



Comece agora

Astra Trident

NetApp
January 14, 2026

Índice

Comece agora	1
Saiba mais sobre o Astra Trident	1
Saiba mais sobre o Astra Trident	1
A arquitetura do Astra Trident	2
Conceitos	5
Início rápido para Astra Trident	9
O que se segue?	10
Requisitos	10
Informações essenciais sobre o Astra Trident	10
Frontens suportados (orquestradores)	10
Backends suportados (armazenamento)	11
Requisitos de recursos	11
Sistemas operacionais de host testados	11
Configuração de host	12
Configuração do sistema de storage	12
Portas Astra Trident	12
Imagens de contêineres e versões correspondentes do Kubernetes	12

Comece agora

Saiba mais sobre o Astra Trident

Saiba mais sobre o Astra Trident

O Astra Trident é um projeto de código aberto com suporte total mantido pela NetApp como parte "[Família de produtos Astra](#)" do . Ele foi desenvolvido para ajudar você a atender às demandas de persistência da sua aplicação em contêineres usando interfaces padrão do setor, como a Container Storage Interface (CSI).

O que é o Astra?

Com o Astra, é mais fácil para as empresas gerenciar, proteger e mover workloads em contêineres com muitos dados executados no Kubernetes dentro e entre nuvens públicas e no local.

O Astra provisiona e fornece storage de contêiner persistente com base no Astra Trident. Ele também oferece recursos avançados de gerenciamento de dados com reconhecimento de aplicações, como snapshot, backup e restauração, logs de atividade e clonagem ativa para proteção de dados, recuperação de desastres/dados, auditoria de dados e casos de uso de migração para workloads Kubernetes.

Saiba mais "[Inscreva-se para uma avaliação gratuita](#)" sobre o .

O que é o Astra Trident?

O Astra Trident permite o consumo e gerenciamento de recursos de storage em todas as plataformas de storage populares da NetApp, na nuvem pública ou no local, incluindo ONTAP (AFF, FAS, Select, nuvem, Amazon FSX for NetApp ONTAP), software Element (NetApp HCI, SolidFire), serviço Azure NetApp Files e Cloud Volumes Service no Google Cloud.

O Astra Trident é um orquestrador de storage dinâmico e em conformidade com "[Kubernetes](#)" a Container Storage Interface (CSI) que se integra nativamente ao . O Astra Trident é executado como um único Pod de controladora e um pod de nó em cada nó de trabalho no cluster. "[A arquitetura do Astra Trident](#)" Consulte para obter detalhes.

O Astra Trident também fornece integração direta com o ecossistema Docker para plataformas de storage NetApp. O plug-in de volume do Docker do NetApp (nDVP) dá suporte ao provisionamento e gerenciamento de recursos de storage da plataforma de storage para hosts do Docker. "[Implante o Astra Trident para Docker](#)" Consulte para obter detalhes.



Se esta for a primeira vez que usa o Kubernetes, você deve se familiarizar com o "[Conceitos e ferramentas do Kubernetes](#)".

Faça o test drive do Astra Trident

Para fazer um test drive, solicite acesso ao "Deploy and Clone Persistent Storage for containerized workloads" "[Unidade de teste do NetApp](#)" usando uma imagem de laboratório pronta para uso. A unidade de teste fornece um ambiente sandbox com um cluster de Kubernetes de três nós e o Astra Trident instalado e configurado. Esta é uma ótima maneira de se familiarizar com o Astra Trident e explorar seus recursos.

Outra opção é a "[Guia de Instalação do kubeadm](#)" fornecida pelo Kubernetes.



Não use um cluster do Kubernetes criado usando essas instruções em um ambiente de produção. Use os guias de implantação de produção fornecidos pela distribuição para clusters prontos para produção.

Para mais informações

- ["Família de produtos NetApp Astra"](#)
- ["Documentação do Astra Control Service"](#)
- ["Documentação do Astra Control Center"](#)
- ["Documentação da API Astra"](#)

A arquitetura do Astra Trident

O Astra Trident é executado como um único Pod de controladora e um pod de nó em cada nó de trabalho no cluster. O pod de nó deve estar em execução em qualquer host onde você queira potencialmente montar um volume Astra Trident.

Compreensão dos pods dos nós e dos pods do controlador

O Astra Trident é implantado como [Pod do controlador Tridentum](#) único e um ou mais [Pods do nó Trident](#) no cluster Kubernetes e usa contentores padrão do Kubernetes [CSI Sidecar](#) para simplificar a implantação de plug-ins do CSI. ["Kubernetes CSI Sidecar contêineres"](#) São mantidos pela comunidade do Kubernetes Storage.

Kubernetes ["seletores de nós"](#) e ["tolerações e taints"](#) são usados para restringir um pod a ser executado em um nó específico ou preferencial. Você pode configurar seletores de nós e tolerâncias para pods de nó e controlador durante a instalação do Astra Trident.

- O plugin controlador lida com o provisionamento e gerenciamento de volume, como snapshots e redimensionamento.
- O plug-in do nó manipula a conexão do armazenamento ao nó.

