



Comece agora

Astra Control Center

NetApp
August 11, 2025

Índice

Comece agora	1
Saiba mais sobre o Astra Control	1
Caraterísticas	1
Modelos de implantação	1
Como funciona o Astra Control Service	2
Como funciona o Astra Control Center	3
Para mais informações	4
Requisitos do Astra Control Center	4
Ambientes de Kubernetes do cluster de host compatíveis	4
Requisitos de recursos de cluster de host	5
Requisitos do Astra Trident	5
Back-ends de armazenamento	6
Registo de imagens	6
Licença do Astra Control Center	7
Requisitos de rede	7
Entrada para clusters do Kubernetes no local	8
Navegadores da Web suportados	8
Requisitos adicionais para clusters de aplicações	9
O que vem a seguir	9
Início rápido para Astra Control Center	9
Para mais informações	10
Visão geral da instalação	10
Instale o Astra Control Center usando o processo padrão	11
Instale o Astra Control Center usando o OpenShift OperatorHub	48
Instalar o Astra Control Center com um back-end de storage do Cloud Volumes ONTAP	56
Configure o Astra Control Center após a instalação	72
Configure o Astra Control Center	78
Adicione uma licença para o Astra Control Center	78
Prepare seu ambiente para gerenciamento de clusters com o Astra Control	79
Adicionar cluster	90
Ativar a autenticação no back-end de storage do ONTAP	91
Adicionar um back-end de storage	98
Adicione um balde	99
O que se segue?	100
Perguntas mais frequentes para o Astra Control Center	101
Visão geral	101
Acesso ao Astra Control Center	101
Licenciamento	101
Registrando clusters do Kubernetes	102
Gerenciamento de aplicações	102
Operações de gerenciamento de dados	103

Comece agora

Saiba mais sobre o Astra Control

O Astra Control é uma solução de gerenciamento de ciclo de vida de dados de aplicações Kubernetes que simplifica as operações de aplicações com estado monitorado. Proteja, faça backup, replique e migre workloads do Kubernetes com facilidade e crie clones de aplicações em funcionamento instantaneamente.

Caraterísticas

O Astra Control oferece funcionalidades essenciais para o gerenciamento do ciclo de vida dos dados da aplicação Kubernetes:

- Gerencie automaticamente o storage persistente
- Crie backups e snapshots sob demanda com reconhecimento de aplicações
- Automatizar operações de backup e snapshot orientadas por políticas
- Migrar aplicações e dados entre clusters do Kubernetes
- Replique aplicações para um sistema remoto usando a tecnologia NetApp SnapMirror (Astra Control Center)
- Clonar aplicações da preparação para a produção
- Visualize a integridade e o status de proteção da aplicação
- Trabalhe com uma IU da Web ou uma API para implementar seus fluxos de trabalho de backup e migração

Modelos de implantação

O Astra Control está disponível em dois modelos de implantação:

- **Astra Control Service:** Um serviço gerenciado pelo NetApp que fornece gerenciamento de dados com reconhecimento de aplicações dos clusters do Kubernetes em vários ambientes de fornecedores de nuvem e clusters do Kubernetes autogerenciados.
- **Astra Control Center:** Software autogerenciado que oferece gerenciamento de dados com reconhecimento de aplicações dos clusters Kubernetes executados em seu ambiente local. O Astra Control Center também pode ser instalado em vários ambientes de fornecedor de nuvem com um back-end de storage da NetApp Cloud Volumes ONTAP.

	Astra Control Service	Astra Control Center
Como é oferecido?	Como um serviço de nuvem totalmente gerenciado da NetApp	Como software que você pode baixar, instalar e gerenciar
Onde está hospedado?	Em uma nuvem pública de escolha da NetApp	No seu próprio cluster Kubernetes
Como é atualizado?	Gerenciado por NetApp	Você gerencia quaisquer atualizações

	Astra Control Service	Astra Control Center
Quais são os backends de armazenamento suportados?	- Amazon Web Services: - Amazon EBS - Amazon FSX para NetApp ONTAP "Cloud Volumes ONTAP" - Google Cloud: - Persistent Disk do Google - NetApp Cloud Volumes Service "Cloud Volumes ONTAP" - Microsoft Azure: - Discos gerenciados do Azure - Azure NetApp Files "Cloud Volumes ONTAP" - Clusters autogerenciados: - Amazon EBS - Persistent Disk do Google - Discos gerenciados do Azure "Cloud Volumes ONTAP"	- Sistemas NetApp ONTAP AFF e FAS "Cloud Volumes ONTAP"

Como funciona o Astra Control Service

O Astra Control Service é um serviço de nuvem gerenciado pela NetApp que está sempre ativo e atualizado com as funcionalidades mais recentes. Ele utiliza vários componentes para habilitar o gerenciamento do ciclo de vida dos dados das aplicações.

Em um alto nível, o Astra Control Service funciona assim:

- Você começa a usar o Astra Control Service configurando seu fornecedor de nuvem e registrando-se em uma conta Astra.
 - Para clusters GKE, o Astra Control Service usa "NetApp Cloud Volumes Service para Google Cloud" ou discos persistentes do Google como back-end de storage para volumes persistentes.
 - Para clusters AKS, o Astra Control Service usa "Azure NetApp Files" ou discos gerenciados do Azure como o back-end de storage para seus volumes persistentes.
 - Para clusters do Amazon EKS, o Astra Control Service usa "Amazon Elastic Block Store" ou "Amazon FSX para NetApp ONTAP" como back-end de storage para volumes persistentes.
- Você adiciona sua primeira computação do Kubernetes ao Astra Control Service. Em seguida, o Astra Control Service faz o seguinte:
 - Cria um armazenamento de objetos na sua conta de fornecedor de nuvem, que é onde as cópias de backup são armazenadas.

No Azure, o Astra Control Service também cria um grupo de recursos, uma conta de storage e chaves para o contêiner de Blob.

- Cria uma nova função de administrador e conta de serviço do Kubernetes no cluster.
- Usa essa nova função de administrador para instalar "[Astra Trident](#)" no cluster e criar uma ou mais classes de armazenamento.
- Se você usar uma oferta de storage de serviço de nuvem da NetApp como back-end de storage, o Astra Control Service usará o Astra Trident para provisionar volumes persistentes para suas aplicações. Se você usar os discos gerenciados do Amazon EBS ou Azure como back-end de armazenamento, precisará instalar um driver CSI específico do provedor. As instruções de instalação são fornecidas na "[Configurar o Amazon Web Services](#)" e "[Configurar o Microsoft Azure com discos gerenciados do Azure](#)".
- Neste ponto, você pode adicionar aplicativos ao cluster. Volumes persistentes serão provisionados na nova classe de armazenamento padrão.
- Depois, você usa o Astra Control Service para gerenciar essas aplicações e começar a criar snapshots, backups e clones.

O Plano Gratuito do Astra Control permite gerenciar até 10 namespaces em sua conta. Se você quiser gerenciar mais de 10, precisará configurar o faturamento atualizando do Plano Gratuito para o Plano Premium.

Como funciona o Astra Control Center

Astra Control Center é executado localmente em sua própria nuvem privada.

O Astra Control Center é compatível com clusters Kubernetes com classe de storage baseado no Astra Trident com um back-end ONTAP 9.5 ou superior.

Em um ambiente conectado à nuvem, o Astra Control Center usa o Cloud Insights para fornecer monitoramento avançado e telemetria. Na ausência de uma conexão Cloud Insights, monitoramento e telemetria limitados (7 dias de métricas) estão disponíveis no Centro de Controle Astra e também exportados para ferramentas de monitoramento nativas do Kubernetes (como Prometheus e Grafana) por meio de pontos finais de métricas abertas.

O Astra Control Center é totalmente integrado ao ecossistema de consultores digitais da AutoSupport e Active IQ (também conhecido como consultor digital) para fornecer aos usuários e ao suporte da NetApp informações de solução de problemas e uso.

Você pode experimentar o Astra Control Center usando uma licença de avaliação incorporada de 90 dias. Enquanto você está avaliando o Astra Control Center, pode obter suporte por meio de e-mail e opções da comunidade. Além disso, você tem acesso a artigos e documentação da base de conhecimento a partir do painel de suporte do produto.

Para instalar e usar o Astra Control Center, você precisará atender a determinados "[requisitos](#)".

Em um alto nível, o Astra Control Center funciona assim:

- Você instala o Astra Control Center em seu ambiente local. Saiba mais sobre como "[Instale o Astra Control Center](#)".
- Você conclui algumas tarefas de configuração, como estas:
 - Configure o licenciamento.
 - Adicione o primeiro cluster.
 - Adicione o back-end de storage descoberto quando você adicionou o cluster.

- Adicione um bucket do armazenamento de objetos que armazenará os backups do aplicativo.

Saiba mais sobre como ["Configure o Astra Control Center"](#) .

Você pode adicionar aplicativos ao cluster. Ou, se você já tiver algumas aplicações no cluster sendo gerenciado, poderá usar o Astra Control Center para gerenciá-las. Depois, use o Astra Control Center para criar snapshots, backups, clones e relacionamentos de replicação.

Para mais informações

- ["Documentação do Astra Control Service"](#)
- ["Documentação do Astra Control Center"](#)
- ["Documentação do Astra Trident"](#)
- ["Use a API Astra Control"](#)
- ["Documentação do Cloud Insights"](#)
- ["Documentação do ONTAP"](#)

Requisitos do Astra Control Center

Comece verificando a prontidão do seu ambiente operacional, clusters de aplicativos, aplicativos, licenças e navegador da Web. Garanta que seu ambiente atenda a esses requisitos para implantar e operar o Astra Control Center.

- [Ambientes de Kubernetes do cluster de host compatíveis](#)
- [Requisitos de recursos de cluster de host](#)
- [Requisitos do Astra Trident](#)
- [Back-ends de armazenamento](#)
- [Registo de imagens](#)
- [Licença do Astra Control Center](#)
- [Licenças ONTAP](#)
- [Requisitos de rede](#)
- [Entrada para clusters do Kubernetes no local](#)
- [Navegadores da Web suportados](#)
- [Requisitos adicionais para clusters de aplicações](#)

Ambientes de Kubernetes do cluster de host compatíveis

O Astra Control Center foi validado com os seguintes ambientes de host do Kubernetes:



Garantir que o ambiente do Kubernetes que você escolher hospedar o Astra Control Center atenda aos requisitos básicos de recursos descritos na documentação oficial do ambiente.

Distribuição do Kubernetes no cluster de host	Versões suportadas
Serviço Kubernetes do Azure no Azure Stack HCI	Azure Stack HCI 21H2 e 22H2 com AKS 1,24.x e 1,25.x
Google Anthos	1,14 a 1,16 (Requisitos de entrada do Google Anthos consulte)
Kubernetes (upstream)	1,25 a 1,27 (o Astra Trident 22,10 ou mais recente é necessário para Kubernetes 1,25 ou mais recente)
Rancher Kubernetes Engine (RKE)	RKE 1,3 com Rancher Manager 2,6 RKE 1,4 com Rancher Manager 2,7 RKE 2 (v1,24.x) com Rancher 2,6 RKE 2 (v1,25.x) com Rancher 2,7
Red Hat OpenShift Container Platform	4,11 a 4,13

Requisitos de recursos de cluster de host

O Astra Control Center requer os seguintes recursos, além dos requisitos de recursos do ambiente:



Esses requisitos presumem que o Astra Control Center é a única aplicação em execução no ambiente operacional. Se o ambiente estiver executando aplicativos adicionais, ajuste esses requisitos mínimos de acordo.

- *** Extensões de CPU*:** As CPUs em todos os nós do ambiente de hospedagem devem ter extensões AVX ativadas.
- **Worker Nodes:** Pelo menos 3 worker node total, com 4 núcleos de CPU e 12GB GB de RAM cada

Requisitos do Astra Trident

Garanta o atendimento aos seguintes requisitos do Astra Trident específicos para as necessidades do seu ambiente:

- **Versão mínima para uso com o Astra Control Center:** Astra Trident 22,10 ou mais recente instalado e configurado.
- **Replicação SnapMirror:** Astra Trident 22,10 ou mais recente instalado para replicação de aplicações baseada em SnapMirror.
- **Para suporte a Kubernetes 1,25 ou mais recente:** Astra Trident 22,10 ou mais recente instalado para Kubernetes 1,25 ou clusters mais recentes (é necessário atualizar para Astra Trident 22,10 antes da atualização para Kubernetes 1,25 ou mais recente)
- **Configuração do ONTAP com o Astra Trident:**
 - **Storage class:** Configure pelo menos uma classe de armazenamento Astra Trident no cluster. Se uma classe de armazenamento padrão estiver configurada, verifique se ela é a única classe de armazenamento com a designação padrão.
 - **Drivers de armazenamento e nós de trabalho:** Certifique-se de que os nós de trabalho no cluster estejam configurados com os drivers de armazenamento apropriados para que os pods possam interagir com o armazenamento de back-end. O Astra Control Center é compatível com os seguintes drivers ONTAP fornecidos pelo Astra Trident:
 - `ontap-nas`
 - `ontap-san`

- `ontap-san-economy` (a replicação de aplicativos não está disponível com esse tipo de classe de armazenamento)
- `ontap-nas-economy` (snapshots, políticas de replicação e políticas de proteção não estão disponíveis com esse tipo de classe de storage)

Back-ends de armazenamento

Certifique-se de ter um back-end compatível com capacidade suficiente.

- **Capacidade de back-end de armazenamento necessária:** Pelo menos 500GB GB disponíveis
- **Backends compatíveis:** O Astra Control Center é compatível com os seguintes back-ends de storage:
 - NetApp ONTAP 9 .8 ou sistemas AFF, FAS e ASA mais recentes
 - NetApp ONTAP Select 9,8 ou mais recente
 - NetApp Cloud Volumes ONTAP 9,8 ou mais recente
 - Longhorn 1.5.0 ou mais recente
 - Requer a criação manual de um objeto VolumeSnapshotClass. Consulte o ["Documentação de Longhorn"](#) para obter instruções.
 - NetApp MetroCluster
 - Os clusters do Kubernetes gerenciado precisam estar em uma configuração mais ampla.

Licenças ONTAP

Para usar o Astra Control Center, verifique se você tem as seguintes licenças do ONTAP, dependendo do que você precisa realizar:

- FlexClone
- SnapMirror: Opcional. Necessário apenas para replicação para sistemas remotos usando a tecnologia SnapMirror. Consulte a ["Informações de licença do SnapMirror"](#).
- Licença S3: Opcional. Necessário apenas para buckets do ONTAP S3

Para verificar se o sistema ONTAP tem as licenças necessárias, ["Gerenciar licenças do ONTAP"](#) consulte .

NetApp MetroCluster

Quando você usa o NetApp MetroCluster como um back-end de storage, precisa especificar uma opção de LIF de gerenciamento de SVM como uma opção de back-end no driver Astra Trident que você usa.

Para configurar o MetroCluster LIF, consulte a documentação do Astra Trident para obter mais informações sobre cada driver:

- ["SAN"](#)
- ["NAS"](#)

Registo de imagens

Você deve ter um Registro de imagem privada do Docker existente para o qual você pode enviar imagens de compilação do Astra Control Center. Você precisa fornecer o URL do Registro de imagens onde você vai carregar as imagens.

Licença do Astra Control Center

O Astra Control Center requer uma licença do Astra Control Center. Quando você instala o Astra Control Center, uma licença de avaliação incorporada de 90 dias para 4.800 unidades CPU já está ativada. Se você precisar de mais capacidade ou termos de avaliação diferentes ou quiser atualizar para uma licença completa, você pode obter uma licença de avaliação diferente ou uma licença completa da NetApp. Você precisa de uma licença para proteger seus aplicativos e dados.

Você pode experimentar o Astra Control Center inscrevendo-se para uma avaliação gratuita. Você pode se inscrever registrando ["aqui"](#).

Para configurar a licença, ["use uma licença de avaliação de 90 dias"](#) consulte a .

Para saber mais sobre como as licenças funcionam, ["Licenciamento"](#) consulte .

Requisitos de rede

Configure seu ambiente operacional para garantir que o Astra Control Center possa se comunicar corretamente. São necessárias as seguintes configurações de rede:

- **Endereço FQDN:** Você deve ter um endereço FQDN para o Astra Control Center.
- **Acesso à internet:** Você deve determinar se tem acesso externo à internet. Se não o fizer, algumas funcionalidades poderão ser limitadas, como receber dados de monitorização e métricas do NetApp Cloud Insights, ou enviar pacotes de suporte para o ["Site de suporte da NetApp"](#).
- **Acesso à porta:** O ambiente operacional que hospeda o Astra Control Center se comunica usando as seguintes portas TCP. Você deve garantir que essas portas sejam permitidas por meio de firewalls e configurar firewalls para permitir qualquer tráfego de saída HTTPS proveniente da rede Astra. Algumas portas exigem conectividade entre o ambiente que hospeda o Astra Control Center e cada cluster gerenciado (observado quando aplicável).



É possível implantar o Astra Control Center em um cluster de Kubernetes de duas stack e o Astra Control Center pode gerenciar aplicações e back-ends de storage configurados para operação de duas stack. Para obter mais informações sobre os requisitos de cluster de pilha dupla, consulte o ["Documentação do Kubernetes"](#).

Fonte	Destino	Porta	Protocolo	Finalidade
PC do cliente	Astra Control Center	443	HTTPS	Acesso de IU / API - garanta que essa porta esteja aberta de ambas as maneiras entre o cluster que hospeda o Astra Control Center e cada cluster gerenciado

Fonte	Destino	Porta	Protocolo	Finalidade
Consumidor de métricas	Nó de trabalho do Astra Control Center	9090	HTTPS	Comunicação de dados de métricas - garanta que cada cluster gerenciado possa acessar essa porta no cluster que hospeda o Astra Control Center (comunicação bidirecional necessária)
Astra Control Center	Serviço Cloud Insights hospedado (https://www.netapp.com/cloud-services/cloud-insights/)	443	HTTPS	Comunicação Cloud Insights
Astra Control Center	Fornecedor de bucket de storage do Amazon S3	443	HTTPS	Comunicação de armazenamento Amazon S3
Astra Control Center	NetApp AutoSupport (https://support.netapp.com)	443	HTTPS	Comunicação NetApp AutoSupport

Entrada para clusters do Kubernetes no local

Você pode escolher o tipo de entrada de rede que o Astra Control Center usa. Por padrão, o Astra Control Center implanta o gateway Astra Control Center (Service/traefik) como um recurso em todo o cluster. O Astra Control Center também é compatível com o uso de um balanceador de carga de serviço, se permitido no seu ambiente. Se você preferir usar um balanceador de carga de serviço e ainda não tiver um configurado, você pode usar o balanceador de carga MetalLB para atribuir automaticamente um endereço IP externo ao serviço. Na configuração do servidor DNS interno, você deve apontar o nome DNS escolhido para o Astra Control Center para o endereço IP com balanceamento de carga.



O balanceador de carga deve usar um endereço IP localizado na mesma sub-rede que os endereços IP do nó de trabalho do Astra Control Center.

Para obter mais informações, "[Configure a entrada para o balanceamento de carga](#)" consulte .

Requisitos de entrada do Google Anthos

Ao hospedar o Astra Control Center em um cluster do Google Anthos, observe que o Google Anthos inclui o balanceador de carga MetalLB e o serviço de ingresso Istio por padrão, permitindo que você simplesmente use os recursos genéricos de entrada do Astra Control Center durante a instalação. "[Configurar o Astra Control Center](#)" Consulte para obter detalhes.

Navegadores da Web suportados

O Astra Control Center suporta versões recentes do Firefox, Safari e Chrome com uma resolução mínima de 1280 x 720.

Requisitos adicionais para clusters de aplicações

Tenha em mente esses requisitos se você planeja usar esses recursos do Astra Control Center:

- * Requisitos de cluster de aplicativos*: ["Requisitos de gerenciamento de clusters"](#)
 - **Requisitos de aplicação gerenciada:** ["Requisitos de gerenciamento de aplicativos"](#)
 - **Requisitos adicionais para replicação de aplicativos:** ["Pré-requisitos de replicação"](#)

O que vem a seguir

Veja a ["início rápido"](#) visão geral.

Início rápido para Astra Control Center

Aqui está uma visão geral das etapas necessárias para começar a usar o Astra Control Center. Os links em cada etapa levam você a uma página que fornece mais detalhes.

1

Analisar os requisitos do cluster do Kubernetes

Certifique-se de que seu ambiente atenda a esses requisitos:

Cluster do Kubernetes

- ["Certifique-se de que o cluster de host atenda aos requisitos do ambiente operacional"](#)
- ["Configurar o ingresso para balanceamento de carga de clusters do Kubernetes no local"](#)

Integração de armazenamento

- ["Garanta que seu ambiente inclua a versão compatível com o Astra Trident"](#)
- ["Prepare os nós de trabalho"](#)
- ["Configurar o back-end de storage do Astra Trident"](#)
- ["Configurar as classes de storage do Astra Trident"](#)
- ["Instalar a controladora Snapshot de volume Astra Trident"](#)
- ["Crie uma classe de instantâneo de volume"](#)

Credenciais ONTAP

- ["Configurar credenciais do ONTAP"](#)

2

Baixe e instale o Astra Control Center

Conclua estas tarefas de instalação:

- ["Faça download do Centro de Controle Astra na página de downloads do site de suporte da NetApp"](#)
- Obtenha o ficheiro de licença NetApp:
 - Se você estiver avaliando o Astra Control Center, uma licença de avaliação incorporada já estará incluída

- ["Se você já comprou o Astra Control Center, gere seu arquivo de licença"](#)

- ["Instale o Astra Control Center"](#)
- ["Execute etapas de configuração opcionais adicionais"](#)

3

Conclua algumas tarefas de configuração inicial

Conclua algumas tarefas básicas para começar:

- ["Adicione uma licença"](#)
- ["Prepare seu ambiente para o gerenciamento de clusters"](#)
- ["Adicione um cluster"](#)
- ["Adicionar um back-end de storage"](#)
- ["Adicione um balde"](#)

4

Use o Astra Control Center

Depois de concluir a configuração do Astra Control Center, use a IU do Astra Control ou a ["API Astra Control"](#) para começar a gerenciar e proteger aplicações:

- ["Gerir aplicações"](#): Defina recursos para gerenciar.
- ["Proteja aplicativos"](#): Configurar políticas de proteção e replicar, clonar e migrar aplicativos.
- ["Gerenciar contas"](#): Usuários, funções, LDAP, credenciais e muito mais.
- ["Opcionalmente, conete-se ao Cloud Insights"](#): Visualizar métricas sobre a integridade do seu sistema.

