



Lustre com o NetApp storage E-Series

NetApp artificial intelligence solutions

NetApp
August 31, 2026

Índice

Lustre com o NetApp storage E-Series	1
Lustre com o NetApp E-Series Storage - Introdução	1
Visão geral da solução	1
Lustre com NetApp storage E-Series - Arquitetura da solução	2
Projeto de building block	2
Recomendações de dimensionamento	3
Arquitetura HA	4
Arquitetura de rede	5
Clientes Lustre	6
Lustre com o NetApp E-Series Storage - Componentes de hardware	6
Requisitos do servidor	6
Bloco de construção padrão EF80	8
Seleção do pool de armazenamento: TLC vs QLC	12
Requisitos de rede	12
Lustre com o NetApp E-Series storage - Componentes de software	13
Software de blocos de construção	13
Automação Ansible	14
Software cliente Lustre	14
Lustre com o NetApp E-Series Storage - Guia de dimensionamento	15
Dimensionamento de capacidade	15
Dimensionamento	16
Desempenho	17
Implante a solução	17
Implemente o Lustre com o NetApp E-Series Storage	17
Hardware de rack e cabeamento Lustre	18
Prepare o ambiente de implantação do Lustre	19
Personalize o inventário Ansible do Lustre	21
Implante o cluster Lustre HA e os clientes	24
Verifique e expanda a implementação do Lustre	27
Lustre com o NetApp E-Series Storage - Recursos relacionados	28
NetApp recursos	29
Comunidade Lustre	29
Soluções de infraestrutura de IA relacionadas da NetApp	29

Lustre com o NetApp storage E-Series

Lustre com o NetApp E-Series Storage - Introdução

O Lustre com NetApp E-Series Storage é uma solução validada de sistema de arquivos paralelo para cargas de trabalho de IA e HPC, construída a partir de blocos repetíveis de pares de servidores HA e arrays all-flash E-Series.

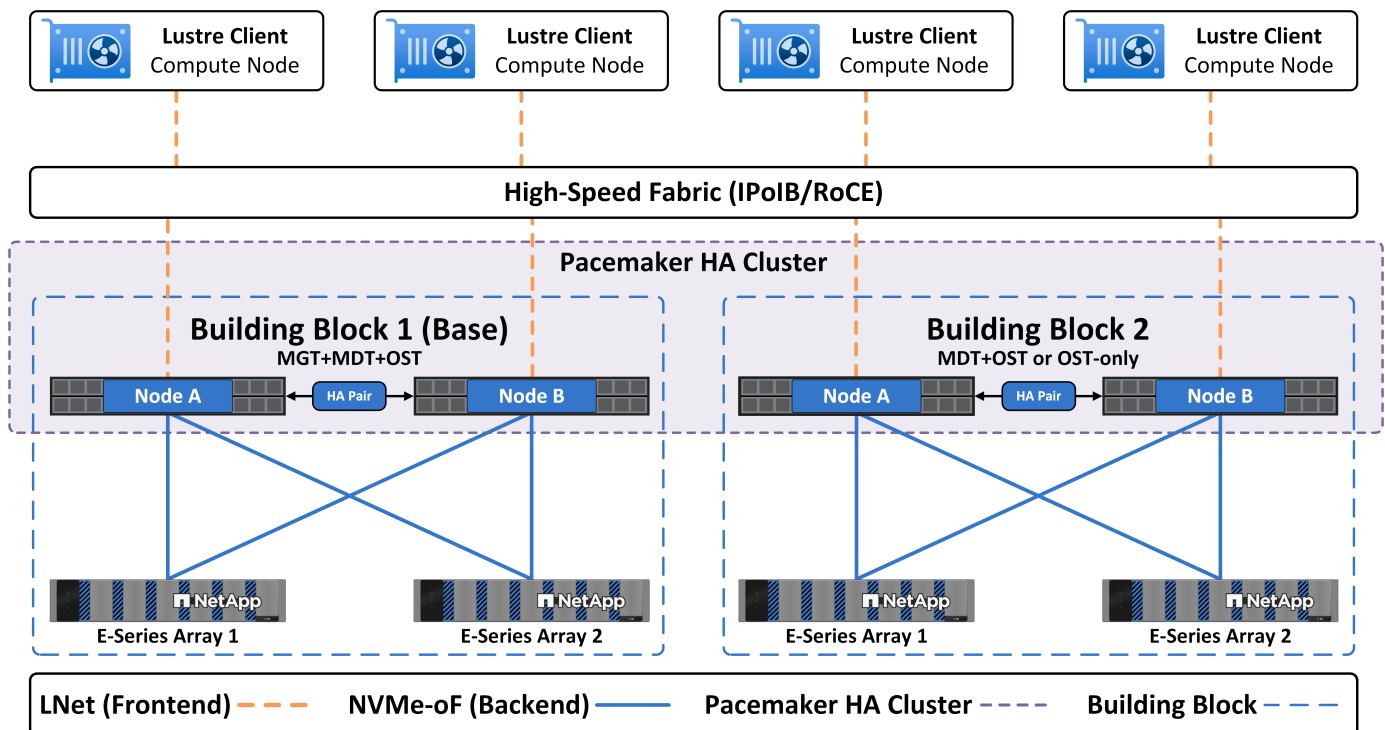
Visão geral da solução

O Lustre com NetApp E-Series Storage combina o sistema de arquivos paralelo open-source Lustre com o NetApp E-Series block storage para fornecer uma base escalável e de alto desempenho para cargas de trabalho com uso intensivo de dados. Cada bloco de construção emparelha dois nós de servidor OSS/MDS configurados para alta disponibilidade (HA) com dois arrays all-flash E-Series. A conectividade NVMe-oF de backend transporta E/S de bloco para os arrays e uma malha LNet separada conecta clientes e nós de servidor OSS/MDS para acesso paralelo ao sistema de arquivos.

Um bloco de construção básico hospeda o servidor de gerenciamento (MGS) e os destinos de metadados (MDTs) e storage de objetos (OSTs) iniciais. Blocos de construção adicionais se registram no MGS básico para dimensionar a capacidade de metadados, a capacidade de dados e a taxa de transferência agregada conforme as demandas de carga de trabalho aumentam.

A figura a seguir mostra um exemplo de implantação com vários building blocks e clientes Lustre conectados por meio da LNet.

Visão geral da solução de armazenamento Lustre com NetApp E-Series



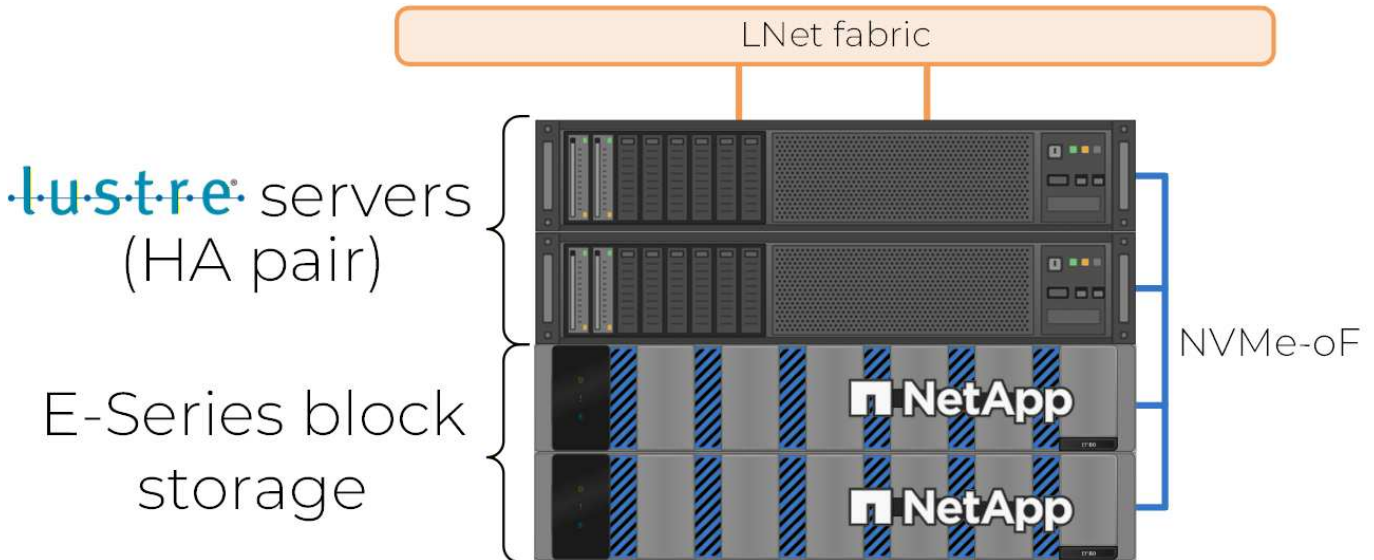
Lustre com NetApp storage E-Series - Arquitetura da solução

Aprenda como o Lustre com NetApp E-Series storage usa blocos de construção, alta disponibilidade, topologia de rede e clientes para que você possa planejar capacidade, desempenho e failover.