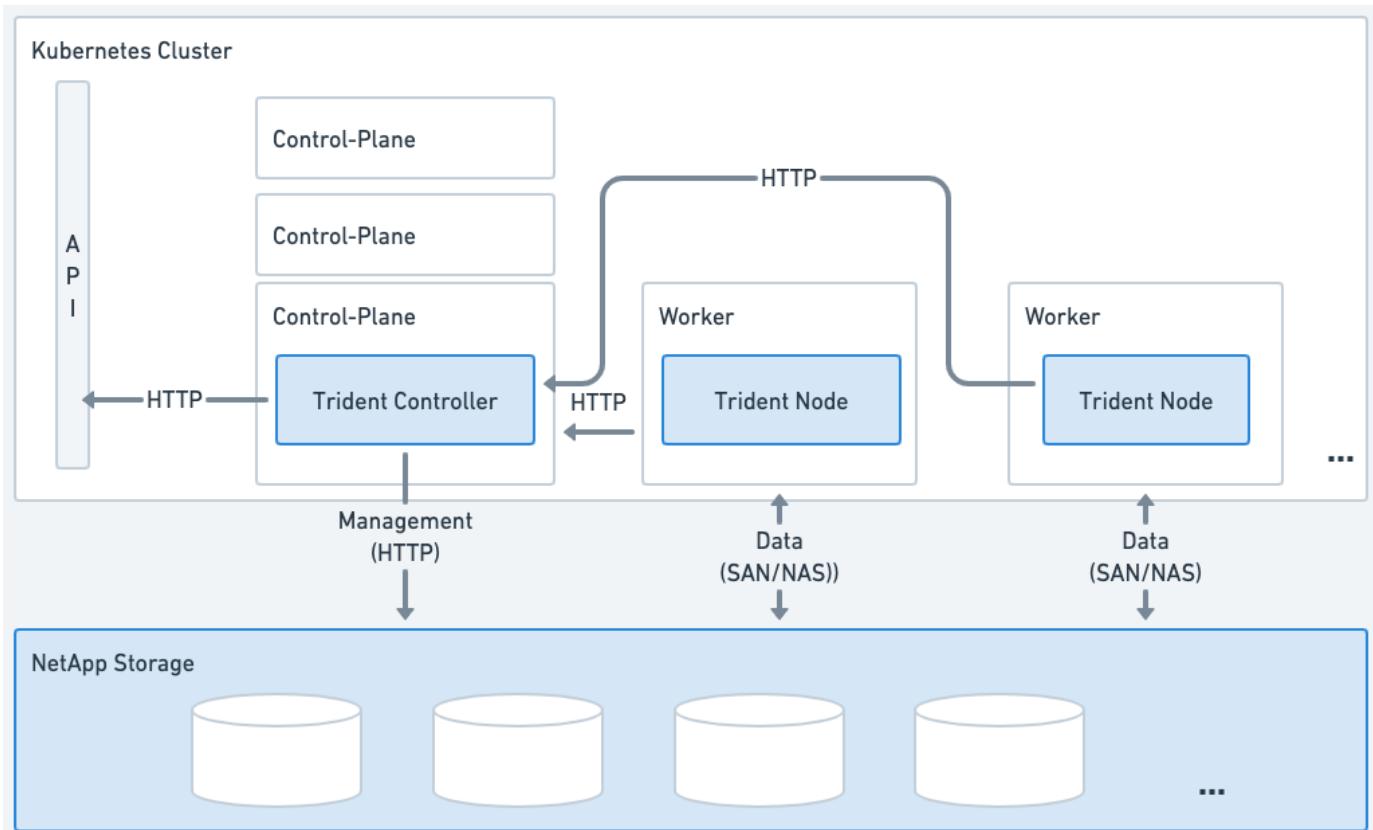


Figura 1. Astra Trident implantado no cluster Kubernetes

Pod do controlador Trident

O Pod do controlador Trident é um único Pod que executa o plug-in do controlador CSI.

- Responsável pelo provisionamento e gerenciamento de volumes no storage NetApp
- Gerenciado por uma implantação do Kubernetes
- Pode ser executado no plano de controle ou nos nós de trabalho, dependendo dos parâmetros de instalação.