Para mais informações

- ["Use a API Astra Control"](#)
- ["Atualizar o Astra Control Center"](#)
- ["Obtenha ajuda com o Astra Control"](#)

Visão geral da instalação

Escolha e conclua um dos seguintes procedimentos de instalação do Astra Control Center:

- ["Instale o Astra Control Center usando o processo padrão"](#)
- ["\(Se você usar o Red Hat OpenShift\) instale o Astra Control Center usando o OpenShift OperatorHub"](#)
- ["Instalar o Astra Control Center com um back-end de storage do Cloud Volumes ONTAP"](#)

Dependendo do seu ambiente, pode haver configuração adicional necessária após a instalação do Astra Control Center:

- ["Configure o Astra Control Center após a instalação"](#)

Instale o Astra Control Center usando o processo padrão

Para instalar o Astra Control Center, faça o download do pacote de instalação no site de suporte da NetApp e execute as etapas a seguir. Você pode usar este procedimento para instalar o Astra Control Center em ambientes conectados à Internet ou com conexão via rede.

Expanda para outros procedimentos de instalação

- **Instalar com RedHat OpenShift OperatorHub:** Use isso ["procedimento alternativo"](#) para instalar o Astra Control Center no OpenShift usando o OperatorHub.
- **Instalar na nuvem pública com o Cloud Volumes ONTAP backend:** Use ["estes procedimentos"](#) para instalar o Astra Control Center no Amazon Web Services (AWS), no Google Cloud Platform (GCP) ou no Microsoft Azure com um back-end de storage do Cloud Volumes ONTAP.

Para uma demonstração do processo de instalação do Astra Control Center, ["este vídeo"](#) consulte .

Antes de começar

- ["Antes de começar a instalação, prepare seu ambiente para a implantação do Astra Control Center"](#).
- Se você configurou ou deseja configurar políticas de segurança de pod no seu ambiente, familiarize-se com as políticas de segurança de pod e como elas afetam a instalação do Astra Control Center. ["restrições de segurança do pod"](#) Consulte a .
- Certifique-se de que todos os serviços de API estão em um estado saudável e disponíveis:

```
kubectl get apiservices
```

- Certifique-se de que o Astra FQDN que pretende utilizar está roteável para este cluster. Isso significa que você tem uma entrada DNS no seu servidor DNS interno ou está usando uma rota URL principal que já está registrada.
- Se um gerenciador de cert já existir no cluster, você precisará executar alguns ["etapas de pré-requisito"](#) para que o Astra Control Center não tente instalar seu próprio gerenciador de cert. Por padrão, o Astra Control Center instala seu próprio gerenciador de cert durante a instalação.



Implante o Astra Control Center em um domínio de terceiros ou local secundário. Isso é recomendado para replicação de aplicativos e recuperação de desastres aprimorada.

Passos

Para instalar o Astra Control Center, siga estas etapas:

- [Faça download e extraia Astra Control Center](#)
- [Instale o plug-in NetApp Astra kubectl](#)
- [Adicione as imagens ao seu registro local](#)
- [Configure namespace e segredo para Registros com requisitos de autenticação](#)
- [Instale o operador do Centro de Controle Astra](#)
- [Configurar o Astra Control Center](#)

- [Instalação completa do operador e do Centro de Controle Astra](#)
- [Verifique o status do sistema](#)
- [Configure a entrada para o balanceamento de carga](#)
- [Faça login na IU do Astra Control Center](#)



Não exclua o operador Astra Control Center (por exemplo, `kubectl delete -f astra_control_center_operator_deploy.yaml`) a qualquer momento durante a instalação ou operação do Astra Control Center para evitar a exclusão de pods.

Faça download e extraia Astra Control Center

1. Faça o download do pacote que contém o Astra Control Center (`astra-control-center-[version].tar.gz`) no ["Página de downloads do Astra Control Center"](#).
2. (Recomendado, mas opcional) Faça o download do pacote certificados e assinaturas para o Astra Control Center (`astra-control-center-certs-[version].tar.gz`) para verificar a assinatura do pacote.

Expanda para obter detalhes

```
tar -vxzf astra-control-center-certs-[version].tar.gz
```

```
openssl dgst -sha256 -verify certs/AstraControlCenter-public.pub  
-signature certs/astra-control-center-[version].tar.gz.sig astra-  
control-center-[version].tar.gz
```

A saída será `Verified OK` exibida após a verificação bem-sucedida.

3. Extraia as imagens do pacote Astra Control Center:

```
tar -vxzf astra-control-center-[version].tar.gz
```

Instale o plug-in NetApp Astra kubectl

Você pode usar o plugin de linha de comando NetApp Astra kubectl para enviar imagens para um repositório local do Docker.

Antes de começar

O NetApp fornece binários de plug-in para diferentes arquiteturas de CPU e sistemas operacionais. Você precisa saber qual CPU e sistema operacional você tem antes de executar esta tarefa.

Se você já tiver o plugin instalado a partir de uma instalação anterior, ["certifique-se de que tem a versão mais recente"](#) antes de concluir estas etapas.

Passos

1. Liste os binários disponíveis do plug-in NetApp Astra kubectl:



A biblioteca de plugins kubectl faz parte do pacote tar e é extraída para a pasta kubectl-astra.

```
ls kubectl-astra/
```

2. Mova o arquivo necessário para o sistema operacional e a arquitetura da CPU para o caminho atual e renomeie-o para kubectl-astra:

```
cp kubectl-astra/<binary-name> /usr/local/bin/kubectl-astra
```

Adicione as imagens ao seu registo local

1. Complete a sequência de passos adequada para o seu motor de contentores:

Docker

1. Mude para o diretório raiz do tarball. Você deve ver o `acc.manifest.bundle.yaml` arquivo e estes diretórios:

```
acc/  
kubectl-astra/  
acc.manifest.bundle.yaml
```

2. Envie as imagens do pacote no diretório de imagens do Astra Control Center para o Registro local. Faça as seguintes substituições antes de executar o `push-images` comando:

- Substitua o `<BUNDLE_FILE>` pelo nome do arquivo do pacote Astra Control (`acc.manifest.bundle.yaml`).
- Substitua o `<MY_FULL_REGISTRY_PATH>` pela URL do repositório Docker; por exemplo "`<a href='\"https://<docker-registry>\"' class='\"bare\">https://<docker-registry>\"</a>`", .
- Substitua o `<MY_REGISTRY_USER>` pelo nome de usuário.
- Substitua o `<MY_REGISTRY_TOKEN>` por um token autorizado para o Registro.

```
kubectl astra packages push-images -m <BUNDLE_FILE> -r  
<MY_FULL_REGISTRY_PATH> -u <MY_REGISTRY_USER> -p  
<MY_REGISTRY_TOKEN>
```

Podman

1. Mude para o diretório raiz do tarball. Você deve ver este arquivo e diretório:

```
acc.manifest.bundle.yaml  
acc/
```

2. Inicie sessão no seu registro:

```
podman login <YOUR_REGISTRY>
```

3. Prepare e execute um dos seguintes scripts personalizados para a versão do Podman que você usa. Substitua o `<MY_FULL_REGISTRY_PATH>` pela URL do seu repositório que inclui quaisquer subdiretórios.

```
<strong>Podman 4</strong>
```

```

export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=23.07.0-25
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //'')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done

```

Podman 3

```

export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=23.07.0-25
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //'')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done

```



O caminho da imagem que o script cria deve ser semelhante ao seguinte, dependendo da configuração do Registro:

```

https://netappdownloads.jfrog.io/docker-astra-control-
prod/netapp/astra/acc/23.07.0-25/image:version

```

Configure namespace e segredo para Registros com requisitos de autenticação

1. Exporte o kubeconfig para o cluster de host Astra Control Center:

```
export KUBECONFIG=[file path]
```



Antes de concluir a instalação, certifique-se de que seu kubeconfig esteja apontando para o cluster onde você deseja instalar o Astra Control Center.

2. Se você usar um Registro que requer autenticação, você precisará fazer o seguinte:

Expanda para obter passos

a. Crie o `netapp-acc-operator` namespace:

```
kubectl create ns netapp-acc-operator
```

b. Crie um segredo para o `netapp-acc-operator` namespace. Adicione informações do Docker e execute o seguinte comando:



O marcador de posição `your_registry_path` deve corresponder à localização das imagens que carregou anteriormente (por exemplo, `[Registry_URL]/netapp/astra/astracc/23.07.0-25`).

```
kubectl create secret docker-registry astra-registry-cred -n  
netapp-acc-operator --docker-server=[your_registry_path] --docker  
-username=[username] --docker-password=[token]
```



Se você excluir o namespace depois que o segredo é gerado, recrie o namespace e, em seguida, regenere o segredo para o namespace.

c. Crie o `netapp-acc` namespace (ou nome personalizado).

```
kubectl create ns [netapp-acc or custom namespace]
```

d. Crie um segredo para o `netapp-acc` namespace (ou nome personalizado). Adicione informações do Docker e execute o seguinte comando:

```
kubectl create secret docker-registry astra-registry-cred -n  
[netapp-acc or custom namespace] --docker  
-server=[your_registry_path] --docker-username=[username]  
--docker-password=[token]
```

Instale o operador do Centro de Controle Astra

1. Altere o diretório:

```
cd manifests
```

2. Edite a implantação do operador Astra Control Center YAML)
(`astra\_control\_center\_operator\_deploy.yaml` para consultar o Registro local e o segredo.

```
vim astra_control_center_operator_deploy.yaml
```



Uma amostra anotada YAML segue estes passos.

- a. Se você usar um Registro que requer autenticação, substitua a linha padrão de `imagePullSecrets:` `[]` pelo seguinte:

```
imagePullSecrets: [{name: astra-registry-cred}]
```

- b. Altere `ASTRA_IMAGE_REGISTRY` para a `kube-rbac-proxy` imagem para o caminho do registo onde as imagens foram empurradas para um [passo anterior](#).
- c. Altere `ASTRA_IMAGE_REGISTRY` para a `acc-operator-controller-manager` imagem para o caminho do registo onde as imagens foram empurradas para um [passo anterior](#).

```

apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    control-plane: controller-manager
  name: acc-operator-controller-manager
  namespace: netapp-acc-operator
spec:
  replicas: 1
  selector:
    matchLabels:
      control-plane: controller-manager
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        control-plane: controller-manager
    spec:
      containers:
        - args:
            - --secure-listen-address=0.0.0.0:8443
            - --upstream=http://127.0.0.1:8080/
            - --logtostderr=true
            - --v=10
          image: ASTRA_IMAGE_REGISTRY/kube-rbac-proxy:v4.8.0
          name: kube-rbac-proxy
          ports:
            - containerPort: 8443
              name: https
        - args:
            - --health-probe-bind-address=:8081
            - --metrics-bind-address=127.0.0.1:8080
            - --leader-elect
          env:
            - name: ACCOP_LOG_LEVEL
              value: "2"
            - name: ACCOP_HELM_INSTALLTIMEOUT
              value: 5m
          image: ASTRA_IMAGE_REGISTRY/acc-operator:23.07.25
          imagePullPolicy: IfNotPresent
          livenessProbe:
            httpGet:
              path: /healthz

```

```
    port: 8081
    initialDelaySeconds: 15
    periodSeconds: 20
  name: manager
  readinessProbe:
    httpGet:
      path: /readyz
      port: 8081
    initialDelaySeconds: 5
    periodSeconds: 10
  resources:
    limits:
      cpu: 300m
      memory: 750Mi
    requests:
      cpu: 100m
      memory: 75Mi
  securityContext:
    allowPrivilegeEscalation: false
imagePullSecrets: []
  securityContext:
    runAsUser: 65532
  terminationGracePeriodSeconds: 10
```

3. Instale o operador do Centro de Controle Astra:

```
kubectl apply -f astra_control_center_operator_deploy.yaml
```

Expandir para resposta da amostra:

```
namespace/netapp-acc-operator created
customresourcedefinition.apiextensions.k8s.io/astracontrolcenters.as
tra.netapp.io created
role.rbac.authorization.k8s.io/acc-operator-leader-election-role
created
clusterrole.rbac.authorization.k8s.io/acc-operator-manager-role
created
clusterrole.rbac.authorization.k8s.io/acc-operator-metrics-reader
created
clusterrole.rbac.authorization.k8s.io/acc-operator-proxy-role
created
rolebinding.rbac.authorization.k8s.io/acc-operator-leader-election-
rolebinding created
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-manager-
rolebinding created
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-proxy-
rolebinding created
configmap/acc-operator-manager-config created
service/acc-operator-controller-manager-metrics-service created
deployment.apps/acc-operator-controller-manager created
```

4. Verifique se os pods estão em execução:

```
kubectl get pods -n netapp-acc-operator
```

Configurar o Astra Control Center

1. Edite o arquivo de recursos personalizados (CR) do Astra Control Center (astra_control_center.yaml) para criar contas, suporte, Registro e outras configurações necessárias:

```
vim astra_control_center.yaml
```



Uma amostra anotada YAML segue estes passos.

2. Modifique ou confirme as seguintes definições:

`<code>accountName</code>`

Definição	Orientação	Tipo	Exemplo
accountName	Altere a accountName cadeia de caracteres para o nome que deseja associar à conta Astra Control Center. Só pode haver uma accountName.	cadeia de caracteres	Example

`<code>astraVersion</code>`

Definição	Orientação	Tipo	Exemplo
astraVersion	A versão do Astra Control Center para implantação. Não é necessária nenhuma ação para esta definição, uma vez que o valor será pré-preenchido.	cadeia de caracteres	23.07.0-25

<code> </code>

Definição	Orientação	Tipo	Exemplo
astraAddress	Altere a astraAddress cadeia de caracteres para o endereço FQDN (recomendado) ou IP que você deseja usar em seu navegador para acessar o Astra Control Center. Esse endereço define como o Astra Control Center será encontrado em seu data center e será o mesmo FQDN ou endereço IP que você provisionou do balanceador de carga quando concluir "Requisitos do Astra Control Center" . NOTA: Não use http:// nem https:// no endereço. Copie este FQDN para uso em um passo posterior .	cadeia de caracteres	astra.example.com

<code> AutoSupport </code>

Suas seleções nesta seção determinam se você participará do aplicativo de suporte Pro-ativo da NetApp, do Consultor Digital e onde os dados são enviados. É necessária uma ligação à Internet (porta 442) e todos os dados de suporte são anonimizados.

Definição	Utilização	Orientação	Tipo	Exemplo
<code>autoSupport.enrolled</code>	enrolled`Os campos ou `url têm de ser selecionados	Alterar enrolled para AutoSupport para false sites sem conectividade com a Internet ou manter true para sites conectados. Uma configuração de true permite que dados anônimos sejam enviados para o NetApp para fins de suporte. A eleição padrão é false e indica que nenhum dado de suporte será enviado para o NetApp.	Booleano	false (este valor é o padrão)
<code>autoSupport.url</code>	enrolled`Os campos ou `url têm de ser selecionados	Esta URL determina onde os dados anônimos serão enviados.	cadeia de caracteres	https://support.netapp.com/asupprod/post/1.0/postAsup

`<code> email</code>`

Definição	Orientação	Tipo	Exemplo
email	Altere a email cadeia de caracteres para o endereço de administrador inicial padrão. Copie este endereço de e-mail para uso em um passo posterior . Este endereço de e-mail será usado como o nome de usuário da conta inicial para fazer login na IU e será notificado de eventos no Astra Control.	cadeia de caracteres	admin@example.com

`<code>firstName</code>`

Definição	Orientação	Tipo	Exemplo
firstName	O primeiro nome do administrador inicial padrão associado à conta Astra. O nome usado aqui será visível em um cabeçalho na IU após seu primeiro login.	cadeia de caracteres	SRE

`<code>LastName</code>`

Definição	Orientação	Tipo	Exemplo
lastName	O sobrenome do administrador inicial padrão associado à conta Astra. O nome usado aqui será visível em um cabeçalho na IU após seu primeiro login.	cadeia de caracteres	Admin

<code> imageRegistry</code>

Suas seleções nesta seção definem o Registro de imagem de contentor que hospeda as imagens do aplicativo Astra, o Operador do Centro de Controle Astra e o repositório do Astra Control Center Helm.

Definição	Utilização	Orientação	Tipo	Exemplo
<code>imageRegistry.name</code>	Obrigatório	O nome do registo de imagens onde as imagens foram enviadas para o passo anterior . Não utilize <code>http://</code> ou <code>https://</code> no nome do registo.	cadeia de caracteres	<code>example.registry.com/astra</code>
<code>imageRegistry.secret</code>	Obrigatório se a cadeia de caracteres inserida para <code>imageRegistry.name</code> requires a secret. IMPORTANT: If you are using a registry that does not require authorization, you must delete this <code>`secret</code> a linha <code>imageRegistry</code> ou a instalação falhar.	O nome do segredo do Kubernetes usado para autenticar com o Registro de imagens.	cadeia de caracteres	<code>astra-registry-cred</code>

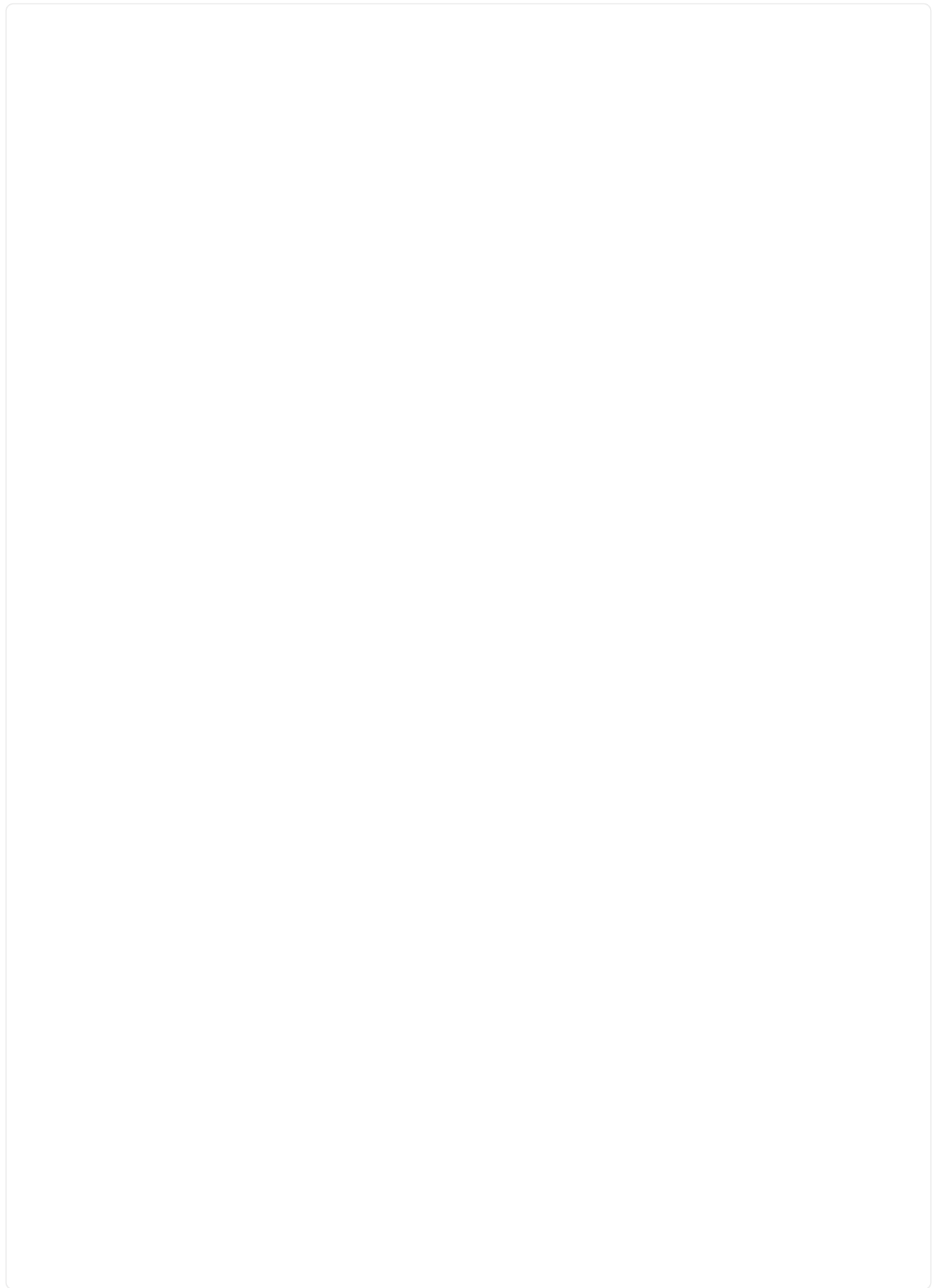
`<code>storageClass</code>`

Definição	Orientação	Tipo	Exemplo
<code>storageClass</code>	Altere <code>storageClass</code> o valor de <code>ontap-gold</code> para outro recurso de <code>storageClass</code> do Astra Trident, conforme exigido pela sua instalação. Execute o comando <code>kubectl get sc</code> para determinar suas classes de armazenamento configuradas existentes. Uma das classes de storage baseadas no Astra Trident deve ser inserida no arquivo MANIFEST (<code>astra-control-center-<version>.manifest</code>) e será usada para PVS Astra. Se não estiver definida, a classe de armazenamento padrão será usada. Nota: Se uma classe de armazenamento padrão estiver configurada, certifique-se de que é a única classe de armazenamento que tem a anotação padrão.	cadeia de caracteres	<code>ontap-gold</code>