Projeto de building block

Um bloco de construção é a unidade fundamental e escalável da solução Lustre NetApp E-Series. Cada bloco de construção consiste em dois nós de servidor configurados como um par de alta disponibilidade (HA) que fornecem as funções de servidor de armazenamento de objetos (OSS) e servidor de metadados (MDS) do Lustre, dois arrays all-flash NetApp E-Series, conectividade dedicada de backend NVMe over Fabrics (NVMe-oF) entre servidores e arrays, e conectividade dedicada de rede Lustre (LNet) para clientes e comunicação entre nós. Um cluster HA Pacemaker e Corosync pode conter o par de servidores de um ou mais blocos de construção. A coleção NetApp `ansible-lustre` utiliza automação Ansible para implantar e configurar os serviços Lustre.

Bloco de construção Lustre



Os blocos de construção são dimensionados para atender aos requisitos de crescimento do sistema de arquivos. Cada sistema de arquivos requer um bloco de construção básico. O bloco de construção básico hospeda o servidor de gerenciamento (MGS) em um destino de gerenciamento (MGT) e fornece capacidade inicial de destino de metadados (MDT) e de destino de storage de objetos (OST). Blocos de construção adicionais estendem o sistema de arquivos e registram seus destinos MDT e OST no MGS do bloco de construção básico. A abordagem modular permite que a capacidade e o desempenho sejam dimensionados independentemente com base nas necessidades da carga de trabalho.

NetApp recomenda os seguintes perfis de configuração de blocos de construção para a maioria das implementações. As contagens de MGS, MDT e OST listadas são valores padrão recomendados, otimizados para maximizar o desempenho e a capacidade dos storage E-Series. Ajuste as contagens de destino, os tamanhos de volume e a alocação de drives E-Series no inventário do Ansible para atender às necessidades da carga de trabalho. Por exemplo, implementações que exigem maior capacidade de storage de metadados podem provisionar mais MDTs e menos OSTs no mesmo hardware de bloco de construção.

A tabela a seguir resume os tipos de blocos de construção e as quantidades-alvo recomendadas.

Tipo	MGS	MDTs	OSTs	Usar
Base	1	8	32	Primeiro componente fundamental de um sistema de arquivos. Hospeda o MGS e a capacidade inicial de MDT e OST.
MDT+OST	0	8	32	Adiciona metadados e capacidade de dados. Registre-se no bloco de construção base MGS.
Somente OST	0	0	32	Adiciona apenas capacidade de dados. Registre-se no bloco de construção base MGS.

- **Tipo de unidade e layout do pool:** O E-Series suporta grupos de volumes RAID tradicionais e Dynamic Disk Pools (DDP). Para unidades NVMe TLC, use grupos de volumes RAID 6 para armazenamento OST e grupos de volumes RAID 1 para armazenamento MGS e MDT. Para unidades NVMe QLC, use DDP para o pool de unidades compartilhadas.
- **Distribuição de alvos:** Os alvos Lustre em cada bloco de construção são balanceados entre os dois nós de servidor OSS/MDS e os dois arrays E-Series. O layout distribui a E/S entre as CPUs do servidor e os controladores de array para melhorar o desempenho do caminho NVMe-oF e manter a redundância. Consulte ["Distribuição de target e conectividade NVMe-oF"](#) em ["Componentes de hardware"](#).

Os sistemas operacionais suportados, as versões do Lustre, o firmware da array e os componentes básicos relacionados estão listados em ["NetApp Ferramenta de Matriz de Interoperabilidade \(IMT\)"](#) em **E-Series Lustre**. Para obter informações detalhadas sobre a contagem de volumes, layouts de unidades e diretrizes de dimensionamento, consulte ["Componentes de hardware"](#). Para os componentes de software, consulte ["Componentes de software"](#).

Recomendações de dimensionamento

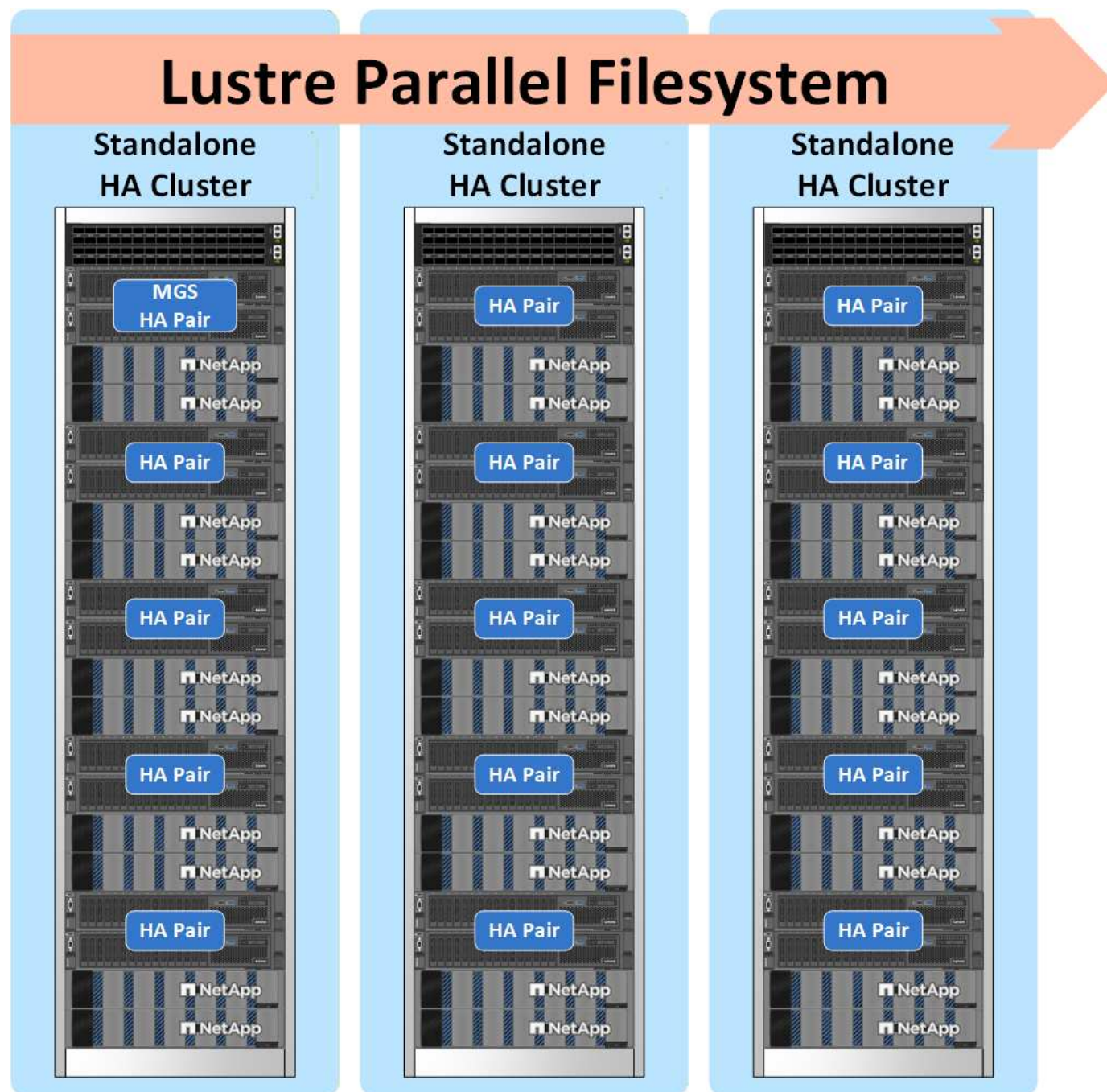
O sistema de arquivos Lustre escala ao adicionar blocos de construção quando a capacidade de metadados, a capacidade de dados ou ambas se tornam limitantes. Escale os MDTs com base na contagem de arquivos e em IOPS de metadados; escale os OSTs com base na capacidade e nos requisitos agregados de taxa de transferência.

1. **Um bloco de construção base por sistema de arquivos:** Implante exatamente um bloco de construção base; ele hospeda o MGS e os destinos iniciais do MDT e do OST.
2. **Perfil de bloco de construção adicional:** Adicione um bloco de construção somente OST quando a capacidade MDT existente for suficiente e a capacidade de dados ou a taxa de transferência precisar ser aumentada. Adicione um bloco de construção MDT+OST quando o desempenho dos metadados ou a capacidade do namespace se tornar o gargalo.
3. **Limite de cluster HA:** limite cada cluster HA do Pacemaker e do Corosync a cinco blocos de construção (dez nós de servidor Lustre). Divida implantações maiores uniformemente em vários clusters HA para

evitar restrições de recursos em configurações grandes.

A figura a seguir mostra até cinco blocos de construção empilhados em um rack (um cluster Pacemaker HA por rack). Vários racks podem participar de um único sistema de arquivos Lustre; somente o bloco de construção base hospeda o MGS.

Dimensionamento de rack do sistema de arquivos paralelo Lustre



Para exemplos de dimensionamento, veja ["Orientação de dimensionamento"](#).