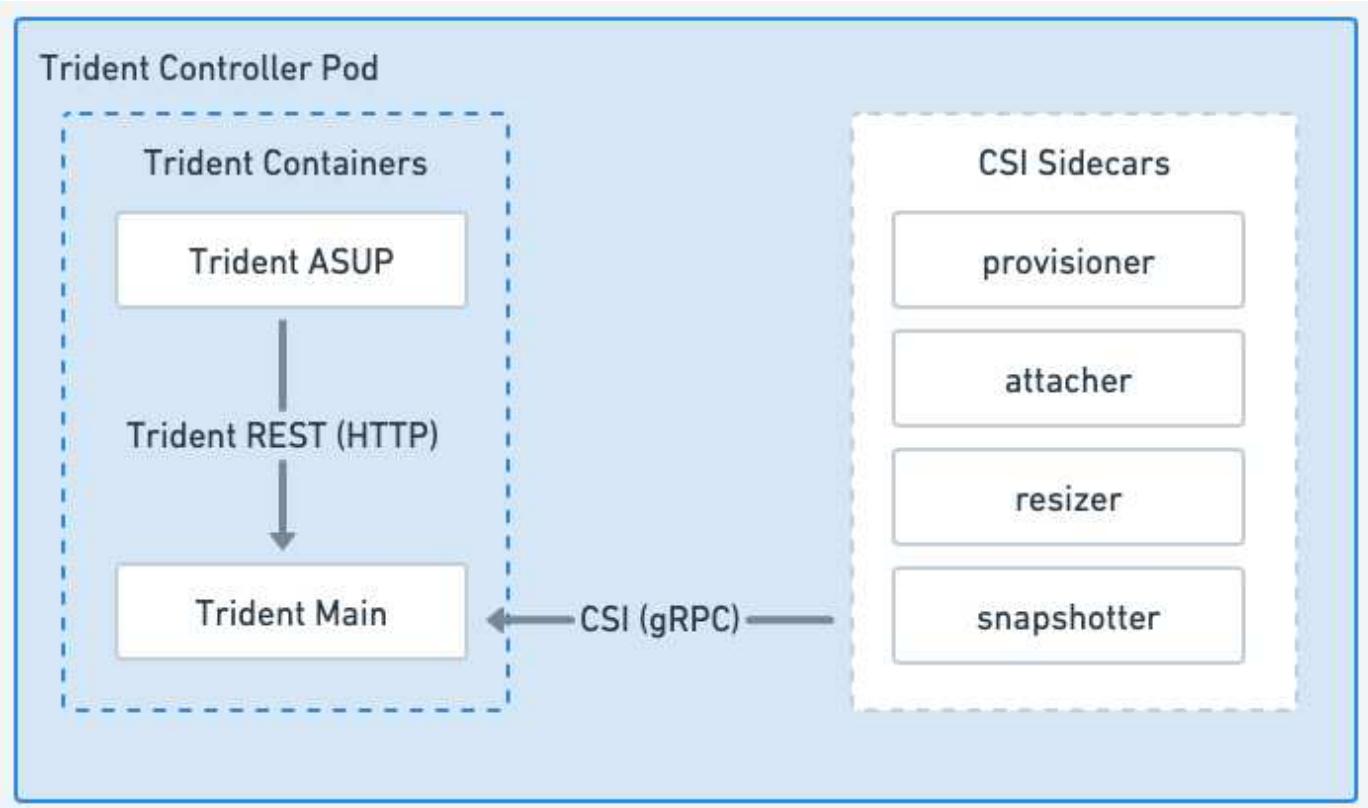


Figura 2. Diagrama do pod do controlador Trident

Pods do nó Trident

Os pods de nó Trident são pods privilegiados que executam o plug-in do nó CSI.

- Responsável pela montagem e desmontagem do armazenamento dos pods em execução no host
- Gerenciado por um DaemonSet Kubernetes
- Deve ser executado em qualquer nó que montar o storage NetApp