`<code> volume ReclaimPolicy</code>`

Definição	Orientação	Tipo	Opções
<code>volumeReclaimPolicy</code>	Isso define a política de recuperação para PVS do Astra. Definir essa política para <code>Retain</code> reter volumes persistentes depois que o Astra for excluído. Definir essa política para <code>Delete</code> excluir volumes persistentes depois que o astra for excluído. Se este valor não for definido, os PVS são retidos.	cadeia de caracteres	• <code>Retain</code> (Este é o valor padrão) • <code>Delete</code>

`<code>ingressType</code>`





Definição	Orientação	Tipo	Opções
ingressType	Use um dos seguintes tipos de entrada: Generic (ingressType: "Generic") (Padrão) Use esta opção quando tiver outro controlador de entrada em uso ou preferir usar seu próprio controlador de entrada. Depois que o Astra Control Center for implantado, você precisará configurar o "controlador de entrada" para expor o Astra Control Center com um URL. AccTraefik (ingressType: "AccTraefik") Utilize esta opção quando preferir não configurar um controlador de entrada. Isso implanta o gateway Astra Control Center traefik como um serviço do tipo Kubernetes LoadBalancer. O Astra Control Center usa um serviço do tipo "LoadBalancer" (svc/traefik no namespace Astra Control Center) e exige que seja atribuído um endereço IP externo acessível. Se os balanceadores de carga forem permitidos em seu ambiente e você ainda não tiver um configurado, você poderá usar o MetalLB ou outro balanceador de carga de serviço externo para atribuir um endereço IP externo ao serviço. Na configuração do servidor DNS interno, você deve apontar o nome DNS escolhido	cadeia de caracteres	• Generic (este é o valor padrão) • AccTraefik

`<code> scaleSize</code>`

Definição	Orientação	Tipo	Opções
scaleSize	Por padrão, o Astra usará alta disponibilidade (HA scaleSize) do Medium, que implanta a maioria dos serviços no HA e implanta várias réplicas para redundância. Com scaleSize as Small, o Astra reduzirá o número de réplicas para todos os serviços, exceto para serviços essenciais para reduzir o consumo. Dica: Medium As implantações consistem em cerca de 100 pods (não incluindo cargas de trabalho transitórias. os pods do 100 são baseados em uma configuração de três nós mestre e três nós de trabalho). Esteja ciente das restrições de limite de rede por pod que podem ser um problema em seu ambiente, especialmente ao considerar cenários de recuperação de desastres.	cadeia de caracteres	• Small • Medium (Este é o valor padrão)

<code>astraResourcesScaler</code>

Definição	Orientação	Tipo	Opções
<code>astraResourcesScaler</code>	Opções de escala para os limites de recursos do AstraControlCenter. Por padrão, o Astra Control Center é implantado com solicitações de recursos definidas para a maioria dos componentes no Astra. Essa configuração permite que a pilha de software Astra Control Center tenha melhor desempenho em ambientes com maior carga e escalabilidade de aplicações. No entanto, em cenários que usam clusters de desenvolvimento ou teste menores, o campo <code>CR</code> <code>astraResourcesScaler</code> pode ser definido como <code>Off</code>. Isso desativa as solicitações de recursos e permite a implantação em clusters menores.	cadeia de caracteres	• <code>Default</code> (Este é o valor padrão) • <code>Off</code>

<code>AdditionalValues</code>



Adicione os seguintes valores adicionais ao Astra Control Center CR para evitar um problema conhecido na instalação do 23,07:

```
additionalValues:
  polaris-keycloak:
    livenessProbe:
      initialDelaySeconds: 180
    readinessProbe:
      initialDelaySeconds: 180
```

- Para a comunicação Astral Control Center e Cloud Insights, a verificação de certificado TLS é desativada por padrão. Você pode habilitar a verificação de certificação TLS para comunicação entre o Cloud Insights e o cluster de host e o cluster gerenciado do Astra Control Center adicionando a seguinte seção em `additionalValues`.

```
additionalValues:
  netapp-monitoring-operator:
    config:
      ciSkipTlsVerify: false
  cloud-insights-service:
    config:
      ciSkipTlsVerify: false
  telemetry-service:
    config:
      ciSkipTlsVerify: false
```

<code> crds</code>

Suas seleções nesta seção determinam como o Astra Control Center deve lidar com CRDs.

Definição	Orientação	Tipo	Exemplo
<code>crds.externalCertManager</code>	Se você usar um gerenciador cert externo, <code>externalCertManager</code> altere para <code>true</code> . O padrão <code>false</code> faz com que o Astra Control Center instale seus próprios CRDs de gerenciador de cert durante a instalação. CRDs são objetos de todo o cluster e instalá-los pode ter um impactos em outras partes do cluster. Você pode usar esse sinalizador para sinalizar para o Astra Control Center que essas CRDs serão instaladas e gerenciadas pelo administrador do cluster fora do Astra Control Center.	Booleano	<code>False</code> (este valor é o padrão)
<code>crds.externalTraefik</code>	Por padrão, o Astra Control Center instalará CRDs Traefik necessários. CRDs são objetos de todo o cluster e instalá-los pode ter um impactos em outras partes do cluster. Você pode usar esse sinalizador para sinalizar para o Astra Control Center que essas CRDs serão instaladas e gerenciadas pelo administrador do cluster fora do Astra Control Center.	Booleano	<code>False</code> (este valor é o padrão)



Certifique-se de que selecionou a classe de armazenamento e o tipo de entrada corretos para a sua configuração antes de concluir a instalação.

Expanda para amostra `astra_control_center.yaml`

```
apiVersion: astra.netapp.io/v1
kind: AstraControlCenter
metadata:
  name: astra
spec:
  accountName: "Example"
  astraVersion: "ASTRA_VERSION"
  astraAddress: "astra.example.com"
  autoSupport:
    enrolled: true
  email: "[admin@example.com]"
  firstName: "SRE"
  lastName: "Admin"
  imageRegistry:
    name: "[your_registry_path]"
    secret: "astra-registry-cred"
  storageClass: "ontap-gold"
  volumeReclaimPolicy: "Retain"
  ingressType: "Generic"
  scaleSize: "Medium"
  astraResourcesScaler: "Default"
  additionalValues:
    polaris-keycloak:
      livenessProbe:
        initialDelaySeconds: 180
      readinessProbe:
        initialDelaySeconds: 180
  crds:
    externalTraefik: false
    externalCertManager: false
```

Instalação completa do operador e do Centro de Controle Astra

1. Se você ainda não fez isso em uma etapa anterior, crie o `netapp-acc` namespace (ou personalizado):

```
kubectl create ns [netapp-acc or custom namespace]
```

2. Instale o Astra Control Center no `netapp-acc` namespace (ou personalizado):

```
kubectl apply -f astra_control_center.yaml -n [netapp-acc or custom namespace]
```



O operador do Astra Control Center executará uma verificação automática dos requisitos de ambiente. A falta "[requisitos](#)" pode fazer com que a instalação falhe ou o Astra Control Center não funcione corretamente. [próxima seção](#) Consulte para verificar se existem mensagens de aviso relacionadas com a verificação automática do sistema.

Verifique o status do sistema

Você pode verificar o status do sistema usando comandos kubectl. Se você preferir usar OpenShift, você pode usar comandos oc comparáveis para etapas de verificação.

Passos

1. Verifique se o processo de instalação não produziu mensagens de avisos relacionadas às verificações de validação:

```
kubectl get acc [astra or custom Astra Control Center CR name] -n [netapp-acc or custom namespace] -o yaml
```



Mensagens de aviso adicionais também são relatadas nos logs do operador do Centro de Controle Astra.

2. Corrija quaisquer problemas com seu ambiente que foram relatados pelas verificações automatizadas de requisitos.



Você pode corrigir problemas garantindo que seu ambiente atenda ao do "[requisitos](#)" para Astra Control Center.

3. Verifique se todos os componentes do sistema foram instalados com êxito.

```
kubectl get pods -n [netapp-acc or custom namespace]
```

Cada pod deve ter um status de `Running`. Pode levar alguns minutos até que os pods do sistema sejam implantados.

Expandir para resposta de amostra

NAME	READY	STATUS	
RESTARTS AGE			
acc-helm-repo-6cc7696d8f-pmhm8 9h	1/1	Running	0
activity-597fb656dc-5rd4l 9h	1/1	Running	0
activity-597fb656dc-mqmcw 9h	1/1	Running	0
api-token-authentication-62f84 9h	1/1	Running	0
api-token-authentication-68nlf 9h	1/1	Running	0
api-token-authentication-ztgrm 9h	1/1	Running	0
asup-669d4ddbc4-fnmwp (9h ago) 9h	1/1	Running	1
authentication-78789d7549-lk686 9h	1/1	Running	0
bucket-service-65c7d95496-24x7l (9h ago) 9h	1/1	Running	3
cert-manager-c9f9fbf9f-k8zq2 9h	1/1	Running	0
cert-manager-c9f9fbf9f-qj1zm 9h	1/1	Running	0
cert-manager-cainjector-dbbbd8447-b5q1l 9h	1/1	Running	0
cert-manager-cainjector-dbbbd8447-p5whs 9h	1/1	Running	0
cert-manager-webhook-6f97bb7d84-4722b 9h	1/1	Running	0
cert-manager-webhook-6f97bb7d84-86kv5 9h	1/1	Running	0
certificates-59d9f6f4bd-2j899 9h	1/1	Running	0
certificates-59d9f6f4bd-9d9k6 9h	1/1	Running	0
certificates-expiry-check-28011180--1-8lkxz 9h	0/1	Completed	0
cloud-extension-5c9c9958f8-jdhrp 9h	1/1	Running	0
cloud-insights-service-5cdd5f7f-pp8r5 9h	1/1	Running	0
composite-compute-66585789f4-hxn5w 9h	1/1	Running	0

composite-volume-68649f68fd-tb7p4 9h	1/1	Running	0
credentials-dfc844c57-jsx92 9h	1/1	Running	0
credentials-dfc844c57-xw26s 9h	1/1	Running	0
entitlement-7b47769b87-4jb6c 9h	1/1	Running	0
features-854d8444cc-c24b7 9h	1/1	Running	0
features-854d8444cc-dv6sm 9h	1/1	Running	0
fluent-bit-ds-9tlv4 9h	1/1	Running	0
fluent-bit-ds-bpkcb 9h	1/1	Running	0
fluent-bit-ds-cxmwx 9h	1/1	Running	0
fluent-bit-ds-jgnhc 9h	1/1	Running	0
fluent-bit-ds-vtr6k 9h	1/1	Running	0
fluent-bit-ds-vxqd5 9h	1/1	Running	0
graphql-server-7d4b9d44d5-zdbf5 9h	1/1	Running	0
identity-6655c48769-4pwk8 9h	1/1	Running	0
influxdb2-0 9h	1/1	Running	0
keycloak-operator-55479d6fc6-slvmt 9h	1/1	Running	0
krakend-f487cb465-78679 9h	1/1	Running	0
krakend-f487cb465-rjsxx 9h	1/1	Running	0
license-64cbc7cd9c-qxsr8 9h	1/1	Running	0
login-ui-5db89b5589-ndb96 9h	1/1	Running	0
loki-0 9h	1/1	Running	0
metrics-facade-8446f64c94-x8h7b 9h	1/1	Running	0
monitoring-operator-6b44586965-pvcl4 9h	2/2	Running	0

nats-0 9h	1/1	Running	0
nats-1 9h	1/1	Running	0
nats-2 9h	1/1	Running	0
nautilus-85754d87d7-756qb 9h	1/1	Running	0
nautilus-85754d87d7-q8j7d 9h	1/1	Running	0
openapi-5f9cc76544-7fnjm 9h	1/1	Running	0
openapi-5f9cc76544-vzr7b 9h	1/1	Running	0
packages-5db49f8b5-lrzhd 9h	1/1	Running	0
polaris-consul-consul-server-0 9h	1/1	Running	0
polaris-consul-consul-server-1 9h	1/1	Running	0
polaris-consul-consul-server-2 9h	1/1	Running	0
polaris-keycloak-0 (9h ago) 9h	1/1	Running	2
polaris-keycloak-1 9h	1/1	Running	0
polaris-keycloak-2 9h	1/1	Running	0
polaris-keycloak-db-0 9h	1/1	Running	0
polaris-keycloak-db-1 9h	1/1	Running	0
polaris-keycloak-db-2 9h	1/1	Running	0
polaris-mongodb-0 9h	1/1	Running	0
polaris-mongodb-1 9h	1/1	Running	0
polaris-mongodb-2 9h	1/1	Running	0
polaris-ui-66fb99479-qp9gq 9h	1/1	Running	0
polaris-vault-0 9h	1/1	Running	0
polaris-vault-1 9h	1/1	Running	0

polaris-vault-2 9h	1/1	Running	0
public-metrics-76fbf9594d-zmxzw 9h	1/1	Running	0
storage-backend-metrics-7d7fbc9cb9-lmd25 9h	1/1	Running	0
storage-provider-5bdd456c4b-2fftc 9h	1/1	Running	0
task-service-87575df85-dnn2q (9h ago) 9h	1/1	Running	3
task-service-task-purge-28011720--1-q6w4r 28m	0/1	Completed	0
task-service-task-purge-28011735--1-vk6pd 13m	1/1	Running	0
telegraf-ds-2r2kw 9h	1/1	Running	0
telegraf-ds-6s9d5 9h	1/1	Running	0
telegraf-ds-96jl7 9h	1/1	Running	0
telegraf-ds-hbp84 9h	1/1	Running	0
telegraf-ds-plwzv 9h	1/1	Running	0
telegraf-ds-sr22c 9h	1/1	Running	0
telegraf-rs-4sbg8 9h	1/1	Running	0
telemetry-service-fb9559f7b-mk9l7 (9h ago) 9h	1/1	Running	3
tenancy-559bbc6b48-5msgg 9h	1/1	Running	0
traefik-d997b8877-7xpf4 9h	1/1	Running	0
traefik-d997b8877-9xv96 9h	1/1	Running	0
trident-svc-585c97548c-d25z5 9h	1/1	Running	0
vault-controller-88484b454-2d6sr 9h	1/1	Running	0
vault-controller-88484b454-fc5cz 9h	1/1	Running	0
vault-controller-88484b454-jktld 9h	1/1	Running	0

4. (Opcional) Assista os `acc-operator` logs para monitorar o progresso:

```
kubectl logs deploy/acc-operator-controller-manager -n netapp-acc-operator -c manager -f
```



`accHost` o registro de cluster é uma das últimas operações e, se falhar, não causará falha na implantação. No caso de uma falha de Registro de cluster indicada nos logs, você pode tentar o Registro novamente por meio da ["Adicione fluxo de trabalho de cluster na IU"](#) API ou.

5. Quando todos os pods estiverem em execução, verifique se a instalação foi bem-sucedida (`READY` é `True`) e obtenha a senha de configuração inicial que você usará quando fizer login no Astra Control Center:

```
kubectl get AstraControlCenter -n [netapp-acc or custom namespace]
```

Resposta:

NAME	UUID	VERSION	ADDRESS
READY			
astra	9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f	23.07.0-25	
10.111.111.111	True		



Copie o valor UUID. A palavra-passe é `ACC-` seguida pelo valor UUID (`ACC-[UUID]` ou, neste exemplo, `ACC-9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f`).

Configure a entrada para o balanceamento de carga

Você pode configurar uma controladora de ingresso do Kubernetes que gerencia o acesso externo a serviços. Esses procedimentos fornecem exemplos de configuração para um controlador de entrada se você usou o padrão do no recurso personalizado do `ingressType: "Generic"` Astra Control Center (`astra_control_center.yaml`). Não é necessário usar este procedimento se você especificou `ingressType: "AccTraefik"` no recurso personalizado do Astra Control Center (`astra_control_center.yaml`).

Depois que o Astra Control Center for implantado, você precisará configurar o controlador Ingress para expor o Astra Control Center com um URL.

As etapas de configuração diferem dependendo do tipo de controlador de entrada que você usa. O Astra Control Center é compatível com muitos tipos de controlador de entrada. Estes procedimentos de configuração fornecem passos de exemplo para alguns tipos comuns de controlador de entrada.

Antes de começar

- O necessário ["controlador de entrada"](#) já deve ser implantado.
- O ["classe de entrada"](#) correspondente ao controlador de entrada já deve ser criado.

Etapas para a entrada do Istio

1. Configurar a entrada do Istio.



Este procedimento pressupõe que o Istio é implantado usando o perfil de configuração "padrão".

2. Reúna ou crie o certificado e o arquivo de chave privada desejados para o Ingress Gateway.

Você pode usar um certificado assinado pela CA ou autoassinado. O nome comum deve ser o endereço Astra (FQDN).

Exemplo de comando:

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout tls.key  
-out tls.crt
```

3. Crie um segredo `tls secret name` do tipo `kubernetes.io/tls` para uma chave privada TLS e um certificado, `istio-system` namespace conforme descrito em segredos TLS.

Exemplo de comando:

```
kubectl create secret tls [tls secret name] --key="tls.key"  
--cert="tls.crt" -n istio-system
```



O nome do segredo deve corresponder ao `spec.tls.secretName` fornecido no `istio-ingress.yaml` arquivo.

4. Implante um recurso de entrada no `netapp-acc` namespace (ou nome personalizado) usando o tipo de recurso `v1` para um esquema (`istio-Ingress.yaml` é usado neste exemplo):

```

apiVersion: networking.k8s.io/v1
kind: IngressClass
metadata:
  name: istio
spec:
  controller: istio.io/ingress-controller
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: ingress
  namespace: [netapp-acc or custom namespace]
spec:
  ingressClassName: istio
  tls:
  - hosts:
    - <ACC address>
    secretName: [tls secret name]
  rules:
  - host: [ACC address]
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: traefik
            port:
              number: 80

```

5. Aplicar as alterações:

```
kubectl apply -f istio-Ingress.yaml
```

6. Verifique o estado da entrada:

```
kubectl get ingress -n [netapp-acc or custom namespace]
```

Resposta:

NAME	CLASS	HOSTS	ADDRESS	PORTS	AGE
ingress	istio	astra.example.com	172.16.103.248	80, 443	1h

7. Concluir a instalação do Astra Control Center.

Etapas para o controlador nginx Ingress

1. Crie um segredo do tipo `kubernetes.io/tls` para uma chave privada TLS e um certificado no `netapp-acc` namespace (ou nome personalizado), conforme descrito em "[Segredos TLS](#)".
2. Implantar um recurso de entrada no `netapp-acc` namespace (ou nome personalizado) usando o tipo de recurso `v1` para um esquema (`nginx-Ingress.yaml` é usado neste exemplo):

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: netapp-acc-ingress
  namespace: [netapp-acc or custom namespace]
spec:
  ingressClassName: [class name for nginx controller]
  tls:
    - hosts:
        - <ACC address>
      secretName: [tls secret name]
  rules:
    - host: <ACC address>
      http:
        paths:
          - path:
              backend:
                service:
                  name: traefik
                  port:
                    number: 80
              pathType: ImplementationSpecific
```

3. Aplicar as alterações:

```
kubectl apply -f nginx-Ingress.yaml
```



O NetApp recomenda a instalação do controlador nginx como uma implementação em vez de um `daemonSet`.

Passos para o controlador OpenShift Ingress

1. Procure seu certificado e prepare os arquivos de chave, certificado e CA para uso pela rota OpenShift.
2. Crie a rota OpenShift:

```
oc create route edge --service=traefik --port=web -n [netapp-acc or custom namespace] --insecure-policy=Redirect --hostname=<ACC address> --cert=cert.pem --key=key.pem
```

Faça login na IU do Astra Control Center

Depois de instalar o Astra Control Center, você alterará a senha do administrador padrão e fará login no painel da IU do Astra Control Center.

Passos

1. Em um navegador, insira o FQDN (incluindo o `https://` prefixo) usado no `astraAddress` `astra_control_center.yaml` CR quando [Você instalou o Astra Control Center](#).
2. Aceite os certificados autoassinados, se solicitado.



Você pode criar um certificado personalizado após o login.

3. Na página de login do Astra Control Center, insira o valor usado `email` no `astra_control_center.yaml` CR quando [Você instalou o Astra Control Center](#), seguido da senha de configuração inicial (`ACC-[UUID]`).



Se você digitar uma senha incorreta três vezes, a conta de administrador será bloqueada por 15 minutos.

4. Selecione **Login**.
5. Altere a senha quando solicitado.



Se este for o seu primeiro login e você esquecer a senha e nenhuma outra conta de usuário administrativo ainda tiver sido criada, entre em Contato ["Suporte à NetApp"](#) para obter assistência de recuperação de senha.

6. (Opcional) Remova o certificado TLS autoassinado existente e substitua-o por um ["Certificado TLS personalizado assinado por uma autoridade de certificação \(CA\)"](#).

Solucionar problemas da instalação

Se algum dos serviços estiver `Error` no estado, pode inspecionar os registros. Procure códigos de resposta da API na faixa 400 a 500. Eles indicam o lugar onde uma falha aconteceu.

Opções

- Para inspecionar os logs do operador do Centro de Controle Astra, digite o seguinte:

```
kubectl logs deploy/acc-operator-controller-manager -n netapp-acc-operator -c manager -f
```

- Para verificar a saída do Astra Control Center CR:

```
kubectl get acc -n [netapp-acc or custom namespace] -o yaml
```

O que vem a seguir

- (Opcional) dependendo do seu ambiente, conclua a pós-instalação "[etapas de configuração](#)".
- Conclua a implantação executando "[tarefas de configuração](#)"o .

Configurar um gerenciador de cert externo

Se um gerenciador de cert já existir no cluster do Kubernetes, você precisará executar algumas etapas de pré-requisito para que o Astra Control Center não instale seu próprio gerenciador de cert.

