoSupport.enabled	booleano	Ativa ou desativa o envio diário de pacotes do Trident Protect AutoSupport. Quando definido como <code>false</code> , os envios diários agendados são desativados, mas você ainda pode gerar pacotes de suporte manualmente. (padrão: <code>true</code>)
restoreSkipNamespaceAnnotations	string	Lista de anotações de namespace separadas por vírgulas para excluir das operações de backup e restauração. Permite filtrar namespaces com base em anotações.

Parâmetro	Tipo	Descrição
restoreSkipNamespaceLabels	string	Lista de rótulos de namespace separados por vírgula para excluir das operações de backup e restauração. Permite filtrar namespaces com base nos rótulos.
useOpenshiftServiceCA	booleano	Habilita a integração com o operador de CA de serviço nativo do OpenShift para criação e injeção automática de certificados TLS para o webhook de admissão do Trident Protect. Quando definido como <code>true</code> , o operador de CA de serviço gerencia automaticamente os certificados TLS do webhook. Use somente em clusters OpenShift. (padrão: <code>false</code>)

Você pode configurar essas opções usando um arquivo de configuração YAML ou parâmetros de linha de comando:

Use o arquivo YAML

Passos

1. Crie um arquivo de configuração e nomeie-o `values.yaml`.
2. No arquivo que você criou, adicione as opções de configuração que deseja personalizar.

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

3. Após preencher o `values.yaml` file com os valores corretos, aplique o arquivo de configuração:

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

Use a flag da CLI

Passos

1. Utilize o seguinte comando com a flag `--set` para especificar parâmetros individuais:

```
helm upgrade trident-protect -n trident-protect netapp-trident-
```

```
protect/trident-protect \
  --set autoSupport.enabled=false \
  --set autoSupport.proxy=http://my.proxy.url \
  --set-string
restoreSkipNamespaceAnnotations="{annotation1,annotation2}" \
  --set-string restoreSkipNamespaceLabels="{label1,label2}" \
  --set useOpenshiftServiceCA=true \
  --reuse-values
```

Restringir os pods do Trident Protect a nós específicos

Você pode usar a restrição de seleção de nós `nodeSelector` do Kubernetes para controlar quais dos seus nós estão qualificados para executar pods do Trident Protect, com base nos rótulos dos nós. Por padrão, Trident Protect é restrito a nós que executam Linux. Você pode personalizar ainda mais essas restrições de acordo com suas necessidades.

Passos

1. Crie um arquivo chamado `nodeSelectorConfig.yaml`.
2. Adicione a opção `nodeSelector` ao arquivo e modifique o arquivo para adicionar ou alterar rótulos de nós para restringir de acordo com as necessidades do seu ambiente. Por exemplo, o seguinte arquivo contém a restrição padrão do sistema operacional, mas também direciona para uma região e um nome de aplicativo específicos:

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

3. Aplique os valores do arquivo `nodeSelectorConfig.yaml`:

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

Isso substitui as restrições padrão pelas que você especificou no arquivo `nodeSelectorConfig.yaml`.

Informações sobre direitos autorais

Copyright © 2026 NetApp, Inc. Todos os direitos reservados. Impresso nos EUA. Nenhuma parte deste documento protegida por direitos autorais pode ser reproduzida de qualquer forma ou por qualquer meio — gráfico, eletrônico ou mecânico, incluindo fotocópia, gravação, gravação em fita ou storage em um sistema de recuperação eletrônica — sem permissão prévia, por escrito, do proprietário dos direitos autorais.

O software derivado do material da NetApp protegido por direitos autorais está sujeito à seguinte licença e isenção de responsabilidade:

ESTE SOFTWARE É FORNECIDO PELA NETAPP "NO PRESENTE ESTADO" E SEM QUAISQUER GARANTIAS EXPRESSAS OU IMPLÍCITAS, INCLUINDO, SEM LIMITAÇÕES, GARANTIAS IMPLÍCITAS DE COMERCIALIZAÇÃO E ADEQUAÇÃO A UM DETERMINADO PROPÓSITO, CONFORME A ISENÇÃO DE RESPONSABILIDADE DESTES DOCUMENTOS. EM HIPÓTESE ALGUMA A NETAPP SERÁ RESPONSÁVEL POR QUALQUER DANO DIRETO, INDIRETO, INCIDENTAL, ESPECIAL, EXEMPLAR OU CONSEQUENCIAL (INCLUINDO, SEM LIMITAÇÕES, AQUISIÇÃO DE PRODUTOS OU SERVIÇOS SOBRESSAIENTES; PERDA DE USO, DADOS OU LUCROS; OU INTERRUPÇÃO DOS NEGÓCIOS), INDEPENDENTEMENTE DA CAUSA E DO PRINCÍPIO DE RESPONSABILIDADE, SEJA EM CONTRATO, POR RESPONSABILIDADE OBJETIVA OU PREJUÍZO (INCLUINDO NEGLIGÊNCIA OU DE OUTRO MODO), RESULTANTE DO USO DESTES DOCUMENTOS, MESMO SE ADVERTIDA DA RESPONSABILIDADE DE TAL DANO.

A NetApp reserva-se o direito de alterar quaisquer produtos descritos neste documento, a qualquer momento e sem aviso. A NetApp não assume nenhuma responsabilidade nem obrigação decorrentes do uso dos produtos descritos neste documento, exceto conforme expressamente acordado por escrito pela NetApp. O uso ou a compra deste produto não representam uma licença sob quaisquer direitos de patente, direitos de marca comercial ou quaisquer outros direitos de propriedade intelectual da NetApp.

O produto descrito neste manual pode estar protegido por uma ou mais patentes dos EUA, patentes estrangeiras ou pedidos pendentes.

LEGENDA DE DIREITOS LIMITADOS: o uso, a duplicação ou a divulgação pelo governo estão sujeitos a restrições conforme estabelecido no subparágrafo (b)(3) dos Direitos em Dados Técnicos - Itens Não Comerciais no DFARS 252.227-7013 (fevereiro de 2014) e no FAR 52.227- 19 (dezembro de 2007).

Os dados aqui contidos pertencem a um produto comercial e/ou serviço comercial (conforme definido no FAR 2.101) e são de propriedade da NetApp, Inc. Todos os dados técnicos e software de computador da NetApp fornecidos sob este Contrato são de natureza comercial e desenvolvidos exclusivamente com despesas privadas. O Governo dos EUA tem uma licença mundial limitada, irrevogável, não exclusiva, intransferível e não sublicenciável para usar os Dados que estão relacionados apenas com o suporte e para cumprir os contratos governamentais desse país que determinam o fornecimento de tais Dados. Salvo disposição em contrário no presente documento, não é permitido usar, divulgar, reproduzir, modificar, executar ou exibir os dados sem a aprovação prévia por escrito da NetApp, Inc. Os direitos de licença pertencentes ao governo dos Estados Unidos para o Departamento de Defesa estão limitados aos direitos identificados na cláusula 252.227-7015(b) (fevereiro de 2014) do DFARS.

Informações sobre marcas comerciais

NETAPP, o logotipo NETAPP e as marcas listadas em <http://www.netapp.com/TM> são marcas comerciais da NetApp, Inc. Outros nomes de produtos e empresas podem ser marcas comerciais de seus respectivos proprietários.