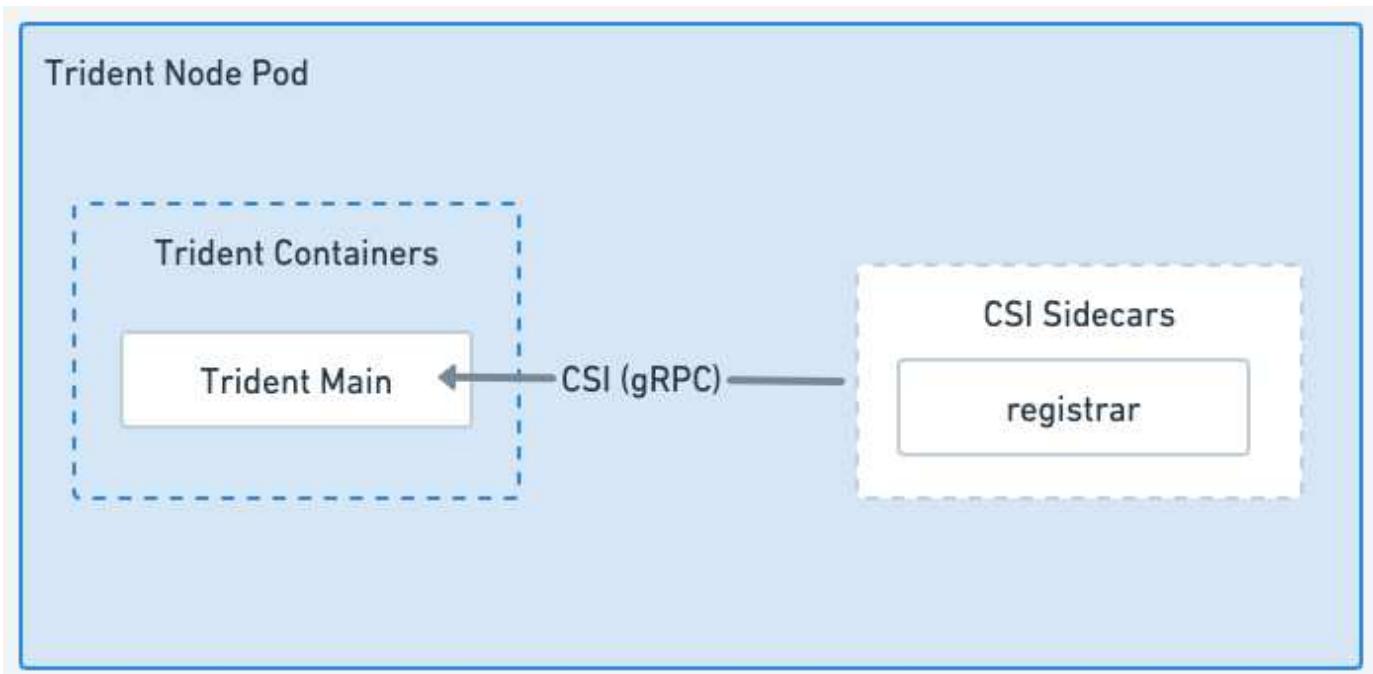


Figura 3. Diagrama do pod do nó Trident

Arquiteturas de cluster Kubernetes compatíveis

O Astra Trident é compatível com as seguintes arquiteturas de Kubernetes:

Arquiteturas de cluster do Kubernetes	Suportado	Instalação predefinida
Único mestre, computação	Sim	Sim
Mestre múltiplo, computação	Sim	Sim
Mestre etcd, , computação	Sim	Sim
Mestre, infraestrutura, computação	Sim	Sim

Conceitos

Provisionamento

O provisionamento no Astra Trident tem duas fases primárias. A primeira fase associa uma classe de armazenamento ao conjunto de conjuntos de armazenamento de back-end adequados e ocorre como uma preparação necessária antes do provisionamento. A segunda fase inclui a própria criação de volume e requer a escolha de um pool de armazenamento daqueles associados à classe de armazenamento do volume pendente.

Associação de classe de armazenamento

A associação de pools de storage de back-end a uma classe de armazenamento depende dos atributos solicitados da classe de armazenamento e `storagePools` das listas, `additionalStoragePools` e `excludeStoragePools`. Quando você cria uma classe de storage, o Trident compara os atributos e pools

oferecidos por cada um de seus back-ends aos solicitados pela classe de storage. Se os atributos e o nome de um pool de storage corresponderem a todos os atributos e nomes de pool solicitados, o Astra Trident adicionará esse pool de storage ao conjunto de pools de storage adequados para essa classe de storage. Além disso, o Astra Trident adiciona todos os pools de storage listados na `additionalStoragePools` lista a esse conjunto, mesmo que seus atributos não atendam a todos ou a qualquer um dos atributos solicitados da classe de storage. Você deve usar a `excludeStoragePools` lista para substituir e remover pools de armazenamento de uso para uma classe de armazenamento. O Astra Trident executa um processo semelhante sempre que você adiciona um novo back-end, verificando se os pools de storage atendem às classes de storage existentes e removendo quaisquer que tenham sido marcados como excluídos.

Criação de volume

Em seguida, o Astra Trident usa as associações entre classes de storage e pools de storage para determinar onde provisionar volumes. Quando você cria um volume, o Astra Trident primeiro obtém o conjunto de pools de storage para a classe de storage desse volume e, se você especificar um protocolo para o volume, o Astra Trident removerá esses pools de storage que não podem fornecer o protocolo solicitado (por exemplo, um back-end NetApp HCI/SolidFire não poderá fornecer um volume baseado em arquivo enquanto um back-end do ONTAP nas não puder fornecer um volume baseado em bloco). O Astra Trident aleatoriza a ordem desse conjunto resultante, para facilitar uma distribuição uniforme de volumes e, em seguida, iterá-lo, tentando provisionar o volume em cada pool de storage por sua vez. Se for bem-sucedido em um, ele retorna com sucesso, registrando quaisquer falhas encontradas no processo. O Astra Trident retorna uma falha **somente se** falhar ao provisionamento em **todos** dos pools de storage disponíveis para a classe de storage e o protocolo solicitados.

Instantâneos de volume

Saiba mais sobre como o Astra Trident lida com a criação de snapshots de volume para seus drivers.