Passos

1. Confirme se você tem um gerenciador cert instalado:

```
kubectl get pods -A | grep 'cert-manager'
```

Resposta da amostra:

cert-manager	essential-cert-manager-84446f49d5-sf2zd	1/1
Running	0	6d5h
cert-manager	essential-cert-manager-cainjector-66dc99cc56-9ldmt	1/1
Running	0	6d5h
cert-manager	essential-cert-manager-webhook-56b76db9cc-fjqrq	1/1
Running	0	6d5h

2. Crie um par de certificados/chaves para o astraAddress FQDN:

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout tls.key -out  
tls.crt
```

Resposta da amostra:

```
Generating a 2048 bit RSA private key
.....+++
.....+++
writing new private key to 'tls.key'
```

3. Crie um segredo com arquivos gerados anteriormente:

```
kubectl create secret tls selfsigned-tls --key tls.key --cert tls.crt -n
<cert-manager-namespace>
```

Resposta da amostra:

```
secret/selfsigned-tls created
```

4. Crie um ClusterIssuer arquivo que seja **exatamente** a seguir, mas inclua o local do namespace onde seus cert-manager pods estão instalados:

```
apiVersion: cert-manager.io/v1
kind: ClusterIssuer
metadata:
  name: astra-ca-clusterissuer
  namespace: <cert-manager-namespace>
spec:
  ca:
    secretName: selfsigned-tls
```

```
kubectl apply -f ClusterIssuer.yaml
```

Resposta da amostra:

```
clusterissuer.cert-manager.io/astra-ca-clusterissuer created
```

5. Verifique se o ClusterIssuer foi apresentado corretamente. Ready deve ser True antes que você possa prosseguir:

```
kubectl get ClusterIssuer
```

Resposta da amostra:

NAME	READY	AGE
astra-ca-clusterissuer	True	9s

- Preencha "[Processo de instalação do Astra Control Center](#)"o . Há um "[Etapa de configuração necessária para o cluster Astra Control Center YAML](#)" em que você altera o valor CRD para indicar que o gerenciador cert está instalado externamente. Você deve concluir esta etapa durante a instalação para que o Astra Control Center reconheça o gerenciador de cert externo.

Instale o Astra Control Center usando o OpenShift OperatorHub

Se você usar o Red Hat OpenShift, poderá instalar o Astra Control Center usando o operador certificado Red Hat. Use este procedimento para instalar o Astra Control Center a partir do "[Catálogo de ecossistemas da Red Hat](#)" ou usando o Red Hat OpenShift Container Platform.

Depois de concluir este procedimento, terá de voltar ao procedimento de instalação para concluir o para verificar o "[passos restantes](#)"êxito da instalação e iniciar sessão.

Antes de começar

- * Pré-requisitos ambientais atendidos*: "[Antes de começar a instalação, prepare seu ambiente para a implantação do Astra Control Center](#)".
- * Operadores de cluster e serviços de API saudáveis*:
 - A partir do cluster OpenShift, certifique-se de que todos os operadores de cluster estão em um estado saudável:

```
oc get clusteroperators
```

- A partir do cluster OpenShift, certifique-se de que todos os serviços de API estão em um estado saudável:

```
oc get apiservices
```

- **Endereço FQDN:** Obtenha um endereço FQDN para o Astra Control Center em seu data center.
- **Permissões OpenShift:** Obtenha as permissões necessárias e o acesso à Red Hat OpenShift Container Platform para executar as etapas de instalação descritas.
- **Gerenciador de cert configurado:** Se um gerenciador de cert já existir no cluster, você precisará executar alguns "[etapas de pré-requisito](#)" para que o Astra Control Center não instale seu próprio gerenciador de cert. Por padrão, o Astra Control Center instala seu próprio gerenciador de cert durante a instalação.
- * Controlador de entrada de Kubernetes*: Se você tiver uma controladora de entrada de Kubernetes que gerencia o acesso externo a serviços, como balanceamento de carga em um cluster, será necessário configurá-la para uso com o Astra Control Center:
 - Crie o namespace do operador:

```
oc create namespace netapp-acc-operator
```

- b. ["Conclua a configuração"](#) para o seu tipo de controlador de entrada.

Passos

- [Faça download e extraia Astra Control Center](#)
- [Instale o plug-in NetApp Astra kubectl](#)
- [Adicione as imagens ao seu registo local](#)
- [Localize a página de instalação do operador](#)
- [Instale o operador](#)
- [Instale o Astra Control Center](#)

Faça download e extraia Astra Control Center

1. Faça o download do pacote que contém o Astra Control Center (`astra-control-center-[version].tar.gz`) no ["Página de downloads do Astra Control Center"](#).
2. (Recomendado, mas opcional) Faça o download do pacote certificados e assinaturas para o Astra Control Center (`astra-control-center-certs-[version].tar.gz`) para verificar a assinatura do pacote.

Expanda para obter detalhes

```
tar -vxzf astra-control-center-certs-[version].tar.gz
```

```
openssl dgst -sha256 -verify certs/AstraControlCenter-public.pub  
-signature certs/astra-control-center-[version].tar.gz.sig astra-  
control-center-[version].tar.gz
```

A saída será `Verified OK` exibida após a verificação bem-sucedida.

3. Extraia as imagens do pacote Astra Control Center:

```
tar -vxzf astra-control-center-[version].tar.gz
```

Instale o plug-in NetApp Astra kubectl

Você pode usar o plugin de linha de comando NetApp Astra kubectl para enviar imagens para um repositório local do Docker.

Antes de começar

O NetApp fornece binários de plug-in para diferentes arquiteturas de CPU e sistemas operacionais. Você precisa saber qual CPU e sistema operacional você tem antes de executar esta tarefa.

Passos

1. Liste os binários disponíveis do plug-in NetApp Astra kubectl e observe o nome do arquivo que você precisa para o seu sistema operacional e arquitetura de CPU:



A biblioteca de plugins kubectl faz parte do pacote tar e é extraída para a pasta kubectl-astra.

```
ls kubectl-astra/
```

2. Mova o binário correto para o caminho atual e renomeie-o para kubectl-astra:

```
cp kubectl-astra/<binary-name> /usr/local/bin/kubectl-astra
```

Adicione as imagens ao seu registo local

1. Complete a sequência de passos adequada para o seu motor de contentores:

Docker

1. Mude para o diretório raiz do tarball. Você deve ver o `acc.manifest.bundle.yaml` arquivo e estes diretórios:

```
acc/  
kubectl-astra/  
acc.manifest.bundle.yaml
```

2. Envie as imagens do pacote no diretório de imagens do Astra Control Center para o Registro local. Faça as seguintes substituições antes de executar o `push-images` comando:

- Substitua o `<BUNDLE_FILE>` pelo nome do arquivo do pacote Astra Control (`acc.manifest.bundle.yaml`).
- Substitua o `<MY_FULL_REGISTRY_PATH>` pela URL do repositório Docker; por exemplo "`<a href='\"https://<docker-registry>\"' class='\"bare\">https://<docker-registry>\"</a>`", .
- Substitua o `<MY_REGISTRY_USER>` pelo nome de usuário.
- Substitua o `<MY_REGISTRY_TOKEN>` por um token autorizado para o Registro.

```
kubectl astra packages push-images -m <BUNDLE_FILE> -r  
<MY_FULL_REGISTRY_PATH> -u <MY_REGISTRY_USER> -p  
<MY_REGISTRY_TOKEN>
```

Podman

1. Mude para o diretório raiz do tarball. Você deve ver este arquivo e diretório:

```
acc.manifest.bundle.yaml  
acc/
```

2. Inicie sessão no seu registro:

```
podman login <YOUR_REGISTRY>
```

3. Prepare e execute um dos seguintes scripts personalizados para a versão do Podman que você usa. Substitua o `<MY_FULL_REGISTRY_PATH>` pela URL do seu repositório que inclui quaisquer subdiretórios.

```
<strong>Podman 4</strong>
```

```

export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=23.07.0-25
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //'')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*://:')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done

```

Podman 3

```

export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=23.07.0-25
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //'')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*://:')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done

```



O caminho da imagem que o script cria deve ser semelhante ao seguinte, dependendo da configuração do Registro:

```

https://netappdownloads.jfrog.io/docker-astra-control-
prod/netapp/astra/acc/23.07.0-25/image:version

```

Localize a página de instalação do operador

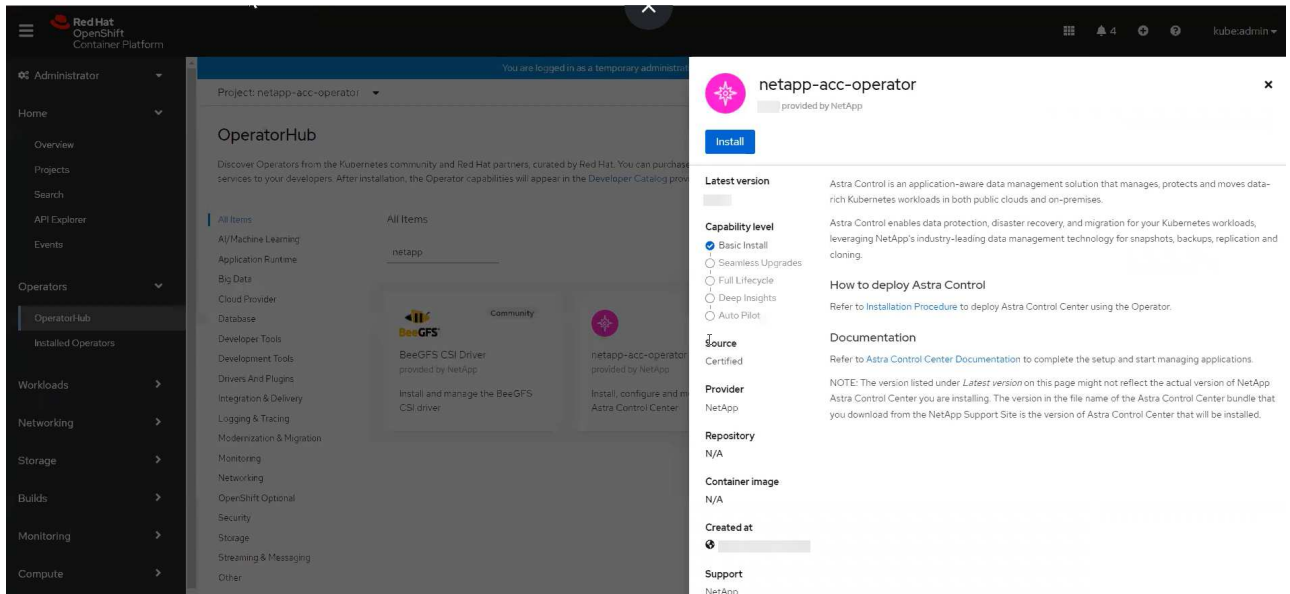
1. Execute um dos seguintes procedimentos para acessar a página de instalação do operador:

- A partir do console web Red Hat OpenShift:
 - i. Faça login na IU da OpenShift Container Platform.
 - ii. No menu lateral, selecione **operadores > OperatorHub**.

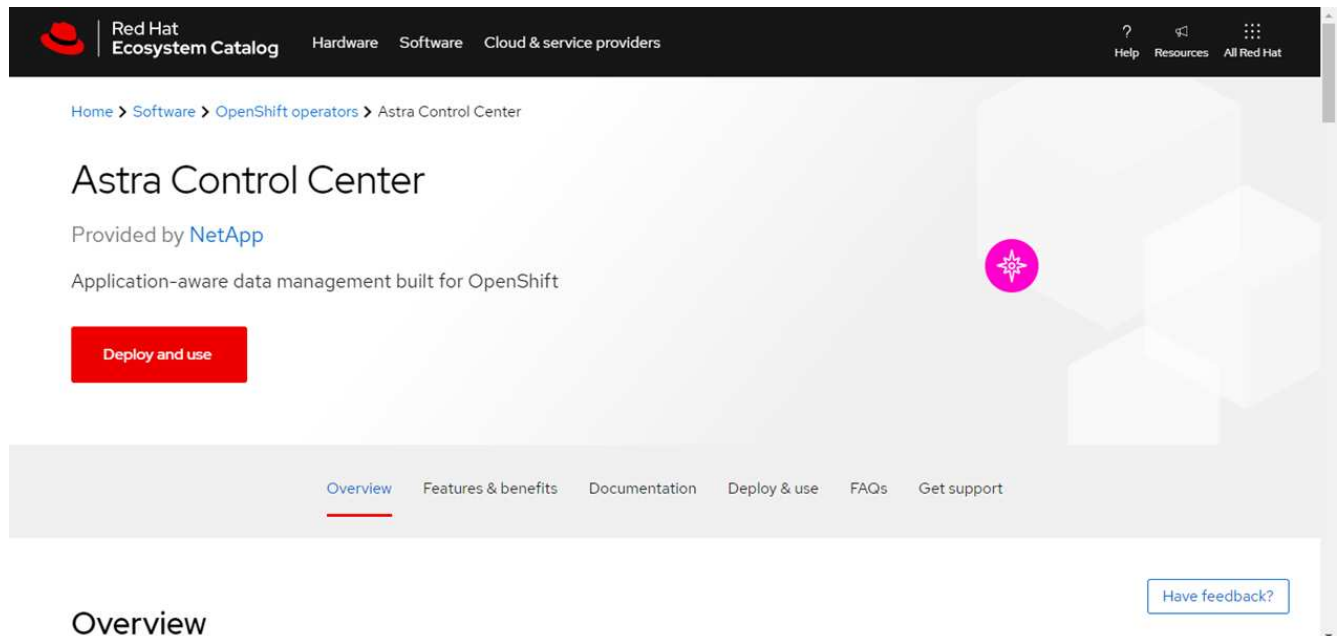


Você só pode fazer upgrade para a versão atual do Astra Control Center usando esse operador.

- iii. Procure e selecione o operador do Centro de Controle NetApp Astra.



- No Red Hat Ecosystem Catalog:
 - i. Selecione o Centro de Controle NetApp Astra "operador" .
 - ii. Selecione **Deploy and use**.



Instale o operador

1. Preencha a página **Instalar Operador** e instale o operador:



O operador estará disponível em todos os namespaces de cluster.

- a. Selecione o namespace do operador ou `netapp-acc-operator` o namespace será criado automaticamente como parte da instalação do operador.
- b. Selecione uma estratégia de aprovação manual ou automática.



Recomenda-se a aprovação manual. Você deve ter apenas uma única instância de operador em execução por cluster.

- c. Selecione **Instalar**.



Se selecionou uma estratégia de aprovação manual, será-lhe pedido que aprove o plano de instalação manual para este operador.

2. No console, vá para o menu OperatorHub e confirme se o operador instalou com êxito.

Instale o Astra Control Center

1. No console dentro da guia **Astra Control Center** do operador Astra Control Center, selecione **Create AstraControlCenter**.

2. Preencha o **Create AstraControlCenter** campo do formulário:
 - a. Mantenha ou ajuste o nome do Astra Control Center.
 - b. Adicione etiquetas para o Astra Control Center.
 - c. Ative ou desative o suporte automático. Recomenda-se a manutenção da funcionalidade de suporte automático.
 - d. Insira o FQDN ou o endereço IP do Centro de Controle Astra. Não introduza `http://` ou `https://` no campo de endereço.
 - e. Digite a versão do Astra Control Center; por exemplo, 23.07.0-25.
 - f. Insira um nome de conta, endereço de e-mail e sobrenome do administrador.
 - g. Escolha uma política de recuperação de volume de `Retain`, `Recycle` ou `Delete`. O valor padrão é

Retain.

h. Selecione o `scaleSize` da instalação.



Por padrão, o Astra usará alta disponibilidade (HA `scaleSize`) do `Medium`, que implanta a maioria dos serviços no HA e implanta várias réplicas para redundância. Com `scaleSize` as `Small`, o Astra reduzirá o número de réplicas para todos os serviços, exceto para serviços essenciais para reduzir o consumo.

i. Selecione o tipo de entrada:

▪ **Generic**(`ingressType`: "**Generic**") (Predefinição)

Utilize esta opção quando tiver outro controlador de entrada em utilização ou preferir utilizar o seu próprio controlador de entrada. Depois que o Astra Control Center for implantado, você precisará configurar o "[controlador de entrada](#)" para expor o Astra Control Center com um URL.

▪ **AccTraefik**(`ingressType`: "**AccTraefik**")

Utilize esta opção quando preferir não configurar um controlador de entrada. Isso implanta o gateway Astra Control Center `traefik` como um serviço do tipo "LoadBalancer" do Kubernetes.

O Astra Control Center usa um serviço do tipo "LoadBalancer" (`svc/traefik` no namespace Astra Control Center) e exige que seja atribuído um endereço IP externo acessível. Se os balanceadores de carga forem permitidos em seu ambiente e você ainda não tiver um configurado, você poderá usar o MetalLB ou outro balanceador de carga de serviço externo para atribuir um endereço IP externo ao serviço. Na configuração do servidor DNS interno, você deve apontar o nome DNS escolhido para o Astra Control Center para o endereço IP com balanceamento de carga.



Para obter detalhes sobre o tipo de serviço "LoadBalancer" e Ingress, "[Requisitos](#)" consulte .

a. Em **Image Registry**, insira seu caminho de Registro de imagem de contentor local. Não introduza `http://` ou `https://` no campo de endereço.

b. Se utilizar um registo de imagens que necessite de autenticação, introduza o segredo da imagem.



Se você usar um Registro que requer autenticação, [crie um segredo no cluster](#).

c. Introduza o nome do administrador.

d. Configurar o dimensionamento de recursos.

e. Forneça a classe de armazenamento padrão.



Se uma classe de armazenamento padrão estiver configurada, certifique-se de que é a única classe de armazenamento que tem a anotação padrão.

f. Definir preferências de tratamento de CRD.

3. Selecione a vista YAML para rever as definições selecionadas.

4. `Create` Selecione .

Crie um segredo de Registro

Se você usar um Registro que requer autenticação, crie um segredo no cluster OpenShift e insira o nome secreto no `Create AstraControlCenter` campo formulário.

1. Crie um namespace para o operador Astra Control Center:

```
oc create ns [netapp-acc-operator or custom namespace]
```

2. Crie um segredo neste namespace:

```
oc create secret docker-registry astra-registry-cred n [netapp-acc-operator or custom namespace] --docker-server=[your_registry_path] --docker-username=[username] --docker-password=[token]
```



O Astra Control suporta apenas segredos de registro do Docker.

3. Preencha os campos restantes em [O campo criar formulário AstraControlCenter](#).

O que vem a seguir

Preencha o "[passos restantes](#)" para verificar se o Astra Control Center foi instalado com sucesso, configure um controlador de entrada (opcional) e faça login na IU. Além disso, você precisará executar "[tarefas de configuração](#)" depois de concluir a instalação.

Instalar o Astra Control Center com um back-end de storage do Cloud Volumes ONTAP

Com o Astra Control Center, você pode gerenciar suas aplicações em um ambiente de nuvem híbrida com clusters Kubernetes autogerenciados e instâncias do Cloud Volumes ONTAP. É possível implantar o Astra Control Center nos clusters do Kubernetes no local ou em um dos clusters do Kubernetes autogerenciado no ambiente de nuvem.

Em uma dessas implantações, você pode executar operações de gerenciamento de dados de aplicações usando o Cloud Volumes ONTAP como um back-end de storage. Você também pode configurar um bucket do S3 como o destino de backup.

Para instalar o Astra Control Center no Amazon Web Services (AWS), no Google Cloud Platform (GCP) e no Microsoft Azure com um back-end de storage do Cloud Volumes ONTAP, execute as etapas a seguir, dependendo do ambiente de nuvem.

- [Implante o Astra Control Center na Amazon Web Services](#)
- [Implante o Astra Control Center no Google Cloud Platform](#)
- [Implante o Astra Control Center no Microsoft Azure](#)

Você pode gerenciar seus aplicativos em distribuições com clusters do Kubernetes autogerenciados, como o OpenShift Container Platform (OCP). Somente clusters de OCP autogeridos são validados para implantar o Astra Control Center.

Implante o Astra Control Center na Amazon Web Services

É possível implantar o Astra Control Center em um cluster Kubernetes autogerenciado hospedado em uma nuvem pública da Amazon Web Services (AWS).

O que você precisará para a AWS

Antes de implantar o Astra Control Center na AWS, você precisará dos seguintes itens:

- Licença do Astra Control Center. "[Requisitos de licenciamento do Astra Control Center](#)" Consulte a .
- "[Atender aos requisitos do Astra Control Center](#)".
- Conta do NetApp Cloud Central
- Se estiver usando OCP, permissões do Red Hat OpenShift Container Platform (OCP) (no nível do namespace para criar pods)
- Credenciais da AWS, ID de acesso e chave secreta com permissões que permitem criar buckets e conetores
- Acesso e login do AWS Account Elastic Container Registry (ECR)
- A zona hospedada da AWS e a entrada do Route 53 são necessárias para acessar a IU do Astra Control

Requisitos de ambiente operacional para a AWS

O Astra Control Center requer o seguinte ambiente operacional para a AWS:

- Red Hat OpenShift Container Platform 4,11 a 4,13



Certifique-se de que o ambiente operacional escolhido para hospedar o Astra Control Center atenda aos requisitos básicos de recursos descritos na documentação oficial do ambiente.

O Astra Control Center requer os seguintes recursos, além dos requisitos de recursos do ambiente:

Componente	Requisito
Capacidade de storage do NetApp Cloud Volumes ONTAP no back-end	Pelo menos 300GB disponível
Nós de trabalho (requisito AWS EC2)	No total, pelo menos 3 nós de trabalho, com 4 núcleos vCPU e 12GB GB de RAM cada
Balanceador de carga	Tipo de serviço "LoadBalancer" disponível para envio de tráfego de entrada para serviços no cluster do ambiente operacional
FQDN	Um método para apontar o FQDN do Astra Control Center para o endereço IP balanceado de carga
Astra Trident (instalado como parte da descoberta de clusters do Kubernetes no NetApp BlueXP , anteriormente chamado Gerenciador de nuvem)	Astra Trident 22,10 ou mais recente instalado e configurado e NetApp ONTAP versão 9,8 ou mais recente como um back-end de storage

Componente	Requisito
Registro de imagens	O NetApp fornece um Registro que você pode usar para obter imagens de compilação do Astra Control Center: http://netappdownloads.jfrog.io/docker-astra-control-prod Entre em Contato com o suporte da NetApp para obter instruções sobre como usar esse Registro de imagem durante o processo de instalação do Astra Control Center. Se você não conseguir acessar o Registro de imagem do NetApp, você deve ter um Registro privado existente, como o AWS Elastic Container Registry (ECR), para o qual você pode enviar imagens de compilação do Astra Control Center. Você precisa fornecer o URL do Registro de imagens onde você vai carregar as imagens.  O cluster hospedado do Astra Control Center e o cluster gerenciado devem ter acesso ao mesmo Registro de imagem para poder fazer backup e restaurar aplicativos usando a imagem baseada em Restic.
Configuração Astra Trident/ONTAP	O Astra Control Center exige que uma classe de storage seja criada e definida como a classe de storage padrão. O Astra Control Center é compatível com as seguintes classes de storage do ONTAP Kubernetes criadas quando você importa o cluster do Kubernetes para o NetApp BlueXP (anteriormente conhecido como Cloud Manager). Eles são fornecidos pelo Astra Trident: • <code>vsaworkingenvironment-<>-ha-nas</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-ha-san</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-single-nas</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-single-san</code> <code>csi.trident.netapp.io</code>



Esses requisitos presumem que o Astra Control Center é a única aplicação em execução no ambiente operacional. Se o ambiente estiver executando aplicativos adicionais, ajuste esses requisitos mínimos de acordo.



O token de Registro da AWS expira em 12 horas, após o qual você terá que renovar o segredo de Registro de imagem do Docker.

Visão geral da implantação para AWS

Aqui está uma visão geral do processo de instalação do Astra Control Center for AWS com o Cloud Volumes ONTAP como um back-end de storage.

Cada uma destas etapas é explicada em mais detalhes abaixo.

1. [Certifique-se de que tem permissões IAM suficientes.](#)
2. [Instale um cluster RedHat OpenShift na AWS.](#)
3. [Configurar a AWS.](#)
4. [Configure o NetApp BlueXP para AWS.](#)
5. [Instalar o Astra Control Center for AWS.](#)

Certifique-se de que tem permissões IAM suficientes

Certifique-se de que você tenha funções e permissões suficientes do IAM que permitam instalar um cluster do RedHat OpenShift e um conector do NetApp BlueXP (antigo Gerenciador de nuvem).