Arquitetura HA

O Lustre com NetApp E-Series storage utiliza uma arquitetura de alta disponibilidade (HA) de disco compartilhado integrada ao NetApp E-Series storage. O Pacemaker gerencia os recursos de HA e o Corosync fornece associação ao cluster e mensagens. Juntos, eles gerenciam o failover do alvo Lustre entre os dois

nós de servidor OSS/MDS em cada bloco de construção. O multipath NVMe-oF conecta ambos os nós de servidor OSS/MDS aos mesmos volumes E-Series, para que qualquer nó possa assumir os serviços de destino quando necessário.

Cada MGS, MDT e OST é configurado como um recurso de alta disponibilidade (HA) com suas dependências em um grupo de recursos Pacemaker. O Pacemaker garante que os recursos iniciem e parem na ordem correta, permaneçam colocados no mesmo nó e sejam executados em apenas um nó por vez. O acesso simultâneo ao mesmo destino a partir de ambos os nós pode corromper o sistema de arquivos subjacente.

- **Failover de destino:** Ambos os nós do servidor OSS/MDS são pares no cluster, mas cada destino Lustre está ativo em apenas um nó por vez. Um recurso de monitoramento Pacemaker acompanha a integridade de cada destino e suas dependências e aciona um failover quando um destino fica indisponível em seu nó atual. Em caso de falha, o Pacemaker reinicia o grupo de recursos afetado no nó parceiro na ordem correta, e os clientes se reconectam de forma transparente ao destino em sua nova localização assim que os serviços são retomados. Como cada destino é ativado em um único nó, o sistema de arquivos nunca é exposto a gravações simultâneas de ambos os nós.
- **Acesso ao storage compartilhado:** Os caminhos NVMe-oF multipath conectam cada nó do servidor OSS/MDS a todos os controladores E-Series no bloco de construção, assim, cada volume de destino é acessível a partir de qualquer um dos nós. Se um nó do servidor falhar, o nó parceiro já possui caminhos ativos para os mesmos volumes e pode assumir a propriedade dos destinos sem reconfigurar a conectividade de storage. Se um controlador de array ou um caminho falhar, o multipath redireciona a E/S para um controlador em funcionamento. Essa redundância dupla permite que os destinos Lustre façam failover entre nós de servidor OSS/MDS ou controladores de array sem perder o acesso aos volumes de suporte.
- **Isolamento STONITH:** Quando ocorre uma falha, o Pacemaker às vezes não consegue se comunicar com o nó defeituoso para confirmar se seus alvos foram interrompidos. Antes de reiniciar esses alvos em outro local, o Pacemaker isola o nó defeituoso, idealmente cortando a energia, para garantir que ele esteja desligado. O isolamento impede um cenário de "split-brain" no qual ambos os nós acessam o mesmo alvo simultaneamente e corrompem o sistema de arquivos subjacente, preservando a integridade de dados durante o failover. NetApp recomenda `fence_redfish` para servidores com controladores de gerenciamento de placa-mãe (BMCs) compatíveis com Redfish. Outros agentes de isolamento, como `fence_apc`, também são suportados.

Para administração de cluster e configuração de fencing, consulte o ["Guia do usuário do HA"](#) no ["coleção ansible-lustre"](#).

Arquitetura de rede

A solução utiliza caminhos de rede separados para o tráfego de backend de storage e para o tráfego de frontend da LNet.

- **Backend (NVMe-oF):** Os nós do servidor OSS/MDS se conectam aos arrays E-Series por meio de NVMe-oF usando NVMe/InfiniBand ou NVMe/RoCE. O caminho NVMe-oF transporta E/S de bloco entre os serviços de destino Lustre e os volumes E-Series. Para cabeamento de backend EF80 six-HCA, consulte ["Rack e cabeamento do hardware Lustre"](#). Para posicionamento preferencial de destino entre nós e arrays, consulte ["distribuição alvo"](#) em ["Componentes de hardware"](#).
- **Frontend (LNet):** Os clientes Lustre e os nós do servidor OSS/MDS comunicam-se pela LNet usando o tipo de rede `@o2ib` com a pilha NVIDIA OFED. InfiniBand e RoCE são ambos transportes de frontend validados. A coleção `ansible-lustre` configura todas as interfaces de frontend para LNet multi-rail, que agrega múltiplas portas para aumentar a largura de banda e fornecer redundância de caminho para o tráfego de clientes e entre nós.
- **Gerenciamento e cluster:** Uma rede de gerenciamento fora da banda fornece gerenciamento de

servidores, acesso ao BMC e gerenciamento de arrays. Agentes de fencing STONITH, como `fence_redfish`, usam essa rede para alcançar os BMCs dos servidores e cortar a energia de um nó com falha. O Corosync também pode executar a comunicação do cluster por meio dessa rede como um anel adicional junto à malha de frontend. Configurar o Corosync com vários anéis adiciona redundância para mensagens do cluster e reduz o risco de que uma única falha de rede interrompa o quorum.

Clientes Lustre

Um cliente Lustre carrega o módulo do kernel do cliente, estabelece conectividade LNet com os nós do servidor OSS/MDS e monta o sistema de arquivos para apresentar um namespace único, coerente e compatível com POSIX para os aplicativos. A E/S é distribuída em paralelo entre os OSTs ativos. Os nós do cliente são externos a um bloco de construção e usam o sistema de arquivos por meio do fabric LNet de frontend. Os sistemas operacionais do cliente não estão listados no IMT para esta solução; use a ["Matriz de suporte Lustre"](#) para distribuições de cliente testadas e versões do kernel compatíveis com a versão Lustre implantada (por exemplo, Red Hat Enterprise Linux 9, SUSE Linux Enterprise Server 15 e Ubuntu 24.04).

Os nós clientes precisam de conectividade com a mesma fabric LNet e tipo de rede que os nós do servidor OSS/MDS. Se necessário, um roteador LNet pode interligar sub-redes ou fabrics LNet separadas.

NetApp fornece pacotes RPM de servidor Lustre pré-compilados para Rocky Linux 9.8 e Red Hat Enterprise Linux 9.8. NetApp não fornece pacotes de cliente pré-compilados. Compile os pacotes de cliente para o sistema operacional e kernel do cliente a partir do código-fonte do Lustre no ["repositório netapp-lustre"](#).

Após a instalação dos pacotes do cliente, a função opcional `lustre_client` no ["coleção ansible-lustre"](#) configura as interfaces de rede do cliente e o LNet, habilita o LNet multi-rail quando selecionado, valida a conectividade e gerencia uma unidade de montagem `systemd` persistente. A função não compila o Lustre. Ela instala os pacotes do cliente somente quando `lustre_client_packages` está preenchido. Se desejado, você pode configurar os clientes manualmente. Para procedimentos passo a passo, consulte ["Implante a solução"](#) e o ["Documentação da função lustre_client"](#).

Lustre com o NetApp E-Series Storage - Componentes de hardware

Use estes requisitos de hardware para servidores, NetApp E-Series arrays, layout de storage e rede ao dimensionar e solicitar o Lustre com NetApp E-Series Storage. Para componentes de software, consulte ["Componentes de software"](#).

Requisitos do servidor

A solução utiliza um modelo "traga seu próprio servidor" (BYOS). Cada componente requer dois nós de servidor OSS/MDS que atendam ou excedam as seguintes especificações.

Especificações do nó

A tabela a seguir lista os requisitos mínimos de servidor por nó OSS/MDS.

Componente	Requisito
Quantidade	2 nós por bloco de construção
CPU	AMD EPYC ou Intel Xeon, 32 núcleos ou mais
Memória	256 GB DDR5 (mínimo)

Componente	Requisito
HCA's de rede	6 HCA's de 200 Gb com duas portas (veja Conectividade HCA)
Slots PCIe	2× PCIe Gen5 x16 e 4× PCIe Gen5 x8 (veja Requisitos do slot PCIe)
Unidades de inicialização	2× discos rígidos em RAID 1 (RAID por software ou hardware recomendado)

NetApp validou a solução usando servidores Lenovo ThinkSystem SR665 V3. Servidores equivalentes de outros fornecedores são suportados quando atendem a esses requisitos.

Conectividade HCA

A tabela a seguir lista os requisitos de HCA, porta front-end LNet e caminho NVMe-oF por nó de servidor OSS/MDS.

HCA's por nó	Portas LNet por nó Lustre	Caminhos NVMe-oF por nó Lustre	Exemplo de HCA
6 (12 portas)	4	8 no total (4 em cada array)	MCX755106AS-HEAT (Placa PCIe gen5 de 200 Gb com duas portas)

NetApp recomenda seis HCA's por nó para maximizar a largura de banda dos arrays de armazenamento EF80.

Requisitos do slot PCIe

Cada nó de servidor OSS/MDS em um bloco de construção EF80 comporta seis HCA's de porta dupla: dois para LNet e quatro para NVMe-oF. A largura do slot determina a largura de banda disponível para cada HCA, então confirme a largura elétrica de cada slot e riser, em vez de apenas a largura da placa. Uma placa Gen5 x16 instalada em um slot Gen5 x8 opera em x8.