Saiba mais sobre a criação de instantâneos de volume

- Para os `ontap-nas` drivers, `ontap-san`, `gcp-cvs` e `azure-netapp-files` cada volume persistente (PV) é mapeado para um FlexVol. Como resultado, os snapshots de volume são criados como snapshots do NetApp. A tecnologia Snapshot da NetApp oferece mais estabilidade, escalabilidade, capacidade de recuperação e desempenho do que as tecnologias de snapshot da concorrência. Essas cópias snapshot são extremamente eficientes no tempo necessário para criá-las e no espaço de storage.
- Para `ontap-nas-flexgroup` o condutor, cada volume persistente (PV) é mapeado para um FlexGroup. Como resultado, os snapshots de volume são criados como snapshots do NetApp FlexGroup. A tecnologia Snapshot da NetApp oferece mais estabilidade, escalabilidade, capacidade de recuperação e desempenho do que as tecnologias de snapshot da concorrência. Essas cópias snapshot são extremamente eficientes no tempo necessário para criá-las e no espaço de storage.
- Para `ontap-san-economy` o driver, os PVS mapeiam para LUNs criados em FlexVols compartilhados. VolumeSnapshots de PVS são obtidos executando FlexClones do LUN associado. Com a tecnologia ONTAP FlexClone, é possível criar cópias dos maiores conjuntos de dados quase instantaneamente. As cópias compartilham blocos de dados com os pais, não consumindo storage, exceto o necessário para os metadados.
- Para `solidfire-san` o driver, cada PV mapeia para um LUN criado no cluster do software/NetApp HCI do NetApp Element. VolumeSnapshots são representados por instantâneos de elementos do LUN subjacente. Esses snapshots são cópias pontuais e ocupam apenas um pequeno espaço e recursos do sistema.
- Ao trabalhar com `ontap-nas` os drivers e `ontap-san`, os snapshots do ONTAP são cópias pontuais do FlexVol e consomem espaço no próprio FlexVol. Isso pode resultar na quantidade de espaço gravável no

volume para reduzir com o tempo, à medida que os snapshots são criados/programados. Uma maneira simples de lidar com isso é aumentar o volume redimensionando pelo Kubernetes. Outra opção é excluir snapshots que não são mais necessários. Quando um VolumeSnapshot criado por meio do Kubernetes é excluído, o Astra Trident excluirá o snapshot associado do ONTAP. Os snapshots do ONTAP que não foram criados pelo Kubernetes também podem ser excluídos.

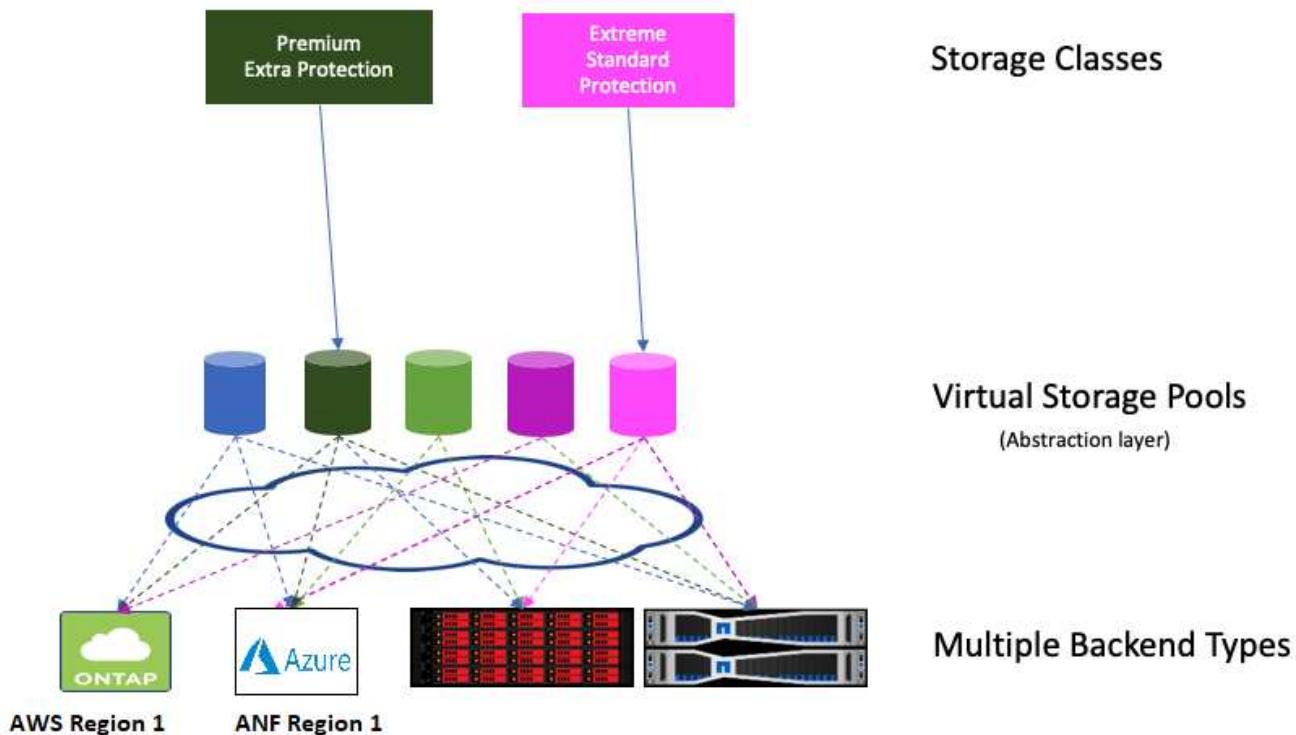
Com o Astra Trident, você pode usar o VolumeSnapshots para criar novos PVS a partir deles. A criação de PVS a partir desses snapshots é realizada usando a tecnologia FlexClone para backends ONTAP e CVS compatíveis. Ao criar um PV a partir de um instantâneo, o volume de apoio é um FlexClone do volume pai do instantâneo. O solidfire-san driver usa clones de volume do software Element para criar PVS a partir de snapshots. Aqui ele cria um clone a partir do snapshot do elemento.