```
https://docs.netapp.com/us-en/cloud-manager-setup-admin/concept-accounts-aws.html#initial-aws-credentials["Credenciais iniciais da AWS"]Consulte .
```

Instale um cluster RedHat OpenShift na AWS

Instale um cluster do RedHat OpenShift Container Platform na AWS.

Para obter instruções de instalação, ["Instalar um cluster na AWS no OpenShift Container Platform"](#) consulte .

Configurar a AWS

Em seguida, configure a AWS para criar uma rede virtual, configurar instâncias de computação EC2 e criar um bucket do AWS S3. Se não conseguir acessar o [Registro de imagem do NetApp Astra Control Center](#), você também precisará criar um ECR (Elastic Container Registry) para hospedar as imagens do Astra Control Center e enviar as imagens para esse Registro.

Siga a documentação da AWS para concluir as etapas a seguir. ["Documentação de instalação da AWS"](#)Consulte .

1. Crie uma rede virtual da AWS.
2. Analise as instâncias de computação do EC2. Isso pode ser um servidor bare metal ou VMs na AWS.
3. Se o tipo de instância ainda não corresponder aos requisitos mínimos de recursos do Astra para nós mestres e trabalhadores, altere o tipo de instância na AWS para atender aos requisitos do Astra. ["Requisitos do Astra Control Center"](#)Consulte a .
4. Crie pelo menos um bucket do AWS S3 para armazenar seus backups.
5. (Opcional) se não conseguir acessar ao [Registro de imagem NetApp](#), faça o seguinte:
 - a. Crie um AWS Elastic Container Registry (ECR) para hospedar todas as imagens do Astra Control Center.



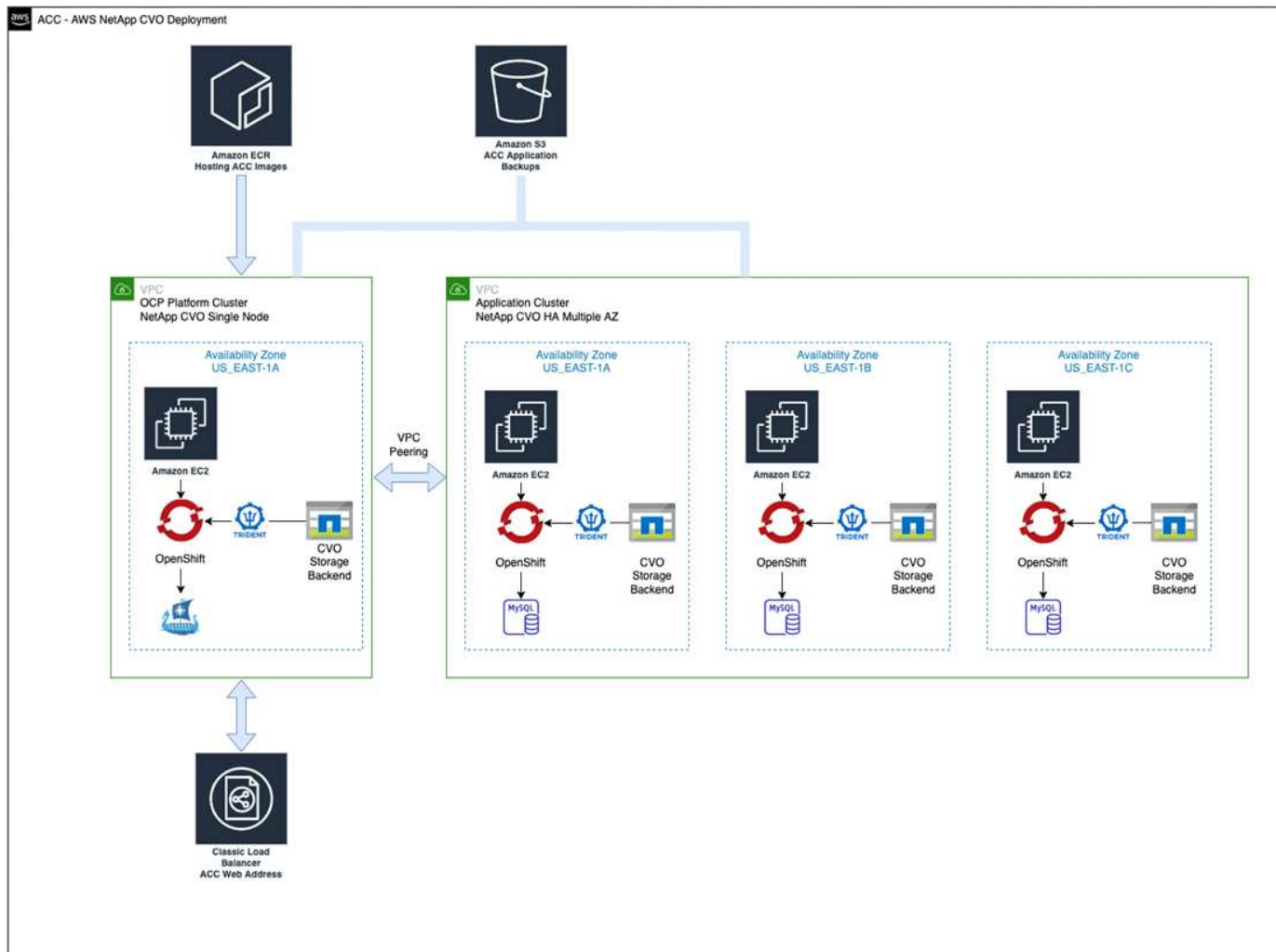
Se você não criar o ECR, o Astra Control Center não poderá acessar dados de monitoramento de um cluster que contém o Cloud Volumes ONTAP com um back-end da AWS. O problema é causado quando o cluster que você tenta descobrir e gerenciar usando o Astra Control Center não tem acesso ao AWS ECR.

- b. Envie as imagens do Astra Control Center para o Registro definido.



O token AWS Elastic Container Registry (ECR) expira após 12 horas e faz com que as operações de clone entre clusters falhem. Esse problema ocorre ao gerenciar um back-end de storage do Cloud Volumes ONTAP configurado para AWS. Para corrigir esse problema, autentique novamente com o ECR e gere um novo segredo para que as operações de clone sejam retomadas com sucesso.

Veja um exemplo de implantação da AWS:



Configure o NetApp BlueXP para AWS

Usando o NetApp BlueXP, crie uma área de trabalho, adicione um conector à AWS, crie um ambiente de trabalho e importe o cluster.

Siga a documentação do BlueXP para concluir as etapas a seguir. Veja o seguinte:

- ["Introdução ao Cloud Volumes ONTAP na AWS"](#).
- ["Crie um conector na AWS usando o BlueXP"](#)

Passos

1. Adicione suas credenciais ao BlueXP.
2. Criar um espaço de trabalho.

3. Adicione um conector para a AWS. Escolha a AWS como o provedor.
4. Crie um ambiente de trabalho para seu ambiente de nuvem.
 - a. Localização: "Amazon Web Services (AWS)"
 - b. Tipo: "Cloud Volumes ONTAP HA"
5. Importe o cluster OpenShift. O cluster se conectará ao ambiente de trabalho que você acabou de criar.
 - a. Veja os detalhes do cluster do NetApp selecionando **K8s > Lista de clusters > Detalhes do cluster**.
 - b. No canto superior direito, observe a versão Astra Trident.
 - c. Observe as classes de storage de cluster do Cloud Volumes ONTAP que mostram o NetApp como o provisionador.

Isso importa seu cluster Red Hat OpenShift e atribui a ele uma classe de armazenamento padrão. Você seleciona a classe de armazenamento. O Astra Trident é instalado automaticamente como parte do processo de importação e descoberta.

6. Observe todos os volumes e volumes persistentes nessa implantação do Cloud Volumes ONTAP.



O Cloud Volumes ONTAP pode operar como um único nó ou em alta disponibilidade. Se a HA estiver ativada, observe o status da HA e o status da implantação do nó em execução na AWS.

Instalar o Astra Control Center for AWS

Siga o padrão "[Instruções de instalação do Astra Control Center](#)".



A AWS usa o tipo de bucket Generic S3.

Implante o Astra Control Center no Google Cloud Platform

É possível implantar o Astra Control Center em um cluster autogerenciado do Kubernetes hospedado em uma nuvem pública do Google Cloud Platform (GCP).

O que você precisará para o GCP

Antes de implantar o Astra Control Center na GCP, você precisará dos seguintes itens:

- Licença do Astra Control Center. "[Requisitos de licenciamento do Astra Control Center](#)" Consulte a .
- "[Atender aos requisitos do Astra Control Center](#)".
- Conta do NetApp Cloud Central
- Se estiver usando OCP, Red Hat OpenShift Container Platform (OCP) 4,11 a 4,13
- Se estiver usando OCP, permissões do Red Hat OpenShift Container Platform (OCP) (no nível do namespace para criar pods)
- Conta de serviço do GCP com permissões que permitem criar buckets e conectores

Requisitos do ambiente operacional do GCP



Certifique-se de que o ambiente operacional escolhido para hospedar o Astra Control Center atenda aos requisitos básicos de recursos descritos na documentação oficial do ambiente.

O Astra Control Center requer os seguintes recursos, além dos requisitos de recursos do ambiente:

Componente	Requisito
Capacidade de storage do NetApp Cloud Volumes ONTAP no back-end	Pelo menos 300GB disponível
Nós de trabalho (requisito de computação do GCP)	No total, pelo menos 3 nós de trabalho, com 4 núcleos vCPU e 12GB GB de RAM cada
Balanceador de carga	Tipo de serviço "LoadBalancer" disponível para envio de tráfego de entrada para serviços no cluster do ambiente operacional
FQDN (ZONA DNS DO GCP)	Um método para apontar o FQDN do Astra Control Center para o endereço IP balanceado de carga
Astra Trident (instalado como parte da descoberta de clusters do Kubernetes no NetApp BlueXP , anteriormente chamado Gerenciador de nuvem)	Astra Trident 22,10 ou mais recente instalado e configurado e NetApp ONTAP versão 9,8 ou mais recente como um back-end de storage
Registro de imagens	O NetApp fornece um Registro que você pode usar para obter imagens de compilação do Astra Control Center: http://netappdownloads.jfrog.io/docker-astra-control-prod Entre em Contato com o suporte da NetApp para obter instruções sobre como usar esse Registro de imagem durante o processo de instalação do Astra Control Center. Se você não conseguir acessar o Registro de imagens do NetApp, você deve ter um Registro privado existente, como o Registro de contentores do Google, para o qual você pode enviar imagens de compilação do Astra Control Center. Você precisa fornecer o URL do Registro de imagens onde você vai carregar as imagens.  Você precisa habilitar o acesso anônimo para extrair imagens Restic para backups.

Componente	Requisito
Configuração Astra Trident/ONTAP	O Astra Control Center exige que uma classe de storage seja criada e definida como a classe de storage padrão. O Astra Control Center é compatível com as seguintes classes de storage do ONTAP Kubernetes criadas quando você importa o cluster do Kubernetes para o NetApp BlueXP . Eles são fornecidos pelo Astra Trident: • <code>vsaworkingenvironment-<>-ha-nas</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-ha-san</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-single-nas</code> <code>csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-single-san</code> <code>csi.trident.netapp.io</code>



Esses requisitos presumem que o Astra Control Center é a única aplicação em execução no ambiente operacional. Se o ambiente estiver executando aplicativos adicionais, ajuste esses requisitos mínimos de acordo.

Visão geral da implantação do GCP

Veja a seguir uma visão geral do processo de instalação do Astra Control Center em um cluster de OCP autogerenciado no GCP, com o Cloud Volumes ONTAP como um back-end de storage.

Cada uma destas etapas é explicada em mais detalhes abaixo.

1. [Instale um cluster RedHat OpenShift no GCP.](#)
2. [Crie um projeto do GCP e uma nuvem privada virtual.](#)
3. [Certifique-se de que tem permissões IAM suficientes.](#)
4. [Configurar o GCP.](#)
5. [Configurar o NetApp BlueXP para GCP.](#)
6. [Instalar o Astra Control Center no GCP.](#)

Instale um cluster RedHat OpenShift no GCP

A primeira etapa é instalar um cluster do RedHat OpenShift no GCP.

Para obter instruções de instalação, consulte o seguinte:

- ["Instalação de um cluster OpenShift no GCP"](#)
- ["Criando uma conta de serviço do GCP"](#)

Crie um projeto do GCP e uma nuvem privada virtual

Crie pelo menos um projeto do GCP e a Virtual Private Cloud (VPC).



OpenShift pode criar seus próprios grupos de recursos. Além disso, você também deve definir uma VPC do GCP. Consulte a documentação do OpenShift.

Você pode querer criar um grupo de recursos de cluster de plataforma e um grupo de recursos de cluster OpenShift de aplicativo de destino.

Certifique-se de que tem permissões IAM suficientes

Certifique-se de que você tenha funções e permissões suficientes do IAM que permitam instalar um cluster do RedHat OpenShift e um conector do NetApp BlueXP (antigo Gerenciador de nuvem).

```
https://docs.netapp.com/us-en/cloud-manager-setup-admin/task-creating-connectors-gcp.html#setting-up-permissions["Credenciais e permissões iniciais do GCP"]Consulte .
```

Configurar o GCP

Em seguida, configure o GCP para criar uma VPC, configurar instâncias de computação e criar um Google Cloud Object Storage. Se não conseguir acessar o [Registro de imagem do NetApp Astra Control Center](#), você também precisará criar um Registro de contendor do Google para hospedar as imagens do Astra Control Center e enviar as imagens para esse Registro.

Siga a documentação do GCP para concluir as etapas a seguir. Consulte Instalando o cluster OpenShift no GCP.

1. Crie um projeto do GCP e uma VPC no GCP que você planeja usar para o cluster do OCP com o back-end do CVO.
2. Revise as instâncias de computação. Isso pode ser um servidor bare metal ou VMs no GCP.
3. Se o tipo de instância ainda não corresponder aos requisitos mínimos de recursos do Astra para nós mestres e trabalhadores, altere o tipo de instância no GCP para atender aos requisitos do Astra. ["Requisitos do Astra Control Center"](#)Consulte a .
4. Crie pelo menos um bucket do GCP Cloud Storage para armazenar seus backups.
5. Crie um segredo, que é necessário para o acesso ao bucket.
6. (Opcional) se não conseguir aceder ao [Registro de imagem NetApp](#), faça o seguinte:
 - a. Crie um Registro de contêiner do Google para hospedar as imagens do Astra Control Center.
 - b. Configure o acesso do Google Container Registry para push/pull do Docker para todas as imagens do Astra Control Center.

Exemplo: As imagens do Astra Control Center podem ser enviadas para esse Registro inserindo o seguinte script:

```
gcloud auth activate-service-account <service account email address>
--key-file=<GCP Service Account JSON file>
```

Este script requer um arquivo de manifesto Astra Control Center e sua localização do Registro de imagens do Google.

Exemplo:

E

```
manifestfile=astra-control-center-<version>.manifest
GCP_CR_REGISTRY=<target image registry>
ASTRA_REGISTRY=<source Astra Control Center image registry>

while IFS= read -r image; do
    echo "image: $ASTRA_REGISTRY/$image $GCP_CR_REGISTRY/$image"
    root_image=${image%:*}
    echo $root_image
    docker pull $ASTRA_REGISTRY/$image
    docker tag $ASTRA_REGISTRY/$image $GCP_CR_REGISTRY/$image
    docker push $GCP_CR_REGISTRY/$image
done < astra-control-center-22.04.41.manifest
```

7. Configurar zonas DNS.

Configurar o NetApp BlueXP para GCP

Usando o NetApp BlueXP , crie uma área de trabalho, adicione um conector ao GCP, crie um ambiente de trabalho e importe o cluster.

Siga a documentação do BlueXP para concluir as etapas a seguir. ["Introdução ao Cloud Volumes ONTAP no GCP"](#) Consulte .

Antes de começar

- Acesso à conta do serviço do GCP com as permissões e funções necessárias do IAM

Passos

1. Adicione suas credenciais ao BlueXP . ["Adicionando contas do GCP"](#) Consulte .
2. Adicione um conector para o GCP.
 - a. Escolha "GCP" como Provedor.
 - b. Insira as credenciais do GCP. ["Criando um conector no GCP a partir do BlueXP"](#) Consulte .
 - c. Certifique-se de que o conector está a funcionar e mude para esse conector.
3. Crie um ambiente de trabalho para seu ambiente de nuvem.
 - a. Localização: "GCP"
 - b. Tipo: "Cloud Volumes ONTAP HA"
4. Importe o cluster OpenShift. O cluster se conectará ao ambiente de trabalho que você acabou de criar.
 - a. Veja os detalhes do cluster do NetApp selecionando **K8s > Lista de clusters > Detalhes do cluster**.
 - b. No canto superior direito, observe a versão do Trident.
 - c. Observe as classes de storage de cluster do Cloud Volumes ONTAP que mostram "NetApp" como o provisionador.

Isso importa seu cluster Red Hat OpenShift e atribui a ele uma classe de armazenamento padrão.

Você seleciona a classe de armazenamento. O Astra Trident é instalado automaticamente como parte do processo de importação e descoberta.

5. Observe todos os volumes e volumes persistentes nessa implantação do Cloud Volumes ONTAP.



O Cloud Volumes ONTAP pode operar como um nó único ou em alta disponibilidade (HA). Se a HA estiver ativada, observe o status de HA e o status de implantação de nós em execução no GCP.

Instalar o Astra Control Center no GCP

Siga o padrão "[Instruções de instalação do Astra Control Center](#)".



O GCP usa o tipo de bucket Generic S3.

1. Gere o segredo do Docker para extrair imagens para a instalação do Astra Control Center:

```
kubectl create secret docker-registry <secret name> --docker
-server=<Registry location> --docker-username=_json_key --docker
-password="$(cat <GCP Service Account JSON file>)" --namespace=pcloud
```

Implante o Astra Control Center no Microsoft Azure

É possível implantar o Astra Control Center em um cluster Kubernetes autogerenciado, hospedado em uma nuvem pública do Microsoft Azure.

O que você precisará para o Azure

Antes de implantar o Astra Control Center no Azure, você precisará dos seguintes itens:

- Licença do Astra Control Center. "[Requisitos de licenciamento do Astra Control Center](#)" Consulte a .
- "[Atender aos requisitos do Astra Control Center](#)".
- Conta do NetApp Cloud Central
- Se estiver usando OCP, Red Hat OpenShift Container Platform (OCP) 4,11 a 4,13
- Se estiver usando OCP, permissões do Red Hat OpenShift Container Platform (OCP) (no nível do namespace para criar pods)
- Credenciais do Azure com permissões que permitem criar buckets e conetores

Requisitos de ambiente operacional para o Azure

Certifique-se de que o ambiente operacional escolhido para hospedar o Astra Control Center atenda aos requisitos básicos de recursos descritos na documentação oficial do ambiente.

O Astra Control Center requer os seguintes recursos, além dos requisitos de recursos do ambiente:

"[Requisitos do ambiente operacional do Astra Control Center](#)" Consulte a .

Componente	Requisito
Capacidade de storage do NetApp Cloud Volumes ONTAP no back-end	Pelo menos 300GB disponível
Nós de trabalho (requisito de computação do Azure)	No total, pelo menos 3 nós de trabalho, com 4 núcleos vCPU e 12GB GB de RAM cada
Balanceador de carga	Tipo de serviço "LoadBalancer" disponível para envio de tráfego de entrada para serviços no cluster do ambiente operacional
FQDN (zona DNS do Azure)	Um método para apontar o FQDN do Astra Control Center para o endereço IP balanceado de carga
Astra Trident (instalado como parte da descoberta de clusters do Kubernetes no NetApp BlueXP)	Astra Trident 22,10 ou mais recente instalado e configurado e o NetApp ONTAP versão 9,8 ou mais recente será usado como back-end de storage
Registo de imagens	O NetApp fornece um Registro que você pode usar para obter imagens de compilação do Astra Control Center: http://netappdownloads.jfrog.io/docker-astra-control-prod Entre em Contato com o suporte da NetApp para obter instruções sobre como usar esse Registro de imagem durante o processo de instalação do Astra Control Center. Se você não conseguir acessar o Registro de imagem do NetApp, você deve ter um Registro privado existente, como o Azure Container Registry (ACR), para o qual você pode enviar imagens de compilação do Astra Control Center. Você precisa fornecer o URL do Registro de imagens onde você vai carregar as imagens.  Você precisa habilitar o acesso anônimo para extrair imagens Restic para backups.
Configuração Astra Trident/ONTAP	O Astra Control Center exige que uma classe de storage seja criada e definida como a classe de storage padrão. O Astra Control Center é compatível com as seguintes classes de storage do ONTAP Kubernetes criadas quando você importa o cluster do Kubernetes para o NetApp BlueXP . Eles são fornecidos pelo Astra Trident: • <code>vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-ha-san csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-single-nas csi.trident.netapp.io</code> • <code>vsaworkingenvironment-<>-single-san csi.trident.netapp.io</code>



Esses requisitos presumem que o Astra Control Center é a única aplicação em execução no ambiente operacional. Se o ambiente estiver executando aplicativos adicionais, ajuste esses requisitos mínimos de acordo.

Visão geral da implantação para o Azure

Aqui está uma visão geral do processo para instalar o Astra Control Center para Azure.

Cada uma destas etapas é explicada em mais detalhes abaixo.

1. [Instale um cluster RedHat OpenShift no Azure.](#)
2. [Criar grupos de recursos do Azure.](#)
3. [Certifique-se de que tem permissões IAM suficientes.](#)
4. [Configurar o Azure.](#)
5. [Configure o NetApp BlueXP \(anteriormente Gerenciador de nuvem\) para Azure.](#)
6. [Instalar e configurar o Astra Control Center para Azure.](#)

Instale um cluster RedHat OpenShift no Azure

O primeiro passo é instalar um cluster RedHat OpenShift no Azure.

Para obter instruções de instalação, consulte o seguinte:

- ["Instalando o cluster OpenShift no Azure".](#)
- ["Instalando uma conta do Azure".](#)

Criar grupos de recursos do Azure

Crie pelo menos um grupo de recursos do Azure.



OpenShift pode criar seus próprios grupos de recursos. Além disso, você também deve definir grupos de recursos do Azure. Consulte a documentação do OpenShift.

Você pode querer criar um grupo de recursos de cluster de plataforma e um grupo de recursos de cluster OpenShift de aplicativo de destino.

Certifique-se de que tem permissões IAM suficientes

Verifique se você tem funções e permissões suficientes do IAM que permitem instalar um cluster do RedHat OpenShift e um NetApp BlueXP Connector.