- **LNet HCA's:** Instale em slots PCIe Gen5 x16 para atingir a taxa de transferência máxima do frontend.
- **NVMe-oF HCA's:** Instale em slots PCIe Gen5 x8 ou PCIe Gen5 x16.
- **Balanceamento NUMA:** Divida os HCA's igualmente entre as duas zonas NUMA, de forma que cada zona hospede duas interfaces LNet e quatro interfaces NVMe-oF.

A tabela a seguir lista o número mínimo de slots para cada nó de servidor OSS/MDS em um bloco de construção EF80.

Tipo de tráfego	HCA's por nó	Largura mínima da ranhura	Slots por zona NUMA
LNet	2	PCIe Gen5 x16	1
NVMe-oF	4	PCIe Gen5 x8	2

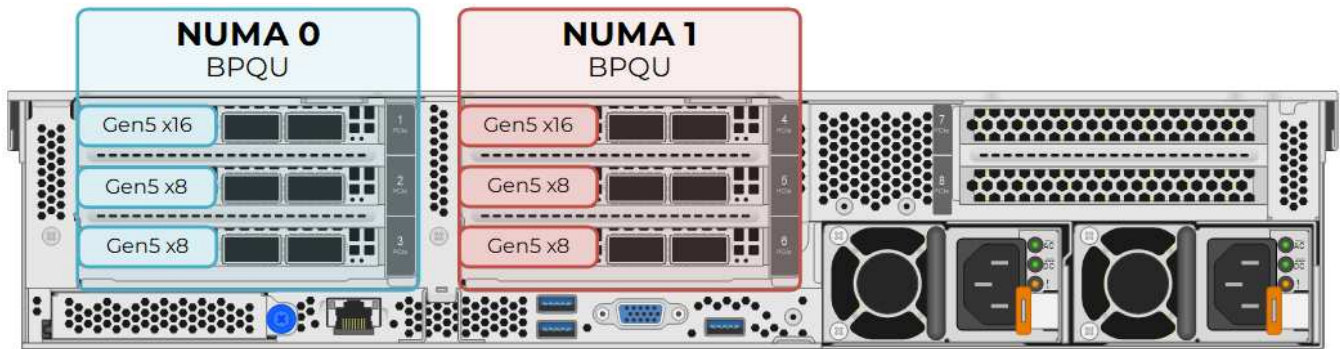
Opcionalmente, você pode dividir as portas em um HCA de porta dupla para que uma porta transporte o tráfego LNet e a outra o tráfego NVMe-oF. Instale todos os quatro HCA's em slots PCIe Gen5 x16 para que cada porta LNet atinja a taxa de transferência máxima, depois adicione mais dois HCA's em slots Gen5 x8 ou Gen5 x16 para as portas NVMe-oF restantes.

▼ **Exemplos de configurações de riser para Lenovo ThinkSystem SR665 V3**

Ambas as combinações de riser a seguir atendem aos requisitos de slot para um bloco de construção EF80.

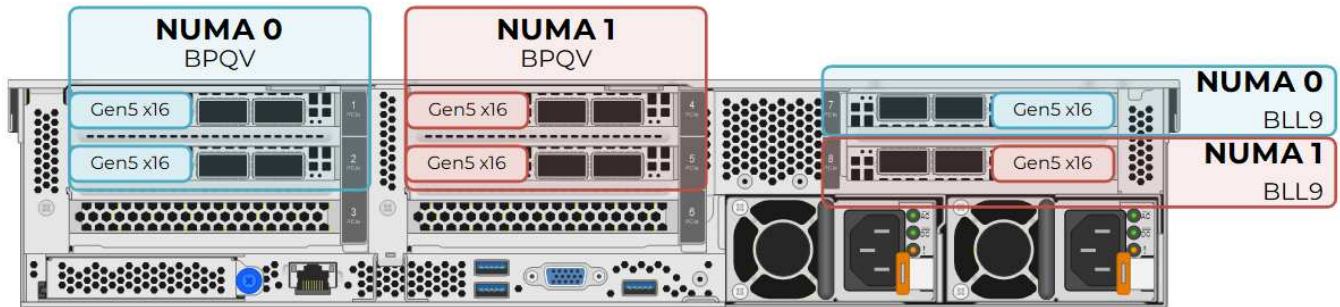
Larguras mínimas de slot

Instale um riser BPQU nas posições 1 e 2. Cada riser BPQU fornece um slot PCIe Gen5 x16 e dois slots PCIe Gen5 x8. Juntos, os risers fornecem dois slots Gen5 x16 para LNet e quatro slots Gen5 x8 para NVMe-oF, distribuídos igualmente entre as zonas NUMA.



Todos os slots Gen5 x16

Instale um riser BPQV nas posições 1 e 2 e um riser BLL9 na posição 3. Cada riser BPQV fornece dois slots PCIe Gen5 x16. O riser BLL9 fornece o slot 7 na zona NUMA 0 e o slot 8 na zona NUMA 1, ambos PCIe Gen5 x16. Essa combinação fornece seis slots Gen5 x16, três por zona NUMA, e suporta a divisão do tráfego LNet e NVMe-oF entre as portas do mesmo HCA.



Bloco de construção padrão EF80

O EF80 é a plataforma padrão validada para esta versão da solução.

Especificações do array

A tabela a seguir lista as especificações do array EF80 para um bloco de construção (dois arrays).

Componente	Especificação
Modelo	NetApp EF80
Fator de forma	chassi base 2U, 24 slots internos para SSD NVMe
Controladores	Controladores duplos (A e B)
Unidades	24 SSDs NVMe por array

Componente	Especificação
Conectividade de E/S	8 portas de host NVMe/IB ou NVMe/RoCE de 200 Gb por array no design Lustre validado

A plataforma EF80 suporta até doze portas de host por array quando três módulos de E/S de host de duas portas são instalados em cada controlador. O projeto Lustre validado utiliza módulos de E/S de host nos slots 1 e 2 para oito portas de host por array.

Cada array EF80 também utiliza o módulo de E/S dedicado do slot 4 para espelhamento entre controladoras. Conecte a porta 4a do controlador A à porta 4a do controlador B e a porta 4b do controlador A à porta 4b do controlador B. Essas conexões fornecem espelhamento de cache e transmissão de E/S e não são usadas para tráfego NVMe-oF do host. Consulte ["Conecte os cabos de espelhamento entre os controladores EF50 e EF80"](#).

Para obter as especificações completas da EF-Series em todos os modelos, consulte o ["NetApp EF-Series datasheet do array all-flash"](#).

Layout de drives (24 drives por array)

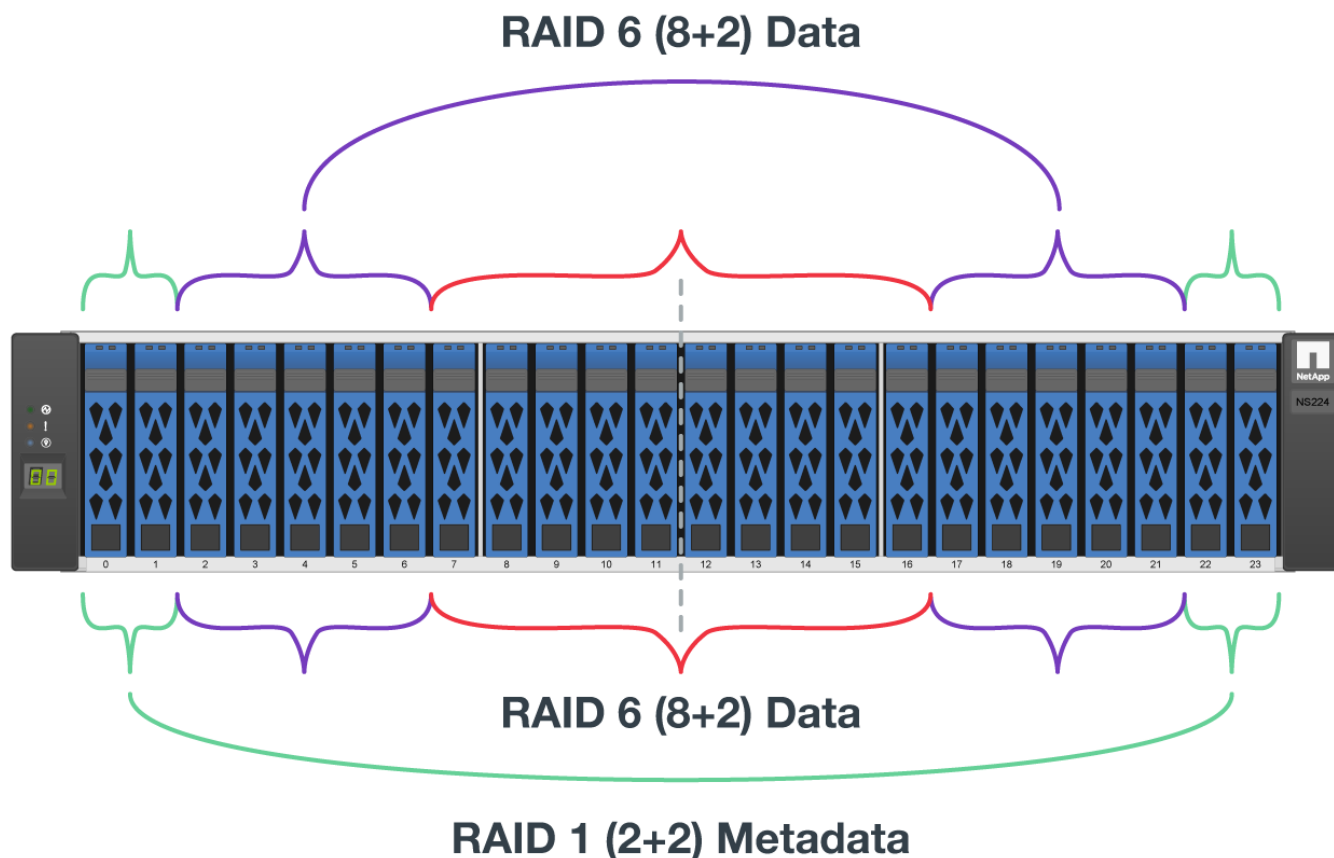
Cada array EF80 em um bloco de construção básico utiliza todos os vinte e quatro slots de unidade NVMe:

- 4 discos em RAID 1 para armazenamento MGS/MDT (grupo de volume compartilhado ou alocação DDP)
- 10 discos em RAID 6 para armazenamento de OST (primeiro pool de OST)
- 10 discos em RAID 6 para armazenamento de OST (segundo pool de OST)

Esse layout se aplica quando você usa discos de 3,84 TB, 7,68 TB ou 15,3 TB com grupos de volume tradicionais. Ao usar discos Capacity Flash (QLC) de 30,7 TB ou 61,4 TB, faça o provisionamento de um único Dynamic Disk Pool em todos os vinte e quatro discos e crie volumes RAID 1 dentro do pool para os volumes MGS e MDT, enquanto utiliza volumes RAID 6 padrão para os OSTs.



Discos rígidos de 1,92 TB não são recomendados atualmente para esta solução. Use uma das capacidades de disco validadas listadas acima.



Volume e contagens de alvos (bloco de construção básico)

A tabela a seguir lista as contagens de MGS, MDT e OST para um bloco de construção básico.

Tipo de destino	Contagem por BB	RAID / pool	Notas
MGS	1	RAID 1	Bloco de construção básico apenas; array 1
MDT	8	RAID 1	4 por array
OST	32	RAID 6 (TLC) ou DDP (QLC)	16 por array

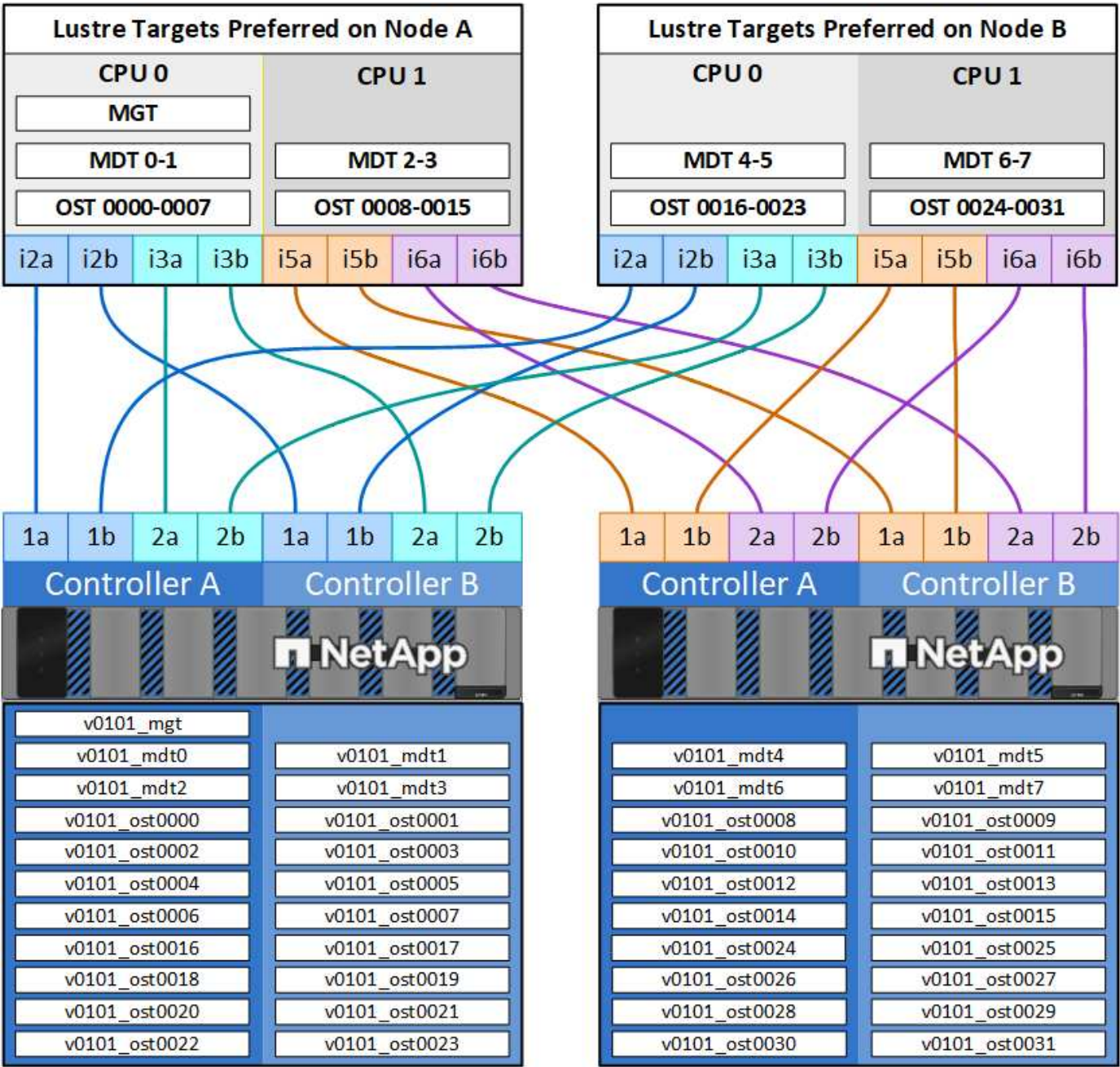
Utilize as seguintes diretrizes de dimensionamento de volumes como ponto de partida no inventário do Ansible.

Volume	Tamanho recomendado	Notas
MGS	5–10 GiB	Somente dados de configuração
MDT	Capacidade RAID 1 ÷ contagem MDT	Aproximadamente 1–2 TiB cada (típico)
OST	Capacidade RAID 6 ou DDP ÷ número de OSTs por pool	Escala com capacidade de unidade; veja "Orientação de dimensionamento"

Distribuição do volume do bloco de construção primário

A figura a seguir mostra o posicionamento preferencial do alvo Lustre e a conectividade NVMe-oF entre os nós do servidor OSS/MDS e os arrays E-Series em um bloco de construção EF80 básico. O diagrama identifica o alvo de gerenciamento como **MGT** (alvo de gerenciamento); um MGT hospeda o serviço MGS (servidor de gerenciamento) mencionado em outra parte deste documento.

Distribuição de destino do bloco de construção base (EF80)



A tabela a seguir lista como os volumes são distribuídos entre os dois arrays e qual servidor cada destino prefere.

Tipo de destino	Matriz 1	Array 2	Servidor 1	Servidor 2	Notas
MGS	1	0	1	0	Bloco de construção básico apenas
MDT	4	4	4	4	Cada servidor prefere 2 MDTs de cada array
OST	16	16	16	16	8 por pool RAID 6 × 2 pools por array, ou alocação DDP equivalente
Volumes totais	21	20	21	20	

Seleção do pool de armazenamento: TLC vs QLC

Escolha o tipo de pool do E-Series com base na capacidade da unidade NVMe nos arrays:

- **Unidades de 3,84 TB, 7,68 TB e 15,3 TB:** Use **grupos de volumes RAID 6** para volumes OST e **grupos de volumes RAID 1** para volumes MGS e MDT, ou use DDP para o pool de unidades compartilhadas.
- **Unidades Flash (QLC) com capacidade de 30,7 TB e 61,4 TB:** Use somente **Dynamic Disk Pools (DDP)**. Crie volumes RAID 1 dentro do DDP para armazenamento de MGS e MDT. Crie volumes RAID 6 para armazenamento de OST a partir do DDP.

Para estimativas de capacidade utilizável por bloco de construção, tamanho de unidade e layout, consulte "[Orientação de dimensionamento](#)".