Pools virtuais

Os pools virtuais fornecem uma camada de abstração entre os back-ends de storage do Astra Trident e o Kubernetes StorageClasses. Eles permitem que um administrador defina aspectos, como localização, desempenho e proteção para cada back-end de uma maneira comum e independente de back-end, sem StorageClass especificar qual backend físico, pool de back-end ou tipo de back-end usar para atender aos critérios desejados.

Saiba mais sobre pools virtuais

O administrador de storage pode definir pools virtuais em qualquer um dos backends do Astra Trident em um arquivo de definição JSON ou YAML.



Qualquer aspecto especificado fora da lista de pools virtuais é global para o back-end e se aplicará a todos os

pools virtuais, enquanto cada pool virtual pode especificar um ou mais aspectos individualmente (substituindo quaisquer aspectos globais de back-end).

- Ao definir pools virtuais, não tente reorganizar a ordem dos pools virtuais existentes em uma definição de back-end.
- Aconselhamos a não modificar atributos para um pool virtual existente. Você deve definir um novo pool virtual para fazer alterações.

A maioria dos aspectos são especificados em termos específicos de back-end. Fundamentalmente, os valores de aspecto não são expostos fora do driver do back-end e não estão disponíveis para correspondência em `StorageClasses`. em vez disso, o administrador define um ou mais rótulos para cada pool virtual. Cada rótulo é um par chave:valor, e os rótulos podem ser comuns em backends exclusivos. Assim como aspectos, os rótulos podem ser especificados por pool ou globais para o back-end. Ao contrário de aspectos, que têm nomes e valores predefinidos, o administrador tem total descrição para definir chaves de rótulo e valores conforme necessário. Por conveniência, os administradores de storage podem definir rótulos por pool virtual e volumes de grupo por rótulo.

A `StorageClass` identifica qual pool virtual usar fazendo referência aos rótulos dentro de um parâmetro seletor. Os seletores de pool virtual suportam os seguintes operadores:

Operador	Exemplo	O valor do rótulo de um pool deve:
=	desempenho superior	Correspondência
!=	performance! extrema	Não corresponde
in	localização em (leste, oeste)	Esteja no conjunto de valores
notin	notificação de desempenho (prata, bronze)	Não estar no conjunto de valores
<key>	proteção	Existe com qualquer valor
!<key>	!proteção	Não existe

Grupos de acesso de volume

Saiba mais sobre como o Astra Trident usa "[grupos de acesso de volume](#)".

 Ignore esta seção se você estiver usando CHAP, que é recomendado para simplificar o gerenciamento e evitar o limite de escala descrito abaixo. Além disso, se você estiver usando Astra Trident no modo CSI, você pode ignorar esta seção. O Astra Trident usa o CHAP quando instalado como um provisionador aprimorado de CSI.

Saiba mais sobre grupos de acesso de volume

O Astra Trident pode usar grupos de acesso a volumes para controlar o acesso aos volumes provisionados. Se o CHAP estiver desativado, ele espera encontrar um grupo de acesso chamado `trident`, a menos que você especifique um ou mais IDs de grupo de acesso na configuração.

Embora o Astra Trident associe novos volumes aos grupos de acesso configurados, ele não cria nem gerencia grupos de acesso por conta própria. Os grupos de acesso devem existir antes que o back-end de storage seja adicionado ao Astra Trident e precisam conter as IQNs iSCSI de todos os nós do cluster do Kubernetes que poderiam potencialmente montar os volumes provisionados por esse back-end. Na maioria das instalações, isso inclui cada nó de trabalho no cluster.

Para clusters de Kubernetes com mais de 64 nós, você deve usar vários grupos de acesso. Cada grupo de acesso pode conter até 64 IQNs e cada volume pode pertencer a quatro grupos de acesso. Com o máximo de quatro grupos de acesso configurados, qualquer nó em um cluster de até 256 nós de tamanho poderá acessar qualquer volume. Para obter os limites mais recentes dos grupos de acesso de volume, ["aqui"](#) consulte .

Se você estiver modificando a configuração de uma que esteja usando o grupo de acesso padrão `trident` para outra que também use outras, inclua a ID do `trident` grupo de acesso na lista.

Início rápido para Astra Trident

Você pode instalar o Astra Trident e começar a gerenciar recursos de storage em apenas algumas etapas. Antes de começar, reveja ["Requisitos do Astra Trident"](#) o .



Para Docker, ["Astra Trident para Docker"](#) consulte .

1

Instale o Astra Trident

O Astra Trident oferece vários métodos e modos de instalação otimizados para uma variedade de ambientes e organizações.

["Instale o Astra Trident"](#)

2

Prepare o nó de trabalho

Todos os nós de trabalho no cluster do Kubernetes precisam ser capazes de montar os volumes provisionados para os pods.