```
https://docs.netapp.com/us-en/cloud-manager-setup-admin/concept-accounts-azure.html["Credenciais e permissões do Azure"]Consulte .
```

Configurar o Azure

Em seguida, configure o Azure para criar uma rede virtual, configurar instâncias de computação e criar um contentor Blob do Azure. Se não conseguir acessar o [Registro de imagem do NetApp Astra Control Center](#), você também precisará criar um ACR (Azure Container Registry) para hospedar as imagens do Astra Control

Center e enviar as imagens para esse Registro.

Siga a documentação do Azure para concluir as etapas a seguir. ["Instalando o cluster OpenShift no Azure"](#) Consulte .

1. Crie uma rede virtual do Azure.
2. Revise as instâncias de computação. Isso pode ser um servidor bare metal ou VMs no Azure.
3. Se o tipo de instância ainda não corresponder aos requisitos mínimos de recursos do Astra para nós mestres e trabalhadores, altere o tipo de instância no Azure para atender aos requisitos do Astra. ["Requisitos do Astra Control Center"](#) Consulte a .
4. Crie pelo menos um contêiner do Blob do Azure para armazenar seus backups.
5. Crie uma conta de armazenamento. Você precisará de uma conta de storage para criar um contêiner para ser usado como um bucket no Astra Control Center.
6. Crie um segredo, que é necessário para o acesso ao bucket.
7. (Opcional) se não conseguir aceder ao [Registro de imagem NetApp](#), faça o seguinte:
 - a. Crie um ACR (Azure Container Registry) para hospedar as imagens do Astra Control Center.
 - b. Configure o acesso ACR para push/pull do Docker para todas as imagens do Astra Control Center.
 - c. Envie as imagens do Astra Control Center para esse Registro usando o seguinte script:

```
az acr login -n <AZ ACR URL/Location>
This script requires the Astra Control Center manifest file and your
Azure ACR location.
```

Exemplo:

```
manifestfile=astra-control-center-<version>.manifest
AZ_ACR_REGISTRY=<target image registry>
ASTRA_REGISTRY=<source Astra Control Center image registry>

while IFS= read -r image; do
    echo "image: $ASTRA_REGISTRY/$image $AZ_ACR_REGISTRY/$image"
    root_image=${image%:*}
    echo $root_image
    docker pull $ASTRA_REGISTRY/$image
    docker tag $ASTRA_REGISTRY/$image $AZ_ACR_REGISTRY/$image
    docker push $AZ_ACR_REGISTRY/$image
done < astra-control-center-22.04.41.manifest
```

8. Configurar zonas DNS.

Configure o NetApp BlueXP (anteriormente Gerenciador de nuvem) para Azure

Usando o BlueXP (antigo Gerenciador de nuvem), crie uma área de trabalho, adicione um conector ao Azure, crie um ambiente de trabalho e importe o cluster.

Siga a documentação do BlueXP para concluir as etapas a seguir. ["Introdução ao BlueXP no Azure"](#) Consulte

Antes de começar

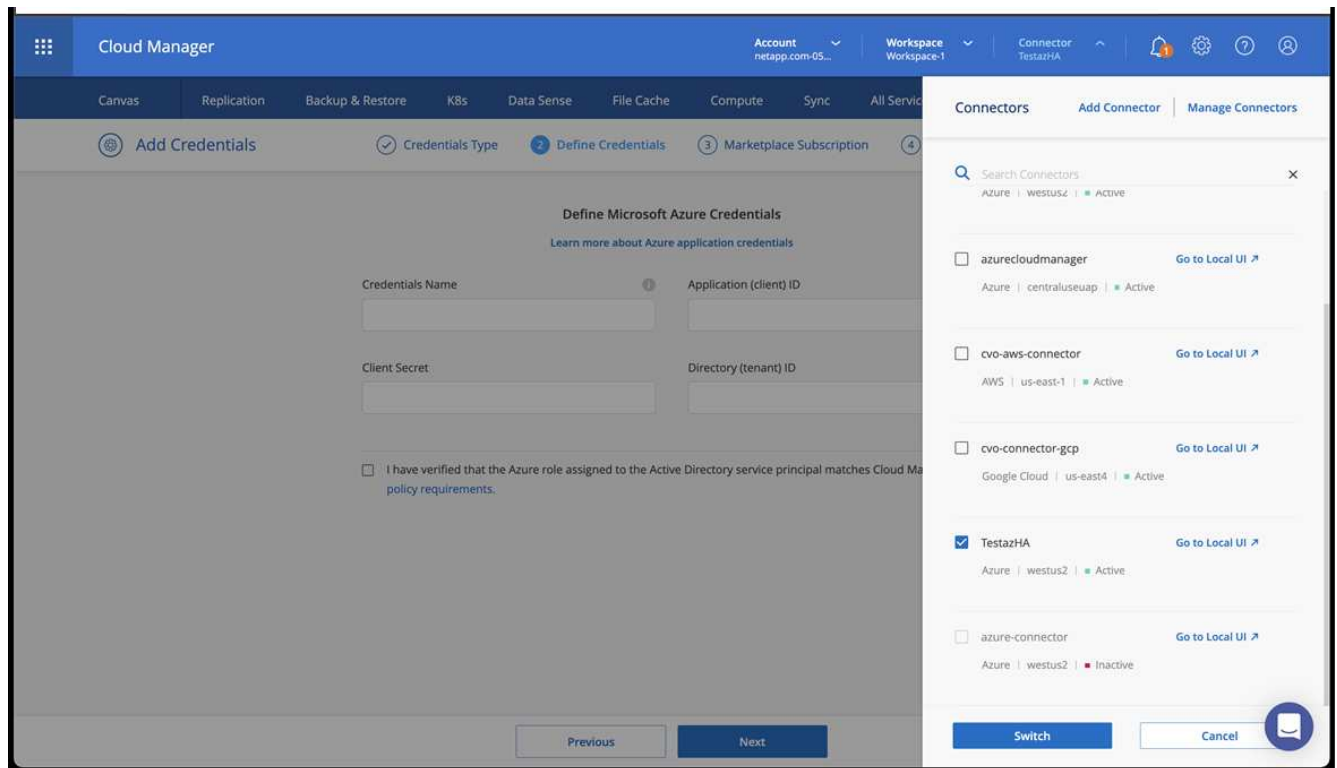
Acesso à conta do Azure com as permissões e funções necessárias do IAM

Passos

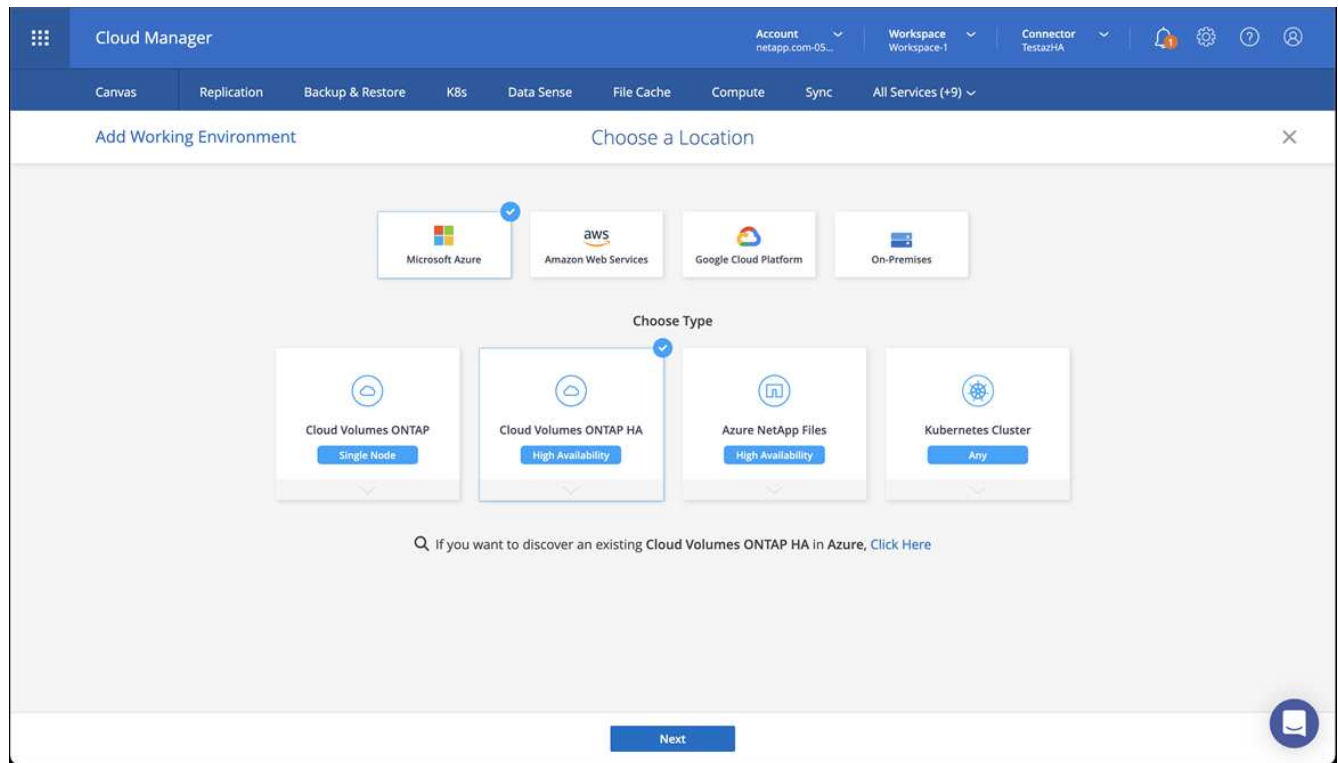
1. Adicione suas credenciais ao BlueXP .
2. Adicione um conector para o Azure. ["Políticas da BlueXP"](#) Consulte .
 - a. Escolha **Azure** como Provedor.
 - b. Insira as credenciais do Azure, incluindo o ID do aplicativo, o segredo do cliente e o ID do diretório (locatário).

```
https://docs.netapp.com/us-en/occm/task_creating_connectors_azure.html["Criando um conector no Azure a partir do BlueXP"]Consulte .
```

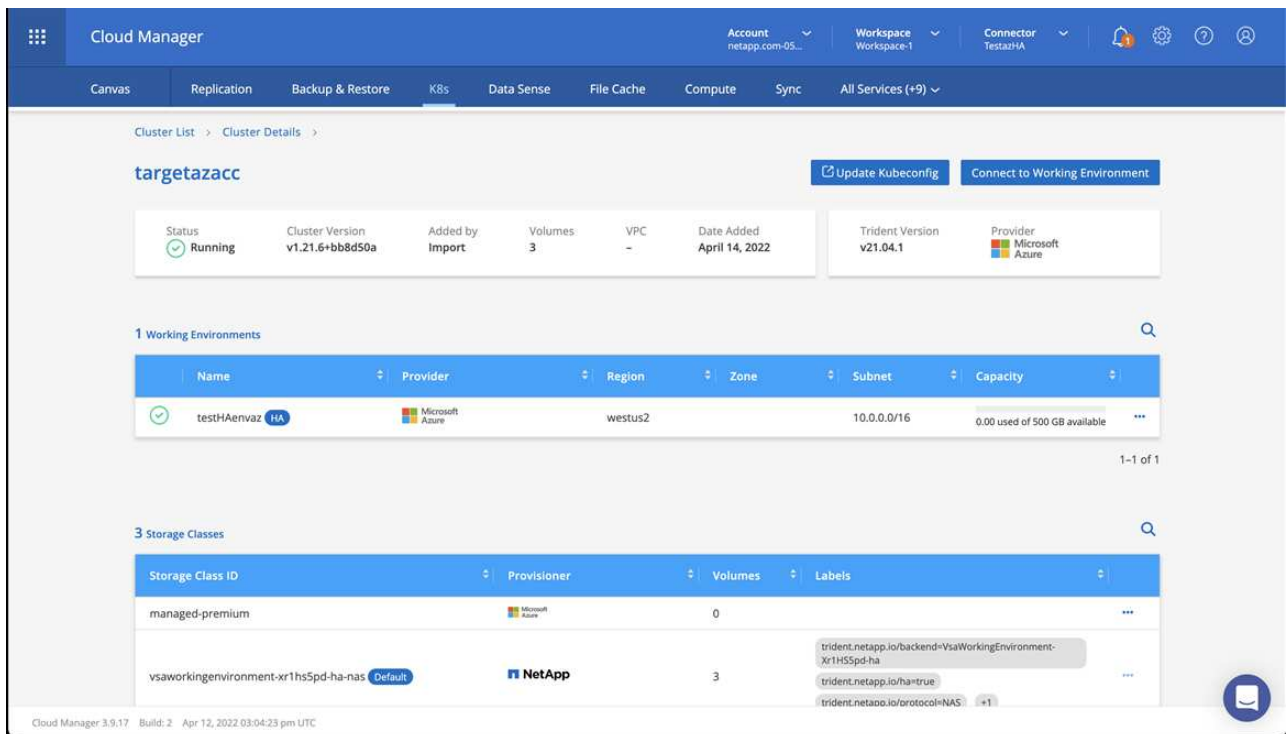
3. Certifique-se de que o conector está a funcionar e mude para esse conector.



4. Crie um ambiente de trabalho para seu ambiente de nuvem.
 - a. Localização: "Microsoft Azure".
 - b. Tipo: "Cloud Volumes ONTAP HA".



5. Importe o cluster OpenShift. O cluster se conetará ao ambiente de trabalho que você acabou de criar.
 - a. Veja os detalhes do cluster do NetApp selecionando **K8s > Lista de clusters > Detalhes do cluster**.



- b. No canto superior direito, observe a versão Astra Trident.
 - c. Observe as classes de storage de cluster do Cloud Volumes ONTAP que mostram o NetApp como o provisionador.
- Isso importa seu cluster Red Hat OpenShift e atribui uma classe de armazenamento padrão. Você

seleciona a classe de armazenamento. O Astra Trident é instalado automaticamente como parte do processo de importação e descoberta.

6. Observe todos os volumes e volumes persistentes nessa implantação do Cloud Volumes ONTAP.
7. O Cloud Volumes ONTAP pode operar como um único nó ou em alta disponibilidade. Se a HA estiver ativada, observe o status da HA e o status da implantação do nó em execução no Azure.

Instalar e configurar o Astra Control Center para Azure

Instalar o Astra Control Center com o padrão ["instruções de instalação"](#).

Usando o Astra Control Center, adicione um bucket do Azure. ["Configure o Astra Control Center e adicione buckets"](#) Consulte a .

Configure o Astra Control Center após a instalação

Dependendo do seu ambiente, pode haver configuração adicional necessária após a instalação do Astra Control Center.

Remover limitações de recursos

Alguns ambientes usam os objetos ResourceQuotes e LimitRanges para impedir que os recursos em um namespace consumam toda a CPU e memória disponíveis no cluster. O Astra Control Center não define limites máximos, por isso não estará em conformidade com esses recursos. Se o seu ambiente estiver configurado dessa forma, você precisará remover esses recursos dos namespaces onde você planeja instalar o Astra Control Center.

Você pode usar as etapas a seguir para recuperar e remover essas cotas e limites. Nestes exemplos, a saída do comando é mostrada imediatamente após o comando.

Passos

1. Obtenha as cotas de recursos no `netapp-acc` namespace (ou nome personalizado):

```
kubectl get quota -n [netapp-acc or custom namespace]
```

Resposta:

NAME	AGE	REQUEST	LIMIT
pods-high	16s	requests.cpu: 0/20, requests.memory: 0/100Gi	
		limits.cpu: 0/200, limits.memory: 0/1000Gi	
pods-low	15s	requests.cpu: 0/1, requests.memory: 0/1Gi	
		limits.cpu: 0/2, limits.memory: 0/2Gi	
pods-medium	16s	requests.cpu: 0/10, requests.memory: 0/20Gi	
		limits.cpu: 0/20, limits.memory: 0/200Gi	

2. Excluir todas as cotas de recursos por nome:

```
kubectl delete resourcequota pods-high -n [netapp-acc or custom namespace]
```

```
kubectl delete resourcequota pods-low -n [netapp-acc or custom namespace]
```

```
kubectl delete resourcequota pods-medium -n [netapp-acc or custom namespace]
```

3. Obtenha os intervalos de limite no netapp-acc namespace (ou nome personalizado):

```
kubectl get limits -n [netapp-acc or custom namespace]
```

Resposta:

NAME	CREATED AT
cpu-limit-range	2022-06-27T19:01:23Z

4. Eliminar os intervalos de limite por nome:

```
kubectl delete limitrange cpu-limit-range -n [netapp-acc or custom namespace]
```

Adicione um certificado TLS personalizado

O Astra Control Center usa um certificado TLS autoassinado por padrão para o tráfego do controlador de entrada (somente em certas configurações) e autenticação da IU da Web com navegadores da Web. Você pode remover o certificado TLS autoassinado existente e substituí-lo por um certificado TLS assinado por uma autoridade de certificação (CA).



O certificado auto-assinado padrão é usado para dois tipos de conexões:

- Conexões HTTPS com a IU da Web do Astra Control Center
- Tráfego do controlador de entrada (somente se a `ingressType: "AccTraefik"` propriedade foi definida no `astra_control_center.yaml` arquivo durante a instalação do Astra Control Center)

A substituição do certificado TLS padrão substitui o certificado usado para autenticação dessas conexões.

Antes de começar

- Cluster do Kubernetes com Astra Control Center instalado
- Acesso administrativo a um shell de comando no cluster para executar `kubectl` comandos
- Arquivos de chave privada e certificado da CA

Remova o certificado autoassinado

Remova o certificado TLS autoassinado existente.

1. Usando SSH, faça login no cluster do Kubernetes que hospeda o Astra Control Center como usuário administrativo.
2. Localize o segredo TLS associado ao certificado atual usando o seguinte comando, substituindo `<ACC-deployment-namespace>` pelo namespace de implantação do Astra Control Center:

```
kubectl get certificate -n <ACC-deployment-namespace>
```

3. Exclua o segredo e o certificado atualmente instalados usando os seguintes comandos:

```
kubectl delete cert cert-manager-certificates -n <ACC-deployment-namespace>
```

```
kubectl delete secret secure-testing-cert -n <ACC-deployment-namespace>
```

Adicione um novo certificado usando a linha de comando

Adicione um novo certificado TLS assinado por uma CA.

1. Use o comando a seguir para criar o novo segredo TLS com a chave privada e os arquivos de certificado da CA, substituindo os argumentos entre colchetes pelas informações apropriadas:

```
kubectl create secret tls <secret-name> --key <private-key-filename>
--cert <certificate-filename> -n <ACC-deployment-namespace>
```

2. Use o comando e exemplo a seguir para editar o arquivo CRD (Custom Resource Definition) do cluster e altere o `spec.selfSigned` valor para `spec.ca.secretName` se referir ao segredo TLS criado anteriormente:

```
kubectl edit clusterissuers.cert-manager.io/cert-manager-certificates -n
<ACC-deployment-namespace>
```

CRD:

```
#spec:
#  selfSigned: {}

spec:
  ca:
    secretName: <secret-name>
```

3. Use o comando e exemplo de saída a seguir para validar se as alterações estão corretas e o cluster está pronto para validar certificados, substituindo <ACC-deployment-namespace> pelo namespace de implantação do Astra Control Center:

```
kubectl describe clusterissuers.cert-manager.io/cert-manager-
certificates -n <ACC-deployment-namespace>
```

Resposta:

```
Status:
  Conditions:
    Last Transition Time: 2021-07-01T23:50:27Z
    Message:             Signing CA verified
    Reason:              KeyPairVerified
    Status:              True
    Type:                Ready
  Events:                <none>
```

4. Crie o `certificate.yaml` arquivo usando o exemplo a seguir, substituindo os valores de espaço reservado entre colchetes por informações apropriadas:

```

apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  <strong>name: <certificate-name></strong>
  namespace: <ACC-deployment-namespace>
spec:
  <strong>secretName: <certificate-secret-name></strong>
  duration: 2160h # 90d
  renewBefore: 360h # 15d
  dnsNames:
    <strong>- <astra.dnsname.example.com></strong> #Replace with the
correct Astra Control Center DNS address
  issuerRef:
    kind: ClusterIssuer
    name: cert-manager-certificates

```

5. Crie o certificado usando o seguinte comando:

```
kubectl apply -f certificate.yaml
```

6. Usando o comando a seguir e exemplo de saída, valide que o certificado foi criado corretamente e com os argumentos especificados durante a criação (como nome, duração, prazo de renovação e nomes DNS).