Requisitos de rede

Backend (NVMe-oF)

- NVMe/InfiniBand ou NVMe/RoCE entre cada nó OSS/MDS e cada E-Series array
- MTU 9000 em interfaces de backend NVMe/RoCE (típico)
- Oito caminhos de cada nó Lustre até os arrays de armazenamento (quatro para cada array)
- O EF80 usa seis HCAs por nó (i2, i3, i5 e i6 para NVMe-oF; i1 e i4 para LNet). Conecte o Nó A às portas do controlador cujos rótulos terminam em a, como 1a e 2a. Conecte o Nó B às portas do controlador cujos rótulos terminam em b, como 1b e 2b. Consulte "[Cabeamento traseiro do EF80 de seis HCA](#)".

Frontend (LNet)

- IPoB ou RoCE para LNet (@o2ib tipo de rede)
- MTU 9000 em interfaces front-end RoCE (típico)
- Recomenda-se o uso de LNet multirail quando quatro portas frontend estiverem disponíveis (EF80: i1 e i4 HCAs, ambas as portas)
- Matriz de comutação dedicada InfiniBand ou RoCE conectando os nós do servidor OSS/MDS e os clientes Lustre
- RoCE sem perdas (PFC) recomendado em fabrics RoCE

Gerenciamento

- Rede de gerenciamento fora de banda para BMCs de servidor e portas de gerenciamento de arrays
- Rede Corosync dedicada ou fabric de frontend compartilhado (dependendo do site)

Lustre com o NetApp E-Series storage - Componentes de software

Revise a pilha de software e a automação do Ansible que implantam o Lustre com NetApp E-Series Storage. Para servidores, storage e redes, consulte "[Componentes de hardware](#)".

Software de blocos de construção

A tabela a seguir lista os componentes de software usados pelos arrays E-Series e pelos nós de servidor OSS/MDS. Use o "[NetApp Ferramenta de Matriz de Interoperabilidade \(IMT\)](#)" como fonte oficial para versões compatíveis e combinações de componentes. Pesquise por **E-Series Lustre** e confirme as versões do sistema operacional, Lustre, SANtricity OS, DOCA-OFED, firmware HCA e pacote HA antes da implantação.

Componente	Função
OS REST SANtricity	Executado em cada storage E-Series e fornece o SANtricity System Manager, alta disponibilidade com controladores dual-active, failover automático do caminho de I/O, balanceamento de carga automático e monitoramento de caminhos, Dynamic Disk Pools e RAID tradicional, reconstrução automática de unidades, proteção de cache espelhado, Data Assurance, monitoramento proativo da integridade das unidades e alterações online de software, firmware e configuração.
Lustre	Apresenta aos clientes um único namespace compatível com POSIX, enquanto distribui metadados e dados de arquivos em vários destinos para acesso paralelo. O servidor de gerenciamento (MGS) armazena a configuração do sistema de arquivos, os destinos de metadados (MDTs) gerenciam o namespace e os destinos de storage de objetos (OSTs) armazenam os dados dos arquivos nos volumes E-Series. Essa separação permite que o desempenho e a capacidade sejam ampliados com a adição de blocos de construção.
Red Hat Enterprise Linux (RHEL) ou Rocky Linux	Fornecer o kernel, os serviços do sistema e a estrutura de pacotes para os serviços de destino do Lustre, clustering HA e conectividade de storage e rede. Os repositórios de sistema operacional suportados também fornecem os pacotes Pacemaker, Corosync, agentes de fence, NVMe-oF e multipath, que são testados juntos como uma pilha de soluções.
Serviços de cluster de alta disponibilidade (Pacemaker, Corosync e fence agents)	Em conjunto, esses serviços proporcionam alta disponibilidade coordenada para destinos Lustre em todos os nós do servidor OSS/MDS. Eles mantêm a associação ao cluster e o quorum, monitoram recursos, orquestram o failover ordenado e isolam nós não responsivos antes de ativar destinos compartilhados em outros locais. Essa coordenação impede o acesso split-brain aos volumes E-Series e protege a integridade do sistema de arquivos durante falhas.

Componente	Função
NVIDIA DOCA-OFED	Fornece drivers HCA, bibliotecas RDMA e utilitários de gerenciamento para InfiniBand e fabrics RoCE. A mesma pilha de software oferece suporte ao acesso backend NVMe-oF de baixa latência aos arrays E-Series e à comunicação frontend LNet de alta taxa de transferência com clientes Lustre.

NetApp fornece pacotes RPM pré-compilados do servidor Lustre para Rocky Linux 9.8 e Red Hat Enterprise Linux 9.8.

Automação Ansible

O Lustre com NetApp E-Series storage é entregue e implementado usando automação Ansible a partir de um nó de controle com acesso de gerenciamento aos arrays E-Series e aos nós de servidor OSS/MDS. Um inventário Ansible modela a solução desejada, incluindo arrays, servidores, atribuições de storage, interfaces de rede e recursos de cluster de alta disponibilidade.

A coleção Ansible Lustre NetApp E-Series orquestra a implantação completamente usando as coleções SANtricity e Host. Juntas, essas coleções provisionam e mapeiam o storage E-Series, preparam o sistema operacional do servidor, configuram a conectividade NVMe-oF e LNet, implementam os targets Lustre e criam recursos Pacemaker. Essa automação torna a configuração repetível e simplifica a adição de blocos de construção a um sistema de arquivos existente.

A tabela a seguir resume os principais pacotes de software usados no nó de controle do Ansible. Consulte a ["Coleção Ansible Lustre do NetApp E-Series"](#) para ver as dependências de pacote atuais.

Pacote	Dependência ou propósito
Pitão	Fornece o runtime para o Ansible e os módulos Python necessários.
o ansible-core	Requer Python e executa as coleções Ansible do NetApp E-Series.
Pacotes adicionais do Python	cryptography, ipaddr, netaddr, passlib, resolvelib, jinja2, e pyyaml
Coleção Ansible Lustre do NetApp E-Series	Configura Lustre, LNet, conectividade de host NVMe-oF e recursos Pacemaker.
Coleção Ansible do NetApp E-Series SANtricity	Configura os arrays E-Series, pools de armazenamento, volumes e mapeamentos de hosts por meio da SANtricity Web Services API.
NetApp E-Series coleção Ansible de host E-Series	Prepara os nós do servidor OSS/MDS para conectividade LNet e NVMe-oF com os arrays E-Series.

Software cliente Lustre

O software cliente Lustre permite que um sistema acesse o sistema de arquivos paralelo através da interface LNet. Cada cliente precisa de módulos de kernel Lustre e utilitários de cliente Lustre que sejam compatíveis tanto com o kernel em execução quanto com a versão do Lustre implantada nos servidores.

Use o ["Matriz de suporte Lustre"](#) para encontrar um sistema operacional cliente e kernel compatíveis com sua versão do Lustre. NetApp não fornece pacotes cliente Lustre pré-compilados. Compile os pacotes cliente para o sistema operacional cliente e kernel a partir do código-fonte Lustre fornecido pela NetApp no ["repositório netapp-lustre"](#). Após a instalação dos pacotes cliente Lustre, a função `lustre_client` pode configurar interfaces de rede, LNet e criar uma unidade de montagem systemd persistente para o sistema de arquivos

Lustre.

Para obter as etapas de implantação do cliente, consulte ["Implante a solução"](#).

Lustre com o NetApp E-Series Storage - Guia de dimensionamento

Dimensione o Lustre com os blocos de construção de armazenamento NetApp E-Series pela capacidade e necessidades de metadados, decida quando adicionar capacidade e analise os fatores que afetam o desempenho.

Dimensionamento de capacidade

Um bloco de construção básico com dois arrays EF80 fornece o seguinte layout de destino Lustre. Para posicionamento de destino preferencial e caminhos NVMe-oF, consulte ["Distribuição do volume do bloco de construção primário"](#) em ["Componentes de hardware"](#).

Componente	Contagem	Tamanho típico do volume (por alvo)
MGS	1	5-10 GiB
MDT	8	O tamanho varia de acordo com a capacidade do disco e o layout RAID
OST	32	O tamanho varia de acordo com a capacidade do disco e o layout RAID

Cada array EF80 utiliza vinte e quatro unidades NVMe. As capacidades suportadas são 3,84 TB, 7,68 TB e 15,3 TB com grupos de volumes tradicionais (RAID 1 para MGS/MDT e RAID 6 para OST) ou DDP, e unidades Capacity Flash (QLC) de 30,7 TB ou 61,4 TB somente com DDP. Unidades de 1,92 TB não são recomendadas para esta solução no momento. Para obter informações sobre o layout dos slots de unidade e a seleção do pool, consulte ["Componentes de hardware"](#).

A tabela a seguir lista a capacidade utilizável aproximada para um bloco de construção EF80 (dois arrays, 48 discos). A capacidade utilizável do array não é a mesma que a capacidade utilizável final do sistema de arquivos Lustre. A alocação de inode, os journals, o sobreprovisionamento e as porcentagens de volume de inventário reduzem a capacidade disponível para os aplicativos.