["Prepare o nó de trabalho"](#)

3

Crie um backend

Um back-end define a relação entre o Astra Trident e um sistema de storage. Ele diz ao Astra Trident como se comunicar com esse sistema de storage e como o Astra Trident deve provisionar volumes a partir dele.

["Configurar um back-end"](#) para o seu sistema de storage

4

Crie um Kubernetes StorageClass

O objeto Kubernetes StorageClass especifica o Astra Trident como o provisionador e permite que você crie uma classe de storage para provisionar volumes com atributos personalizáveis. O Astra Trident cria uma classe de storage correspondente para objetos Kubernetes que especificam o provisionador do Astra Trident.

["Crie uma classe de armazenamento"](#)

5

Provisionar um volume

Um *Persistentvolume* (PV) é um recurso de armazenamento físico provisionado pelo administrador de cluster em um cluster do Kubernetes. O *PersistentVolumeClaim* (PVC) é um pedido de acesso ao Persistentvolume

no cluster.

Crie um Persistentvolume (PV) e um PersistentVolumeClaim (PVC) que use o Kubernetes StorageClass configurado para solicitar acesso ao PV. Em seguida, pode montar o PV num pod.

["Provisionar um volume"](#)

O que se segue?

Agora você pode adicionar backends adicionais, gerenciar classes de armazenamento, gerenciar backends e executar operações de volume.

Requisitos

Antes de instalar o Astra Trident, você deve analisar esses requisitos gerais de sistema. Backends específicos podem ter requisitos adicionais.

Informações essenciais sobre o Astra Trident

Você deve ler as seguintes informações críticas sobre o Astra Trident.

** informações essenciais sobre o Astra Trident **

- O Kubernetes 1,27 agora é compatível com o Trident. Atualize o Trident antes de atualizar o Kubernetes.
- O Astra Trident reforça estritamente o uso de configuração multipathing em ambientes SAN, com um valor recomendado de `find_multipaths`: `no` no arquivo `multipath.conf`.

O uso de configuração não multipathing ou o uso `find_multipaths`: `yes` de ou `find_multipaths`: `smart` valor no arquivo `multipath.conf` resultará em falhas de montagem. A Trident recomenda o uso de `find_multipaths`: `no` desde a versão 21,07.

Frontens suportados (orquestradores)

O Astra Trident é compatível com vários mecanismos de contêiner e orquestradores, incluindo os seguintes:

- Anthos On-Prem (VMware) e Anthos em bare metal 1,12
- Kubernetes 1,22 - 1,27
- Mecanismo do Kubernetes do Mirantis 3,5
- OpenShift 4,10 - 4,13

O operador Trident é suportado com estas versões:

- Anthos On-Prem (VMware) e Anthos em bare metal 1,12
- Kubernetes 1,22 - 1,27
- OpenShift 4,10 - 4,13

O Astra Trident também trabalha com uma série de outras ofertas do Kubernetes totalmente gerenciadas e autogeridas, incluindo o Google Kubernetes Engine (GKE), o Amazon Elastic Kubernetes Services (EKS), o

Azure Kubernetes Service (AKS), o Rancher e o portfólio VMware Tanzu.



Antes de atualizar um cluster do Kubernetes do 1,24 para o 1,25 ou posterior que tenha o Astra Trident instalado, ["Atualize uma instalação do Helm"](#) consulte .

Backends suportados (armazenamento)

Para usar o Astra Trident, você precisa de um ou mais dos seguintes back-ends compatíveis:

- Amazon FSX para NetApp ONTAP
- Azure NetApp Files
- Cloud Volumes ONTAP
- Cloud Volumes Service para GCP
- FAS/AFF/Seleciona 9,5 ou posterior
- NetApp All SAN Array (ASA)
- Software NetApp HCI/Element 11 ou superior

Requisitos de recursos

A tabela abaixo resume os recursos disponíveis com esta versão do Astra Trident e as versões do Kubernetes compatíveis.

Recurso	Versão do Kubernetes	É necessário ter portões?
Astra Trident	1,22 - 1,27	Não
Instantâneos de volume	1,22 - 1,27	Não
PVC a partir de instantâneos de volume	1,22 - 1,27	Não
Redimensionamento iSCSI PV	1,22 - 1,27	Não
ONTAP bidirecional CHAP	1,22 - 1,27	Não
Políticas de exportação dinâmica	1,22 - 1,27	Não
Operador Trident	1,22 - 1,27	Não
Topologia de CSI	1,22 - 1,27	Não

Sistemas operacionais de host testados

Embora o Astra Trident não seja oficialmente compatível com sistemas operacionais específicos, sabe-se que os seguintes itens funcionam:

- Versões do RedHat CoreOS (RHCOS) suportadas pela OpenShift Container Platform (AMD64 e ARM64)

- RHEL 8 OU SUPERIOR (AMD64 E ARM64)
- Ubuntu 22,04 ou posterior (AMD64 e ARM64)
- Windows Server 2019 (AMD64 bits)

Por padrão, o Astra Trident é executado em um contentor e, portanto, será executado em qualquer trabalhador Linux. No entanto, esses funcionários precisam ser capazes de montar os volumes que o Astra Trident fornece usando o cliente NFS padrão ou iniciador iSCSI, dependendo dos backends que você está usando.