```
kubectl describe certificate -n <ACC-deployment-namespace>
```

Resposta:

```

Spec:
  Dns Names:
    astra.example.com
  Duration: 125h0m0s
  Issuer Ref:
    Kind:      ClusterIssuer
    Name:      cert-manager-certificates
  Renew Before: 61h0m0s
  Secret Name:  <certificate-secret-name>
Status:
  Conditions:
    Last Transition Time: 2021-07-02T00:45:41Z
    Message:             Certificate is up to date and has not expired
    Reason:              Ready
    Status:              True
    Type:               Ready
  Not After:            2021-07-07T05:45:41Z
  Not Before:           2021-07-02T00:45:41Z
  Renewal Time:         2021-07-04T16:45:41Z
  Revision:             1
  Events:               <none>

```

7. Edite o TLS armazena o CRD para apontar para o novo nome secreto do certificado usando o comando e o exemplo a seguir, substituindo os valores do espaço reservado entre parênteses> por informações apropriadas

```
kubectl edit tlsstores.traefik.io -n <ACC-deployment-namespace>
```

CRD:

```

...
spec:
  defaultCertificate:
    secretName: <certificate-secret-name>

```

8. Edite a opção TLS de CRD de entrada para apontar para o novo segredo de certificado usando o comando e o exemplo a seguir, substituindo os valores de espaço reservado entre colchetes> por informações apropriadas:

```
kubectl edit ingressroutes.traefik.io -n <ACC-deployment-namespace>
```

CRD:

```
...
tls:
  secretName: <certificate-secret-name>
```

9. Usando um navegador da Web, navegue até o endereço IP de implantação do Astra Control Center.
10. Verifique se os detalhes do certificado correspondem aos detalhes do certificado que você instalou.
11. Exporte o certificado e importe o resultado para o gerenciador de certificados no navegador da Web.

Configure o Astra Control Center

Depois de instalar o Astra Control Center, fazer login na IU e alterar sua senha, você deseja configurar uma licença, adicionar clusters, habilitar a autenticação, gerenciar armazenamento e adicionar buckets.

Tarefas

- [Adicione uma licença para o Astra Control Center](#)
- [Prepare seu ambiente para gerenciamento de clusters com o Astra Control](#)
- [Adicionar cluster](#)
- [Ativar a autenticação no back-end de storage do ONTAP](#)
- [Adicionar um back-end de storage](#)
- [Adicione um balde](#)

Adicione uma licença para o Astra Control Center

Quando você instala o Astra Control Center, uma licença de avaliação incorporada já está instalada. Se você estiver avaliando o Astra Control Center, ignore esta etapa.

Você pode adicionar uma nova licença usando a IU do Astra Control ou ["API Astra Control"](#)o .

As licenças do Astra Control Center medem recursos de CPU usando unidades de CPU Kubernetes e contam os recursos de CPU atribuídos aos nós de trabalho de todos os clusters gerenciados do Kubernetes. As licenças são baseadas no uso do vCPU. Para obter mais informações sobre como as licenças são calculadas, ["Licenciamento"](#)consulte .



Se a instalação aumentar para exceder o número licenciado de unidades de CPU, o Astra Control Center impedirá que você gerencie novas aplicações. É apresentado um alerta quando a capacidade é ultrapassada.



Para atualizar uma avaliação existente ou uma licença completa, ["Atualizar uma licença existente"](#)consulte .

Antes de começar

- Acesso a uma instância recém-instalada do Astra Control Center.
- Permissões de função de administrador.
- A ["Ficheiro de licença do NetApp"](#) (NLF).

Passos

1. Faça login na IU do Astra Control Center.
2. Selecione **conta > Licença**.
3. Selecione **Adicionar licença**.
4. Navegue até o arquivo de licença (NLF) que você baixou.
5. Selecione **Adicionar licença**.

A página **Account > License** exibe as informações da licença, data de validade, número de série da licença, ID da conta e unidades CPU usadas.



Se você tiver uma licença de avaliação e não estiver enviando dados para o AutoSupport, lembre-se de armazenar o ID da conta para evitar a perda de dados em caso de falha do Centro de Controle Astra.

Prepare seu ambiente para gerenciamento de clusters com o Astra Control

Você deve garantir que as seguintes condições de pré-requisito sejam atendidas antes de adicionar um cluster. Você também deve executar verificações de qualificação para garantir que seu cluster esteja pronto para ser adicionado ao Astra Control Center e criar funções para gerenciamento de clusters.

Antes de começar

- Verifique se os nós de trabalho no cluster estão configurados com os drivers de armazenamento apropriados para que os pods possam interagir com o armazenamento de back-end.
- Seu ambiente atende ao "[requisitos do ambiente operacional](#)" Astra Trident e Astra Control Center.
- Se você estiver adicionando o cluster usando um arquivo kubeconfig que faça referência a uma Autoridade de certificação privada (CA), adicione a seguinte linha à `cluster` seção do arquivo kubeconfig. Isso permite que o Astra Control adicione o cluster:

```
insecure-skip-tls-verify: true
```

- Uma versão do Astra Trident "[Compatível com Astra Control Center](#)" instalada:



Você pode "[Implante o Astra Trident](#)" usar o operador Astra Trident (manualmente ou usando o gráfico Helm) ou `tridentctl`. Antes de instalar ou atualizar o Astra Trident, revise o "[interfaces suportadas, backends e configurações de host](#)".

- **Back-end de storage do Astra Trident configurado:** Pelo menos um back-end de storage do Astra Trident precisa estar "[configurado](#)" no cluster.
- **Classes de storage Astra Trident configuradas:** Pelo menos uma classe de storage Astra Trident deve estar "[configurado](#)" no cluster. Se uma classe de armazenamento padrão estiver configurada, certifique-se de que é a única classe de armazenamento que tem a anotação padrão.
- **Controlador de volume e classe de snapshot de volume Astra Trident instalado e configurado:** A controladora de volume deve ser "[instalado](#)" para que os snapshots possam ser criados no Astra Control. Pelo menos um Astra Trident `VolumeSnapshotClass` foi "[configuração](#)" feito por um administrador.
- **Kubeconfig acessível:** Você tem acesso ao "[cluster predefinido kubeconfig](#)" "[você configurou durante a instalação](#)" that .

- *** Credenciais ONTAP*:** Você precisa de credenciais ONTAP e um superusuário e ID de usuário definidos no sistema ONTAP de backup para fazer backup e restaurar aplicativos com o Astra Control Center.

Execute os seguintes comandos na linha de comando ONTAP:

```
export-policy rule modify -vserver <storage virtual machine name>  
-policyname <policy name> -ruleindex 1 -superuser sys  
export-policy rule modify -vserver <storage virtual machine name>  
-policyname <policy name> -ruleindex 1 -anon 65534
```

- **Somente Rancher:** Ao gerenciar clusters de aplicativos em um ambiente Rancher, modifique o contexto padrão do cluster de aplicativos no arquivo kubeconfig fornecido pelo Rancher para usar um contexto de plano de controle em vez do contexto do servidor da API Rancher. Isso reduz a carga no servidor de API Rancher e melhora o desempenho.

Execute verificações de qualificação

Execute as seguintes verificações de qualificação para garantir que o cluster esteja pronto para ser adicionado ao Astra Control Center.

Passos

1. Verifique a versão Astra Trident.

```
kubectl get tridentversions -n trident
```

Se o Astra Trident existir, você verá uma saída semelhante à seguinte:

NAME	VERSION
trident	23.XX.X

Se o Astra Trident não existir, você verá uma saída semelhante à seguinte:

```
error: the server doesn't have a resource type "tridentversions"
```



Se o Astra Trident não estiver instalado ou a versão instalada não for a mais recente, você precisará instalar a versão mais recente do Astra Trident antes de continuar. Consulte o ["Documentação do Astra Trident"](#) para obter instruções.

2. Certifique-se de que os pods estão em execução:

```
kubectl get pods -n trident
```

3. Determine se as classes de storage estão usando os drivers Astra Trident compatíveis. O nome do provisionador deve ser `csi.trident.netapp.io`. Veja o exemplo a seguir:

```
kubectl get sc
```

Resposta da amostra:

NAME	PROVISIONER	RECLAIMPOLICY
VOLUMEBINDINGMODE	ALLOWVOLUMEEXPANSION	AGE
ontap-gold (default)	csi.trident.netapp.io	Delete
true	5d23h	Immediate

Crie uma função de cluster kubeconfig

Você pode, opcionalmente, criar uma função de administrador de permissão limitada ou expandida para o Astra Control Center. Este não é um procedimento necessário para a configuração do Astra Control Center, uma vez que já configurou um kubeconfig como parte do ["processo de instalação"](#).

Este procedimento ajuda você a criar um kubeconfig separado se qualquer um dos seguintes cenários se aplicar ao seu ambiente:

- Você deseja limitar as permissões do Astra Control nos clusters que ele gerencia
- Você usa vários contextos e não pode usar o kubeconfig padrão do Astra Control configurado durante a instalação ou uma função limitada com um único contexto não funcionará em seu ambiente

Antes de começar

Certifique-se de que tem o seguinte para o cluster que pretende gerir antes de concluir as etapas do procedimento:

- kubectl v1,23 ou posterior instalado
- Acesso kubectl ao cluster que você pretende adicionar e gerenciar com o Astra Control Center



Para esse procedimento, você não precisa de acesso kubectl ao cluster que está executando o Astra Control Center.

- Um kubeconfig ativo para o cluster que pretende gerir com direitos de administrador de cluster para o contexto ativo

Passos

1. Criar uma conta de serviço:

a. Crie um arquivo de conta de serviço `astracontrol-service-account.yaml` chamado .

Ajuste o nome e o namespace conforme necessário. Se as alterações forem feitas aqui, você deve aplicar as mesmas alterações nas etapas a seguir.

```
<strong>astracontrol-service-account.yaml</strong>
```

+

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: astracontrol-service-account
  namespace: default
```

a. Aplique a conta de serviço:

```
kubectl apply -f astracontrol-service-account.yaml
```

2. Crie uma das seguintes funções de cluster com permissões suficientes para que um cluster seja gerenciado pelo Astra Control:

- **Função de cluster limitada:** Essa função contém as permissões mínimas necessárias para que um cluster seja gerenciado pelo Astra Control:

Expanda para obter passos

- i. Crie um ClusterRole arquivo chamado, por exemplo `astra-admin-account.yaml`, .

Ajuste o nome e o namespace conforme necessário. Se as alterações forem feitas aqui, você deve aplicar as mesmas alterações nas etapas a seguir.

```
<strong>astra-admin-account.yaml</strong>
```

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: astra-admin-account
rules:

# Get, List, Create, and Update all resources
# Necessary to backup and restore all resources in an app
- apiGroups:
  - '*'
  resources:
  - '*'
  verbs:
  - get
  - list
  - create
  - patch

# Delete Resources
# Necessary for in-place restore and AppMirror failover
- apiGroups:
  - ""
  - apps
  - autoscaling
  - batch
  - crd.projectcalico.org
  - extensions
  - networking.k8s.io
  - policy
  - rbac.authorization.k8s.io
  - snapshot.storage.k8s.io
  - trident.netapp.io
  resources:
  - configmaps
  - cronjobs
  - daemonsets
```

```

- deployments
- horizontalpodautoscalers
- ingresses
- jobs
- namespaces
- networkpolicies
- persistentvolumeclaims
- poddisruptionbudgets
- pods
- podtemplates
- podsecuritypolicies
- replicaset
- replicationcontrollers
- replicationcontrollers/scale
- rolebindings
- roles
- secrets
- serviceaccounts
- services
- statefulsets
- tridentmirrorrelationships
- tridentnapshotinfos
- volumesnapshots
- volumesnapshotcontents
verbs:
- delete

# Watch resources
# Necessary to monitor progress
- apiGroups:
  - ""
  resources:
  - pods
  - replicationcontrollers
  - replicationcontrollers/scale
  verbs:
  - watch

# Update resources
- apiGroups:
  - ""
  - build.openshift.io
  - image.openshift.io
  resources:
  - builds/details
  - replicationcontrollers

```

```
- replicationcontrollers/scale
- imagestreams/layers
- imagestreamtags
- imagetags
verbs:
- update
```

```
# Use PodSecurityPolicies
- apiGroups:
  - extensions
  - policy
resources:
- podsecuritypolicies
verbs:
- use
```

- ii. (Somente para clusters OpenShift) Append o seguinte no final `astra-admin-account.yaml` do arquivo ou após a `# Use PodSecurityPolicies` seção:

```
# OpenShift security
- apiGroups:
  - security.openshift.io
resources:
- securitycontextconstraints
verbs:
- use
```

- iii. Aplique a função de cluster:

```
kubectl apply -f astra-admin-account.yaml
```

- **Função de cluster expandida:** Essa função contém permissões expandidas para um cluster a ser gerenciado pelo Astra Control. Você pode usar essa função se você usar vários contextos e não puder usar o kubeconfig padrão do Astra Control configurado durante a instalação ou uma função limitada com um único contexto não funcionará em seu ambiente:



As etapas a seguir `ClusterRole` são um exemplo geral do Kubernetes. Consulte a documentação da distribuição do Kubernetes para obter instruções específicas para o seu ambiente.

Expanda para obter passos

- i. Crie um ClusterRole arquivo chamado, por exemplo `astra-admin-account.yaml`, .

Ajuste o nome e o namespace conforme necessário. Se as alterações forem feitas aqui, você deve aplicar as mesmas alterações nas etapas a seguir.

```
<strong>astra-admin-account.yaml</strong>
```

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: astra-admin-account
rules:
- apiGroups:
  - '*'
  resources:
  - '*'
  verbs:
  - '*'
- nonResourceURLs:
  - '*'
  verbs:
  - '*'
```

- ii. Aplique a função de cluster:

```
kubectl apply -f astra-admin-account.yaml
```

3. Crie a vinculação de função de cluster para a função de cluster à conta de serviço:

- a. Crie um ClusterRoleBinding arquivo chamado `astracontrol-clusterrolebinding.yaml`.

Ajuste quaisquer nomes e namespaces modificados ao criar a conta de serviço conforme necessário.

```
<strong>astracontrol-clusterrolebinding.yaml</strong>
```

+

```

apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: astracontrol-admin
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: astra-admin-account
subjects:
- kind: ServiceAccount
  name: astracontrol-service-account
  namespace: default

```

- a. Aplicar a vinculação de funções do cluster:

```
kubectl apply -f astracontrol-clusterrolebinding.yaml
```

4. Crie e aplique o segredo do token:

- a. Crie um arquivo secreto de token `secret-astracontrol-service-account.yaml` chamado .

```
<strong>secret-astracontrol-service-account.yaml</strong>
```

```

apiVersion: v1
kind: Secret
metadata:
  name: secret-astracontrol-service-account
  namespace: default
  annotations:
    kubernetes.io/service-account.name: "astracontrol-service-account"
type: kubernetes.io/service-account-token

```

- b. Aplique o segredo do token:

```
kubectl apply -f secret-astracontrol-service-account.yaml
```

5. Adicione o segredo do token à conta de serviço adicionando seu nome ao `secrets` array (a última linha no exemplo a seguir):

```
kubectl edit sa astracontrol-service-account
```

```

apiVersion: v1
imagePullSecrets:
- name: astracontrol-service-account-dockercfg-48xhx
kind: ServiceAccount
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |

{"apiVersion":"v1","kind":"ServiceAccount","metadata":{"annotations":{},"name":"astracontrol-service-account","namespace":"default"},"creationTimestamp":"2023-06-14T15:25:45Z","name":"astracontrol-service-account","namespace":"default","resourceVersion":"2767069","uid":"2ce068c4-810e-4a96-ada3-49cbf9ec3f89"}
secrets:
- name: astracontrol-service-account-dockercfg-48xhx
<strong>- name: secret-astracontrol-service-account</strong>

```

6. Liste os segredos da conta de serviço, substituindo <context> pelo contexto correto para sua instalação:

```

kubectl get serviceaccount astracontrol-service-account --context
<context> --namespace default -o json

```

O final da saída deve ser semelhante ao seguinte:

```

"secrets": [
{ "name": "astracontrol-service-account-dockercfg-48xhx"},
{ "name": "secret-astracontrol-service-account"}
]

```

Os índices para cada elemento no secrets array começam com 0. No exemplo acima, o índice para astracontrol-service-account-dockercfg-48xhx seria 0 e o índice para secret-astracontrol-service-account seria 1. Na sua saída, anote o número do índice para o segredo da conta de serviço. Você precisará deste número de índice na próxima etapa.

7. Gere o kubeconfig da seguinte forma:

- a. Crie um create-kubeconfig.sh arquivo. Substitua TOKEN_INDEX no início do script a seguir pelo valor correto.

```

<strong>create-kubeconfig.sh</strong>

```

```

# Update these to match your environment.
# Replace TOKEN_INDEX with the correct value
# from the output in the previous step. If you
# didn't change anything else above, don't change
# anything else here.

SERVICE_ACCOUNT_NAME=astraccontrol-service-account
NAMESPACE=default
NEW_CONTEXT=astraccontrol
KUBECONFIG_FILE='kubeconfig-sa'

CONTEXT=$(kubectl config current-context)

SECRET_NAME=$(kubectl get serviceaccount ${SERVICE_ACCOUNT_NAME} \
  --context ${CONTEXT} \
  --namespace ${NAMESPACE} \
  -o jsonpath='{.secrets[TOKEN_INDEX].name}')
TOKEN_DATA=$(kubectl get secret ${SECRET_NAME} \
  --context ${CONTEXT} \
  --namespace ${NAMESPACE} \
  -o jsonpath='{.data.token}')

TOKEN=$(echo ${TOKEN_DATA} | base64 -d)

# Create dedicated kubeconfig
# Create a full copy
kubectl config view --raw > ${KUBECONFIG_FILE}.full.tmp

# Switch working context to correct context
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp config use-context
${CONTEXT}

# Minify
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp \
  config view --flatten --minify > ${KUBECONFIG_FILE}.tmp

# Rename context
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  rename-context ${CONTEXT} ${NEW_CONTEXT}

# Create token user
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-credentials ${CONTEXT}-${NAMESPACE}-token-user \
  --token ${TOKEN}

# Set context to use token user
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \

```

```
set-context ${NEW_CONTEXT} --user ${CONTEXT}-${NAMESPACE}-token
-user

# Set context to correct namespace
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-context ${NEW_CONTEXT} --namespace ${NAMESPACE}

# Flatten/minify kubeconfig
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  view --flatten --minify > ${KUBECONFIG_FILE}

# Remove tmp
rm ${KUBECONFIG_FILE}.full.tmp
rm ${KUBECONFIG_FILE}.tmp
```

b. Forneça os comandos para aplicá-los ao cluster do Kubernetes.

```
source create-kubeconfig.sh
```

8. (Opcional) Renomear o kubeconfig para um nome significativo para o cluster.

```
mv kubeconfig-sa YOUR_CLUSTER_NAME_kubeconfig
```

O que se segue?

Agora que você verificou que os pré-requisitos foram atendidos, você está pronto para [adicione um cluster](#).

Adicionar cluster

Para começar a gerenciar suas aplicações, adicione um cluster do Kubernetes e gerencie-o como um recurso de computação. Você precisa adicionar um cluster para Astra Control Center para descobrir suas aplicações Kubernetes.



Recomendamos que o Astra Control Center gerencie o cluster em que ele é implantado primeiro antes de adicionar outros clusters ao Astra Control Center para gerenciar. Ter o cluster inicial sob gerenciamento é necessário enviar dados do Kubemetrics e dados associados ao cluster para métricas e solução de problemas.

Antes de começar

- Antes de adicionar um cluster, revise e execute o [tarefas pré-requisitos](#) necessário .

Passos

1. Navegue pelo menu Dashboard ou clusters:
 - Em **Dashboard** no Resumo de recursos, selecione **Add** no painel clusters.
 - Na área de navegação à esquerda, selecione **clusters** e, em seguida, selecione **Adicionar cluster** na

página clusters.

2. Na janela **Add Cluster** que se abre, carregue um `kubeconfig.yaml` ficheiro ou cole o conteúdo de um `kubeconfig.yaml` ficheiro.



O `kubeconfig.yaml` arquivo deve incluir **somente a credencial de cluster para um cluster**.