Capacidade de unidade	Layout (por array)	Capacidade utilizável aproximada (bloco de construção)
3,84 TB	RAID 1 (2+2) + RAID 6 (10) + RAID 6 (10)	138,08 TB
3,84 TB	DDP (24 unidades, 2 reservadas)	133,63 TB
7,68 TB	RAID 1 (2+2) + RAID 6 (10) + RAID 6 (10)	276,35 TB
7,68 TB	DDP (24)	267,47 TB
15,3 TB	RAID 1 (2+2) + RAID 6 (10) + RAID 6 (10)	552,88 TB

Capacidade de unidade	Layout (por array)	Capacidade utilizável aproximada (bloco de construção)
15,3 TB	DDP (24)	535,15 TB
30,7 TB	Somente DDP	1.070,35 TB
61,4 TB	Somente DDP	2.162,70 TB

Dimensione os metadados e os dados separadamente e, em seguida, defina os tamanhos dos volumes no inventário do Ansible. Os modelos de implantação formatam os targets com `ldiskfs`; o MDT usa a proporção de inodes padrão do Lustre e os modelos OST definem `-i 4096`.

- **Metadados (MDT):** Dimensione os MDTs de acordo com a quantidade esperada de arquivos. Planeje aproximadamente o dobro da quantidade esperada de arquivos para o crescimento. Um MDT subdimensionado pode impedir a criação de novos arquivos mesmo quando ainda houver capacidade disponível no OST.
- **Dados (OST):** Dimensione OSTs com base na capacidade utilizável e nas necessidades de taxa de transferência.
- **MGS:** Utilize um alvo de gerenciamento pequeno (5-10 GiB) apenas para configuração do sistema de arquivos.

Ajuste os tamanhos de volume MDT e OST no inventário para a capacidade de unidade selecionada. Altere `format_options.mkfsoptions` em `eseries_lustre_filesystem_mdt` ou `eseries_lustre_filesystem_ost` somente quando um site precisar de uma proporção de inodes diferente. Para informações sobre proporções de inodes, consulte a ["Wiki do Lustre: ajuste do Lustre"](#). Para saber quando adicionar blocos de construção, consulte [Dimensionamento](#).

Dimensionamento

Adicione blocos de construção quando a capacidade ou a carga de metadados assim exigir:

- **Somente OST:** A contagem de arquivos e a carga de metadados estão dentro da capacidade atual do MDT, e você precisa de mais capacidade de dados ou taxa de transferência agregada. Adiciona 32 OSTs e dois nós de servidor OSS/MDS.
- **MDT+OST:** A contagem de arquivos ou a taxa de metadados está se aproximando da capacidade do MDT, ou você precisa de mais throughput de metadados e de capacidade de dados. Adiciona 8 MDTs e 32 OSTs.

Utilize `mdt.*.md_stats` em MDTs existentes para confirmar se os metadados estão saturados. Limite cada cluster Pacemaker/Corosync a cinco blocos de construção (dez nós de servidor OSS/MDS). Para implantações maiores, divida os blocos de construção uniformemente entre vários clusters de alta disponibilidade. Consulte ["Arquitetura da solução"](#) para as regras de escalonamento.

O exemplo a seguir mostra um sistema de arquivos com um bloco de construção básico e um bloco de construção somente para OST em um cluster de alta disponibilidade.

Bloco de construção	Tipo	Servidores	Matrizes	MGS	MDT	OST
BB1	Base	2	2	1	8	32
BB2	Somente OST	2	2	0	0	32

Bloco de construção	Tipo	Servidores	Matrizes	MGS	MDT	OST
Total		4	4	1	8	64

O BB2 registra seus alvos OST no MGS do BB1 usando `mgssnode=`. Os índices OST para o BB2 são atribuídos globalmente (por exemplo, OST0032 a OST0063) para que nenhum índice entre em conflito com o BB1.

Desempenho

A validação formal utiliza IOR, mdtest e fio de clientes Lustre para caracterizar a taxa de transferência relativa, IOPS e o comportamento de metadados, além de validar o failover de alta disponibilidade. Esses testes não divulgam números de desempenho garantido. Os benchmarks sintéticos medem o comportamento no melhor cenário e podem não refletir o desempenho real da aplicação.

O desempenho escala aproximadamente com o número de blocos de construção. Os resultados obtidos dependem do tipo de unidade e do layout do pool, da quantidade de caminhos NVMe-oF, da malha LNet e da quantidade de clientes, bem como da combinação de dados versus metadados em E/S.

Entre em contato com sua equipe de contas da NetApp para obter recomendações de dimensionamento e desempenho específicas para sua carga de trabalho.

Para obter informações sobre o layout das unidades e a contagem de volumes, consulte "[Componentes de hardware](#)". Para obter instruções sobre as etapas de implantação, consulte "[Implante a solução](#)". Para obter orientações sobre o inventário em campo, consulte "[Personalize o inventário do Lustre Ansible](#)".

Implante a solução

Implemente o Lustre com o NetApp E-Series Storage

Implemente o Lustre com o NetApp E-Series Storage concluindo as tarefas de hardware, preparação do ambiente, inventário, implantação do Ansible e verificação na ordem indicada.

Pré-requisitos

Antes de iniciar a implantação, certifique-se do seguinte:

- Você seguiu as orientações de dimensionamento e componentes básicos em "[Orientação de dimensionamento](#)" e selecionou o hardware que corresponde a "[Componentes de hardware](#)".
- Você confirmou que a combinação de servidor, HCA, E-Series, sistema operacional, Lustre, SANtricity OS e DOCA-OFED está listada para **E-Series Lustre** no "[NetApp Ferramenta de Matriz de Interoperabilidade](#)".
- Todos os nós de servidor OSS/MDS e os E-Series arrays para os blocos de construção planejados estão no local e prontos para instalar no rack.
- Você possui credenciais e acesso à rede para os BMCs do servidor, portas de gerenciamento do array e o host que servirá como nó de controle do Ansible.



NetApp oferece suporte à implementação do cluster Lustre HA e dos clientes Lustre com o "[Coleção Ansible Lustre do NetApp E-Series](#)". Não configure os destinos Lustre, a LNet, a

conectividade de host NVMe-oF ou os recursos do Pacemaker manualmente.

Fluxo de trabalho de implantação

Conclua as seguintes tarefas na ordem indicada:

1. ["Rack e cabeamento do hardware Lustre"](#).

Instale os servidores Lustre e os arrays EF80 no rack e, em seguida, conecte os cabos das redes NVMe-oF de backend e LNet de frontend.

2. ["Prepare o ambiente de implantação do Lustre"](#).

Configure os arrays e servidores e, em seguida, instale o Ansible e suas dependências no nó de controle.

3. ["Personalize o inventário do Lustre Ansible"](#).

Selecione o modelo para o fabric do site e preencha os servidores, arrays, redes, componentes básicos e credenciais.

4. ["Implante o cluster Lustre HA e os clientes"](#).

Prepare o sistema operacional do servidor, instale NVIDIA DOCA-OFED, execute o playbook de implantação principal e configure os clientes Lustre.

5. ["Verifique e expanda a implementação do Lustre"](#).

Confirme a integridade do cluster, os pontos de montagem e a capacidade antes de utilizá-lo em produção. Expanda o inventário existente quando o sistema de arquivos precisar de outro bloco de construção.

Informações relacionadas

- ["Coleção Ansible Lustre do NetApp E-Series"](#)
- ["NetApp Lustre repositório de origem"](#)
- ["Guia do usuário de administração de HA"](#)
- ["Guia de ajuste de desempenho"](#)
- ["Modelos de implantação do cliente Lustre"](#)

Hardware de rack e cabeamento Lustre

Coloque os servidores Lustre e os arrays NetApp EF80 no rack e, em seguida, conecte as malhas NVMe-oF de backend e LNet de frontend para cada bloco de construção.

Antes de começar

Consulte as contagens HCA, os requisitos de slot PCIe, os requisitos de caminho e as orientações de MTU em ["Componentes de hardware"](#).

Passos

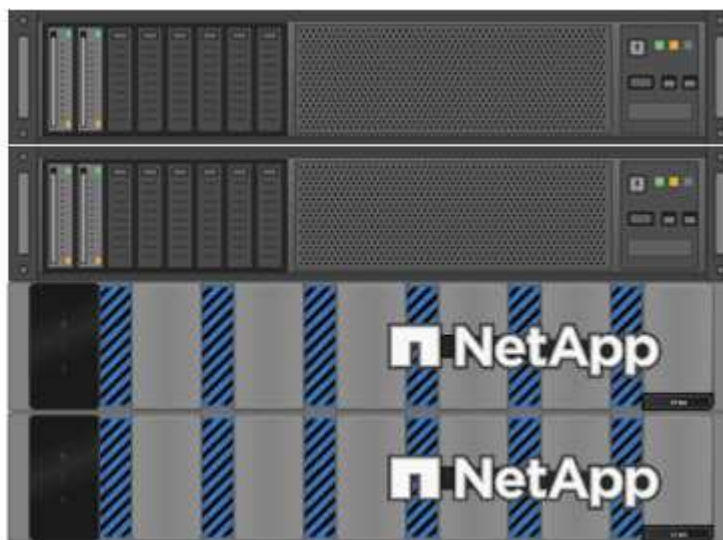
1. Instale e conecte todos os servidores Lustre e storage E-Series de acordo com a topologia do bloco de construção.