O `tridentctl` utilitário também é executado em qualquer uma dessas distribuições do Linux.

Configuração de host

Todos os nós de trabalho no cluster do Kubernetes precisam ser capazes de montar os volumes provisionados para os pods. Para preparar os nós de trabalho, você deve instalar ferramentas NFS ou iSCSI com base na seleção de driver.

["Prepare o nó de trabalho"](#)

Configuração do sistema de storage

O Astra Trident pode exigir alterações em um sistema de storage antes que uma configuração de back-end o use.

["Configurar backends"](#)

Portas Astra Trident

O Astra Trident requer acesso a portas específicas para comunicação.

["Portas Astra Trident"](#)

Imagens de contêineres e versões correspondentes do Kubernetes

Para instalações com conexão de ar, a lista a seguir é uma referência das imagens de contêiner necessárias para instalar o Astra Trident. Use o `tridentctl images` comando para verificar a lista de imagens de contentor necessárias.

Versão do Kubernetes	Imagen do recipiente
v1.22.0	<ul style="list-style-type: none"> • docker.io/NetApp/Trident:23.07.1 • docker.io/NetApp/Trident-AutoSupport:23.07 • provisionador do registry.k8s.io/sig-storage/csi:v3.5.0 • registry.k8s.io/sig-storage/csi-attacher:v4.3.0 • registry.k8s.io/sig-storage/csi-resizer:v1.8.0 • registry.k8s.io/sig-storage/csi-snapshotter:v6.2.2 • registry.k8s.io/sig-storage/csi-node-driver-registrador:v2.8.0 • docker.io/NetApp/Trident-operador:23.07.1 (opcional)
v1.23.0	<ul style="list-style-type: none"> • docker.io/NetApp/Trident:23.07.1 • docker.io/NetApp/Trident-AutoSupport:23.07 • provisionador do registry.k8s.io/sig-storage/csi:v3.5.0 • registry.k8s.io/sig-storage/csi-attacher:v4.3.0 • registry.k8s.io/sig-storage/csi-resizer:v1.8.0 • registry.k8s.io/sig-storage/csi-snapshotter:v6.2.2 • registry.k8s.io/sig-storage/csi-node-driver-registrador:v2.8.0 • docker.io/NetApp/Trident-operador:23.07.1 (opcional)
v1.24.0	<ul style="list-style-type: none"> • docker.io/NetApp/Trident:23.07.1 • docker.io/NetApp/Trident-AutoSupport:23.07 • provisionador do registry.k8s.io/sig-storage/csi:v3.5.0 • registry.k8s.io/sig-storage/csi-attacher:v4.3.0 • registry.k8s.io/sig-storage/csi-resizer:v1.8.0 • registry.k8s.io/sig-storage/csi-snapshotter:v6.2.2 • registry.k8s.io/sig-storage/csi-node-driver-registrador:v2.8.0 • docker.io/NetApp/Trident-operador:23.07.1 (opcional)

Versão do Kubernetes	Imagen do recipiente
v1.25.0	<ul style="list-style-type: none"> • docker.io/NetApp/Trident:23.07.1 • docker.io/NetApp/Trident-AutoSupport:23.07 • provisionador do registry.k8s.io/sig-storage/csi:v3.5.0 • registry.k8s.io/sig-storage/csi-attacher:v4.3.0 • registry.k8s.io/sig-storage/csi-resizer:v1.8.0 • registry.k8s.io/sig-storage/csi-snapshotter:v6.2.2 • registry.k8s.io/sig-storage/csi-node-driver-registrador:v2.8.0 • docker.io/NetApp/Trident-operador:23.07.1 (opcional)
v1.26.0	<ul style="list-style-type: none"> • docker.io/NetApp/Trident:23.07.1 • docker.io/NetApp/Trident-AutoSupport:23.07 • provisionador do registry.k8s.io/sig-storage/csi:v3.5.0 • registry.k8s.io/sig-storage/csi-attacher:v4.3.0 • registry.k8s.io/sig-storage/csi-resizer:v1.8.0 • registry.k8s.io/sig-storage/csi-snapshotter:v6.2.2 • registry.k8s.io/sig-storage/csi-node-driver-registrador:v2.8.0 • docker.io/NetApp/Trident-operador:23.07.1 (opcional)
v1.27.0	<ul style="list-style-type: none"> • docker.io/NetApp/Trident:23.07.1 • docker.io/NetApp/Trident-AutoSupport:23.07 • provisionador do registry.k8s.io/sig-storage/csi:v3.5.0 • registry.k8s.io/sig-storage/csi-attacher:v4.3.0 • registry.k8s.io/sig-storage/csi-resizer:v1.8.0 • registry.k8s.io/sig-storage/csi-snapshotter:v6.2.2 • registry.k8s.io/sig-storage/csi-node-driver-registrador:v2.8.0 • docker.io/NetApp/Trident-operador:23.07.1 (opcional)

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 DESTE DOCUMENTO. 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 SOBRESSAENTES; 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 DESTE SOFTWARE, 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.