Se você criar seu próprio `kubeconfig` arquivo, você deve definir apenas **um** elemento de contexto nele. "[Documentação do Kubernetes](#)" Consulte para obter informações sobre a criação `kubeconfig` de ficheiros. Se você criou um `kubeconfig` para uma função de cluster limitada usando o [processo acima](#) , certifique-se de carregar ou colar esse `kubeconfig` nesta etapa.

3. Forneça um nome de credencial. Por padrão, o nome da credencial é preenchido automaticamente como o nome do cluster.
4. Selecione **seguinte**.
5. Selecione a classe de armazenamento padrão a ser usada para este cluster Kubernetes e selecione **Next**.



Você deve selecionar uma classe de storage Astra Trident com o suporte de storage ONTAP.

6. Revise as informações e, se tudo estiver bem, selecione **Adicionar**.

Resultado

O cluster entra no estado **Descobrendo** e depois muda para **saudável**. Agora você está gerenciando o cluster com Astra Control Center.



Depois de adicionar um cluster a ser gerenciado no Astra Control Center, talvez demore alguns minutos para implantar o operador de monitoramento. Até então, o ícone de notificação fica vermelho e Registra um evento **Falha na verificação do status do agente de monitoramento**. Você pode ignorar isso, porque o problema resolve quando o Astra Control Center obtém o status correto. Se o problema não resolver em alguns minutos, vá para o cluster e execute `oc get pods -n netapp-monitoring` como ponto de partida. Você precisará examinar os logs do operador de monitoramento para depurar o problema.

Ativar a autenticação no back-end de storage do ONTAP

O Astra Control Center oferece dois modos de autenticação de um back-end do ONTAP:

- **Autenticação baseada em credenciais:** O nome de usuário e senha para um usuário do ONTAP com as permissões necessárias. Você deve usar uma função de login de segurança pré-definida, como `admin` ou `vsadmin` para garantir a máxima compatibilidade com as versões do ONTAP.
- **Autenticação baseada em certificado:** O Astra Control Center também pode se comunicar com um cluster ONTAP usando um certificado instalado no back-end. Você deve usar o certificado de cliente, a chave e o certificado de CA confiável, se usado (recomendado).

Você pode atualizar posteriormente os backends existentes para passar de um tipo de autenticação para outro método. Apenas um método de autenticação é suportado de cada vez.

Ative a autenticação baseada em credenciais

O Astra Control Center requer as credenciais para um cluster com escopo `admin` para se comunicar com o back-end do ONTAP. Você deve usar funções padrão e predefinidas, `admin` como . Isso garante compatibilidade direta com futuras versões do ONTAP que podem expor APIs de recursos a serem usadas por futuras versões do Astra Control Center.



Uma função de login de segurança personalizada pode ser criada e usada com o Astra Control Center, mas não é recomendada.

Uma definição de backend de exemplo se parece com esta:

```
{
  "version": 1,
  "backendName": "ExampleBackend",
  "storageDriverName": "ontap-nas",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "svm": "svm_nfs",
  "username": "admin",
  "password": "secret"
}
```

A definição de back-end é o único lugar onde as credenciais são armazenadas em texto simples. A criação ou atualização de um backend é a única etapa que requer conhecimento das credenciais. Como tal, é uma operação somente de administração, realizada pelo Kubernetes ou pelo administrador de storage.

Ativar autenticação baseada em certificado

O Centro de Controle Astra pode usar certificados para se comunicar com backends ONTAP novos e existentes. Você deve inserir as seguintes informações na definição de back-end.

- `clientCertificate`: Certificado do cliente.
- `clientPrivateKey`: Chave privada associada.
- `trustedCACertificate`: Certificado de CA confiável. Se estiver usando uma CA confiável, esse parâmetro deve ser fornecido. Isso pode ser ignorado se nenhuma CA confiável for usada.

Você pode usar um dos seguintes tipos de certificados:

- Certificado auto-assinado
- Certificado de terceiros

Ative a autenticação com um certificado autoassinado

Um fluxo de trabalho típico envolve as etapas a seguir.

Passos

1. Gerar um certificado e chave de cliente. Ao gerar, defina o Nome Comum (CN) para o usuário ONTAP para autenticar como.

```
openssl req -x509 -nodes -days 1095 -newkey rsa:2048 -keyout k8senv.key  
-out k8senv.pem -subj "/C=US/ST=NC/L=RTP/O=NetApp/CN=<common-name>"
```

2. Instale o certificado de cliente de tipo `client-ca` e chave no cluster do ONTAP.

```
security certificate install -type client-ca -cert-name <certificate-  
name> -vserver <vserver-name>  
security ssl modify -vserver <vserver-name> -client-enabled true
```

3. Confirme se a função de login de segurança do ONTAP suporta o método de autenticação de certificado.

```
security login create -user-or-group-name vsadmin -application ontapi  
-authentication-method cert -vserver <vserver-name>  
security login create -user-or-group-name vsadmin -application http  
-authentication-method cert -vserver <vserver-name>
```

4. Teste a autenticação usando o certificado gerado. Substitua o ONTAP Management LIF> e o <vserver name> pelo IP de LIF de gerenciamento e nome da SVM. Você deve garantir que o LIF tenha sua política de serviço definida como `default-data-management`.

```
curl -X POST -Lk https://<ONTAP-Management-  
LIF>/servlets/netapp.servlets.admin.XMLrequest_filer --key k8senv.key  
--cert ~/k8senv.pem -d '<?xml version="1.0" encoding="UTF-8"?><netapp  
xmlns=http://www.netapp.com/filer/admin version="1.21" vfiler="<vserver-  
name>"><vserver-get></vserver-get></netapp>
```

5. Usando os valores obtidos na etapa anterior, adicione o back-end de storage na IU do Astra Control Center.

Ative a autenticação com um certificado de terceiros

Se você tiver um certificado de terceiros, poderá configurar a autenticação baseada em certificado com estas etapas.

Passos

1. Gerar a chave privada e CSR:

```
openssl req -new -newkey rsa:4096 -nodes -sha256 -subj "/" -outform pem  
-out ontap_cert_request.csr -keyout ontap_cert_request.key -addext  
"subjectAltName = DNS:<ONTAP_CLUSTER_FQDN_NAME>,IP:<ONTAP_MGMT_IP>"
```

2. Passe o CSR para a CA do Windows (CA de terceiros) e emita o certificado assinado.

3. Baixe o certificado assinado e nomeie-o como "ONTAP_signed_cert.crt"
4. Exporte o certificado raiz da CA do Windows (CA de terceiros).
5. Nomeie este arquivo `ca_root.crt`

Agora você tem os seguintes três arquivos:

- **Chave privada:** `ontap_signed_request.key` (Esta é a chave correspondente para o certificado do servidor no ONTAP. É necessário ao instalar o certificado do servidor.)
 - **Certificado assinado:** `ontap_signed_cert.crt` (Isso também é chamado de *certificado do servidor* no ONTAP.)
 - **Certificado CA raiz:** (Também é chamado de *certificado CA* `ca_root.crt` *Server-CA* no ONTAP.)
6. Instale estes certificados no ONTAP. Gerar, instalar `server` e `server-ca` certificados no ONTAP.

Expanda para Sample.yaml

```
# Copy the contents of ca_root.crt and use it here.
```

```
security certificate install -type server-ca
```

```
Please enter Certificate: Press <Enter> when done
```

```
-----BEGIN CERTIFICATE-----
```

```
<certificate details>
```

```
-----END CERTIFICATE-----
```

You should keep a copy of the CA-signed digital certificate for future reference.

The installed certificate's CA and serial number for reference:

CA:

serial:

The certificate's generated name for reference:

===

```
# Copy the contents of ontap_signed_cert.crt and use it here. For  
key, use the contents of ontap_cert_request.key file.
```

```
security certificate install -type server
```

```
Please enter Certificate: Press <Enter> when done
```

```
-----BEGIN CERTIFICATE-----
```

```
<certificate details>
```

```
-----END CERTIFICATE-----
```

```
Please enter Private Key: Press <Enter> when done
```

```
-----BEGIN PRIVATE KEY-----
```

```
<private key details>
```

```
-----END PRIVATE KEY-----
```

Enter certificates of certification authorities (CA) which form the certificate chain of the server certificate. This starts with the issuing CA certificate of the server certificate and can range up to the root CA certificate.

Do you want to continue entering root and/or intermediate

```
certificates {y|n}: n
```

The provided certificate does not have a common name in the subject field.

Enter a valid common name to continue installation of the certificate: <ONTAP_CLUSTER_FQDN_NAME>

You should keep a copy of the private key and the CA-signed digital certificate for future reference.

The installed certificate's CA and serial number for reference:

CA:

serial:

The certificate's generated name for reference:

```
==
```

```
# Modify the vsrver settings to enable SSL for the installed certificate
```

```
ssl modify -vsrver <vsrver_name> -ca <CA> -server-enabled true  
-serial <serial number> (security ssl modify)
```

```
==
```

```
# Verify if the certificate works fine:
```

```
openssl s_client -CAfile ca_root.crt -showcerts -servername server  
-connect <ONTAP_CLUSTER_FQDN_NAME>:443
```

```
CONNECTED(00000005)
```

```
depth=1 DC = local, DC = umca, CN = <CA>
```

```
verify return:1
```

```
depth=0
```

```
verify return:1
```

```
write W BLOCK
```

```
---
```

```
Certificate chain
```

```
0 s:
```

```
    i:/DC=local/DC=umca/<CA>
```

```
-----BEGIN CERTIFICATE-----
```

```
<Certificate details>
```

7. Crie o certificado de cliente para o mesmo host para comunicação sem senha. O Centro de Controle Astra usa esse processo para se comunicar com o ONTAP.
8. Gerar e instalar os certificados de cliente no ONTAP:

Expanda para Sample.yaml

```
# Use /CN=admin or use some other account which has privileges.
openssl req -x509 -nodes -days 1095 -newkey rsa:2048 -keyout
ontap_test_client.key -out ontap_test_client.pem -subj "/CN=admin"

Copy the content of ontap_test_client.pem file and use it in the
below command:
security certificate install -type client-ca -vserver <vserver_name>

Please enter Certificate: Press <Enter> when done

-----BEGIN CERTIFICATE-----
<Certificate details>
-----END CERTIFICATE-----

You should keep a copy of the CA-signed digital certificate for
future reference.
The installed certificate's CA and serial number for reference:

CA:
serial:
The certificate's generated name for reference:

==

ssl modify -vserver <vserver_name> -client-enabled true
(security ssl modify)

# Setting permissions for certificates
security login create -user-or-group-name admin -application ontapi
-authentication-method cert -role admin -vserver <vserver_name>

security login create -user-or-group-name admin -application http
-authentication-method cert -role admin -vserver <vserver_name>

==

#Verify passwordless communication works fine with the use of only
certificates:

curl --cacert ontap_signed_cert.crt --key ontap_test_client.key
--cert ontap_test_client.pem
https://<ONTAP_CLUSTER_FQDN_NAME>/api/storage/aggregates
{
```

```

"records": [
{
"uuid": "f84e0a9b-e72f-4431-88c4-4bf5378b41bd",
"name": "<aggr_name>",
"node": {
"uuid": "7835876c-3484-11ed-97bb-d039ea50375c",
"name": "<node_name>",
"_links": {
"self": {
"href": "/api/cluster/nodes/7835876c-3484-11ed-97bb-d039ea50375c"
}
}
},
"_links": {
"self": {
"href": "/api/storage/aggregates/f84e0a9b-e72f-4431-88c4-4bf5378b41bd"
}
}
},
],
"num_records": 1,
"_links": {
"self": {
"href": "/api/storage/aggregates"
}
}
}%

```

9. Adicione o back-end de storage à IU do Astra Control Center e forneça os seguintes valores:

- **Certificado do cliente:** ONTAP_test_client.pem
- **Chave privada:** ONTAP_test_client.key
- **Certificado de CA confiável:** ONTAP_signed_cert.crt

Adicionar um back-end de storage

Você pode adicionar um back-end de storage do ONTAP existente ao Astra Control Center para gerenciar seus recursos.

O gerenciamento de clusters de storage no Astra Control como um back-end de storage permite que você tenha vínculos entre volumes persistentes (PVS) e o back-end de storage, bem como métricas de storage adicionais.

Depois de configurar as credenciais ou as informações de autenticação de certificado, você poderá adicionar um back-end de storage do ONTAP existente ao Astra Control Center para gerenciar seus recursos.

Passos

1. No Painel na área de navegação à esquerda, selecione **backends**.
2. Selecione **Adicionar**.
3. Na seção usar existente da página Adicionar storage backend, selecione **ONTAP**.
4. Selecione uma das seguintes opções:
 - **Use as credenciais de administrador:** Insira o endereço IP e as credenciais de administrador de gerenciamento de cluster do ONTAP. As credenciais devem ser credenciais de todo o cluster.



O usuário cujas credenciais você inserir aqui deve ter o `ontapi` método de acesso de login de usuário habilitado no Gerenciador de sistema do ONTAP no cluster do ONTAP. Se você planeja usar a replicação do SnapMirror, aplique credenciais de usuário com a função "admin", que tem os métodos de acesso `ontapi` e `http`, nos clusters ONTAP de origem e destino. ["Gerenciar contas de usuário na documentação do ONTAP"](#) Consulte para obter mais informações.

- **Use um certificado:** Carregue o arquivo de certificado `.pem`, o arquivo de chave de certificado `.key` e, opcionalmente, o arquivo de autoridade de certificação.
5. Selecione **seguinte**.
 6. Confirme os detalhes do backend e selecione **Manage**.

Resultado

O backend aparece no `online` estado da lista com informações de resumo.



Talvez seja necessário atualizar a página para que o backend apareça.

Adicione um balde

Você pode adicionar um bucket usando a IU do Astra Control ou ["API Astra Control"](#)o . Adicionar fornecedores de bucket do armazenamento de objetos é essencial para fazer backup das aplicações e do storage persistente ou clonar aplicações entre clusters. O Astra Control armazena os backups ou clones nos buckets do armazenamento de objetos que você define.

Você não precisa de um bucket no Astra Control se estiver clonando a configuração da aplicação e o storage persistente para o mesmo cluster. A funcionalidade de instantâneos de aplicações não requer um intervalo.

Antes de começar

- Um bucket que pode ser acessado a partir dos clusters gerenciados pelo Astra Control Center.
- Credenciais para o bucket.
- Um balde dos seguintes tipos:
 - NetApp ONTAP S3
 - NetApp StorageGRID S3
 - Microsoft Azure
 - Genérico S3



A Amazon Web Services (AWS) e o Google Cloud Platform (GCP) usam o tipo de bucket Generic S3.



Embora o Astra Control Center ofereça suporte ao Amazon S3 como um provedor de bucket do Generic S3, o Astra Control Center pode não oferecer suporte a todos os fornecedores de armazenamento de objetos que claim o suporte ao S3 da Amazon.

Passos

1. Na área de navegação à esquerda, selecione **Buckets**.
2. Selecione **Adicionar**.
3. Selecione o tipo de balde.



Quando você adiciona um bucket, selecione o provedor de bucket correto e forneça as credenciais certas para esse provedor. Por exemplo, a IU aceita o NetApp ONTAP S3 como o tipo e aceita credenciais StorageGRID; no entanto, isso fará com que todos os backups e restaurações futuros de aplicativos que usam esse bucket falhem.

4. Insira um nome de bucket existente e uma descrição opcional.



O nome e a descrição do bucket aparecem como um local de backup que você pode escolher mais tarde ao criar um backup. O nome também aparece durante a configuração da política de proteção.

5. Introduza o nome ou endereço IP do endpoint S3.
6. Em **Selecionar credenciais**, escolha a guia **Adicionar** ou **usar existente**.
 - Se você escolheu **Add**:
 - i. Insira um nome para a credencial que a distingue de outras credenciais no Astra Control.
 - ii. Insira a ID de acesso e a chave secreta colando o conteúdo da área de transferência.
 - Se você escolheu **Use existing**:
 - i. Selecione as credenciais existentes que você deseja usar com o bucket.
7. `Add` Selecione .



Quando você adiciona um balde, o Astra Control marca um balde com o indicador de balde padrão. O primeiro bucket que você criar se torna o bucket padrão. À medida que você adiciona buckets, você pode decidir mais tarde "[defina outro intervalo padrão](#)".

O que se segue?

Agora que você fez login e adicionou clusters ao Astra Control Center, está pronto para começar a usar os recursos de gerenciamento de dados de aplicações do Astra Control Center.

- "[Gerencie usuários e funções locais](#)"
- "[Comece a gerenciar aplicativos](#)"
- "[Proteja aplicativos](#)"
- "[Gerenciar notificações](#)"
- "[Conecte-se ao Cloud Insights](#)"
- "[Adicione um certificado TLS personalizado](#)"

- ["Altere a classe de armazenamento padrão"](#)

Encontre mais informações

- ["Use a API Astra Control"](#)
- ["Problemas conhecidos"](#)

Perguntas mais frequentes para o Astra Control Center

Este FAQ pode ajudar se você está apenas procurando uma resposta rápida para uma pergunta.

Visão geral

As seções a seguir fornecem respostas a algumas perguntas adicionais que você pode encontrar ao usar o Astra Control Center. Para esclarecimentos adicionais, entre em Contato com o NetApp.com

Acesso ao Astra Control Center

Qual é a URL do Astra Control?

O Astra Control Center usa autenticação local e uma URL específica para cada ambiente.

Para o URL, em um navegador, digite o nome de domínio totalmente qualificado (FQDN) definido no campo spec.astraAddress no arquivo Astra_control_center.yaml custom resource (CR) quando você instalou o Astra Control Center. O e-mail é o valor definido no campo spec.email no astra_control_center.yaml CR.

Licenciamento

Estou usando uma licença de avaliação. Como faço para mudar para a licença completa?

Você pode facilmente mudar para uma licença completa obtendo o arquivo de licença NetApp (NLF) da NetApp.

Passos

1. Na navegação à esquerda, selecione **conta > Licença**.
2. Na visão geral da licença, à direita das informações da licença, selecione o menu Opções.
3. Selecione **Substituir**.
4. Navegue até o arquivo de licença que você baixou e selecione **Adicionar**.

Estou usando uma licença de avaliação. Ainda posso gerenciar aplicativos?

Sim, você pode testar a funcionalidade de gerenciamento de aplicativos com uma licença de avaliação (incluindo a licença de avaliação incorporada instalada por padrão). Não há diferença em recursos ou recursos entre uma licença de avaliação e uma licença completa; a licença de avaliação simplesmente tem uma vida útil mais curta. ["Licenciamento"](#) Consulte para obter mais informações.

Registrando clusters do Kubernetes

Eu preciso adicionar nós de trabalho ao meu cluster do Kubernetes depois de adicionar ao Astra Control. O que devo fazer?

Novos nós de trabalho podem ser adicionados a pools existentes. Eles serão descobertos automaticamente pelo Astra Control. Se os novos nós não estiverem visíveis no Astra Control, verifique se os novos nós de trabalho estão executando o tipo de imagem suportado. Você também pode verificar a integridade dos novos nós de trabalho usando o `kubectl get nodes` comando.

Como faço para desgerenciar corretamente um cluster?

1. ["Desgerenciar as aplicações do Astra Control"](#).
2. ["Desgerenciar o cluster a partir do Astra Control"](#).

O que acontece com minhas aplicações e dados após a remoção do cluster Kubernetes do Astra Control?

A remoção de um cluster do Astra Control não fará alterações na configuração do cluster (aplicações e storage persistente). Todos os snapshots ou backups do Astra Control feitos de aplicações nesse cluster não estarão disponíveis para restauração. Os backups de storage persistente criados pelo Astra Control permanecem no Astra Control, mas não estão disponíveis para restauração.



Sempre remova um cluster do Astra Control antes de excluí-lo por meio de outros métodos. A exclusão de um cluster usando outra ferramenta enquanto ele ainda está sendo gerenciado pelo Astra Control pode causar problemas para sua conta Astra Control.

O NetApp Astra Trident é desinstalado automaticamente de um cluster quando eu o desgerencio?

Quando você desgerencia um cluster do Astra Control Center, o Astra Trident não é desinstalado automaticamente do cluster. Para desinstalar o Astra Trident, você precisará ["Siga estas etapas na documentação do Astra Trident"](#).

Gerenciamento de aplicações

O Astra Control pode implantar uma aplicação?

O Astra Control não implanta aplicações. As aplicações precisam ser implantadas fora do Astra Control.

O que acontece com as aplicações depois de parar de gerenciá-las do Astra Control?

Quaisquer backups ou snapshots existentes serão excluídos. Aplicativos e dados permanecem disponíveis. As operações de gerenciamento de dados não estarão disponíveis para aplicativos não gerenciados ou backups ou snapshots que pertençam a eles.

O Astra Control pode gerenciar uma aplicação que esteja em um storage que não seja da NetApp?

Não. Embora o Astra Control possa descobrir aplicações que estão usando storage que não é NetApp, ele não pode gerenciar uma aplicação que esteja usando storage que não seja NetApp.

Devo gerenciar o próprio Astra Control?

O Astra Control Center não é mostrado por padrão como uma aplicação que pode ser gerenciada, mas é possível fazer backup e restaurar uma instância do Astra Control Center usando outra instância do Astra Control Center.

Os pods não saudáveis afetam o gerenciamento de aplicativos?

Não, a integridade dos pods não afeta o gerenciamento de aplicativos.

Operações de gerenciamento de dados

Meu aplicativo usa vários PVS. O Astra Control fará snapshots e backups desses PVS?

Sim. Uma operação de snapshot em uma aplicação do Astra Control inclui snapshots de todos os PVS vinculados aos PVCs da aplicação.

Posso gerenciar snapshots feitos pelo Astra Control diretamente por meio de uma interface ou storage de objetos diferente?

Não. Os snapshots e backups feitos pelo Astra Control só podem ser gerenciados com o Astra Control.

Informações sobre direitos autorais

Copyright © 2025 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 SOBRESSALENTE; 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 SOFTWARES, 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.