Node A

Node B

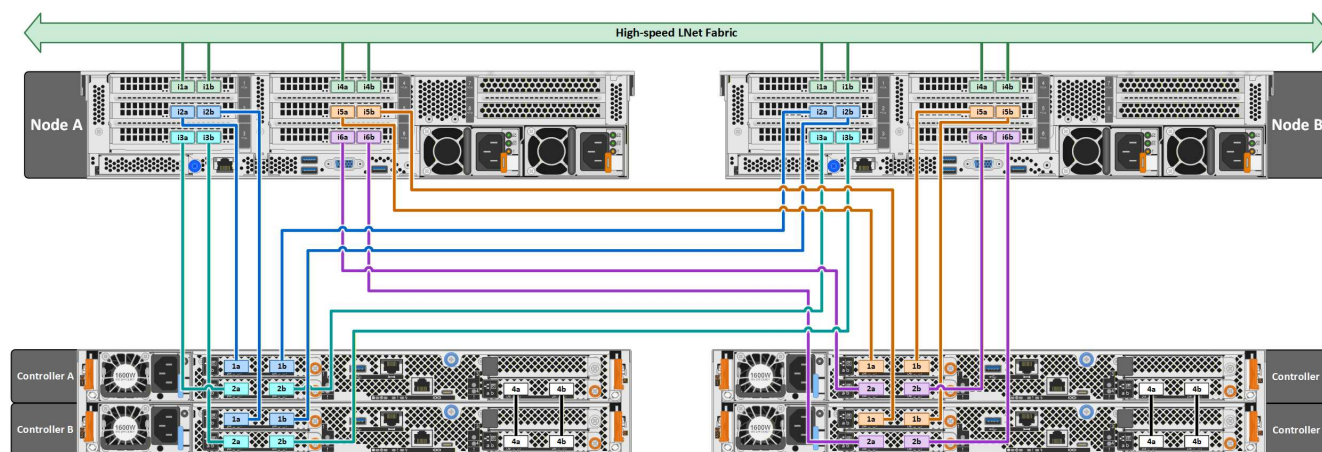
Array 1

Array 2



2. Para cada array EF80, conecte o controlador A à porta 4a do controlador B e o controlador A à porta 4b do controlador B. Essas conexões dedicadas fornecem espelhamento entre controladores e não são usadas para tráfego do host. Consulte ["Conecte os cabos de espelhamento entre os controladores EF50 e EF80"](#).
3. Conecte a rede NVMe-oF de backend entre servidores e arrays de acordo com a topologia de seis HCAs do EF80.

Cabeamento de backend EF80 de seis HCA (RoCE ou InfiniBand)



4. Conecte as quatro interfaces front-end LNet (os HCAs i1 e i4) em cada servidor Lustre e os clientes Lustre à malha de switches dedicada InfiniBand ou RoCE.
5. Ao usar RoCE, configure a malha para operação sem perdas com controle de fluxo prioritário (PFC). Os playbooks do Ansible configuram o RoCE sem perdas em todas as interfaces do bloco de construção.

Depois que você terminar

["Prepare os servidores, arrays e o nó de controle Ansible"](#).

Prepare o ambiente de implantação do Lustre

Configure o acesso de gerenciamento e as configurações básicas em cada array EF80 e servidor Lustre, depois prepare um nó de controle Ansible para implantar a solução.

Preparar as matrizes E-Series

Passos

1. Configure a interface de gerenciamento em cada controlador e verifique se o SANtricity System Manager está acessível.
2. Defina o nome do array e as credenciais administrativas que o inventário do Ansible utilizará.
3. Confirme se o array está em estado ideal e se todas as unidades estão presentes e reportando corretamente.

Para obter instruções sobre como configurar a conexão de gerenciamento, consulte o ["Documentação do E-Series"](#).

Prepare os servidores Lustre

Passos

1. Configure o controlador de gerenciamento da placa-mãe (BMC) em cada servidor para acesso fora de banda e defina as configurações do sistema UEFI para máximo desempenho.
2. Instale um sistema operacional compatível (RHEL ou Rocky Linux), depois configure o nome do host e a porta de rede de gerenciamento.
3. Prepare todos os pacotes de firmware e software para atender aos requisitos da solução **E-Series Lustre**, conforme descrito em ["Componentes de software"](#).

Prepare o nó de controle do Ansible

Designe um sistema Linux virtual ou físico com acesso à rede de gerenciamento a todos os servidores Lustre e arrays NetApp E-Series para ser o nó de controle do Ansible.

Os passos a seguir utilizam o Ubuntu 24.04. Para instruções para outra distribuição Linux, consulte o ["Documentação de instalação do Ansible"](#).

Passos

1. Instale o Python, o pip e o pacote de ambiente virtual do Python:

```
sudo apt-get install python3 python3-pip python3-setuptools python3-venv
```

2. Crie e ative um ambiente virtual Python:

```
python3 -m venv ~/pyenv  
source ~/pyenv/bin/activate
```

3. Instale o Ansible e os pacotes Python necessários no ambiente virtual ativado:

```
pip install "ansible-core>=2.19" cryptography jinja2 ipaddr netaddr \  
passlib pyyaml resolvelib
```

4. Instale a coleção Ansible Lustre da NetApp E-Series. A instalação da coleção também instala as coleções dependentes, incluindo as coleções SANtricity e Host:

```
ansible-galaxy collection install netapp_eseries.lustre
```

5. Verifique as versões instaladas do Ansible, Python e da coleção:

```
ansible --version
python3 --version
ansible-galaxy collection list netapp_eseries.lustre
```

6. Configure o SSH sem senha do nó de controle para cada servidor Lustre. Substitua <ip_or_hostname> pelo endereço IP de gerenciamento ou nome do host de um servidor:

```
ssh-keygen
ssh-copy-id <ip_or_hostname>
```

Repita `ssh-copy-id` para cada servidor Lustre.



Não configure SSH sem senha para os arrays E-Series. O Ansible gerencia os arrays por meio da SANtricity Web Services API, utilizando as credenciais definidas no inventário.

Depois que você terminar

["Personalize o inventário do Ansible"](#).

Personalize o inventário Ansible do Lustre

Crie uma cópia funcional de um modelo de implantação do EF80 e personalize seu inventário para seus servidores, arrays, redes e blocos de construção do Lustre.

Mantenha o inventário selecionado em seu diretório de malha e plataforma. Mantenha o diretório `playbooks\compartilhado` e o arquivo `passwords.yml` compartilhado na raiz de `\building_blocks`.

Selecione um modelo de implantação

Comece com o modelo que corresponde à malha de rede:

- **InfiniBand:** `"IB_H2_IB_DAC_A2_EF80/lustre_building_block_cluster_1"`
- **RoCE:** `"ROCE_H2_ROCE_DAC_A2_EF80/lustre_building_block_cluster_1"`

Configurar as configurações específicas do site

Defina os valores específicos do site no inventário, juntamente com o repositório local e as configurações de regras udev. As referências de linha na tabela apontam para os arquivos de modelo RoCE EF80 na tag `ansible-lustre release/1.0.0`. O modelo InfiniBand usa os mesmos nomes de variáveis e estrutura, exceto pela variável de interface LNet, que é indicada na tabela.

Campo	Variável Ansible	Arquivo de modelo (release/1.0.0)	Escopo	Notas
Nome do sistema de arquivos	<code>eseries_lustre_filesystem_mgt.format_options.fsname,</code> <code>eseries_lustre_filesystem_mdt.format_options.fsname,</code> <code>eseries_lustre_filesystem_ost.format_options.fsname</code>	" ha_cluster.yml#L76 ", " #L90 ", " #L110 "	Em todo o cluster	Aninhado em <code>format_options</code> para os mapas MGT, MDT e OST. Máximo 8 caracteres. Use o mesmo valor em todos os três locais. Uma chave <code>fsname</code> de nível superior não tem efeito.
MGS NIDs para registro de alvos	<code>eseries_lustre_filesystem_mdt.format_options.mgsnode,</code> <code>eseries_lustre_filesystem_ost.format_options.mgsnode</code>	" ha_cluster.yml#L98 ", " #L118 "	Em todo o cluster	Aninhado em MDT e OST <code>format_options</code> . Inclua os NIDs da rede LNet de ambos os nós no par de HA. Liste primeiro os NIDs do nó configurado como proprietário preferencial do MGS, seguidos pelos NIDs do nó parceiro.
Nome do cluster Pacemaker	<code>eseries_lustre_pacemaker_cib_properties_overrides.cluster_properties.cluster-name</code>	" ha_cluster.yml#L228 "	Em todo o cluster	Aninhado em substituições de propriedade CIB do Pacemaker. Nome exclusivo para o cluster de HA.
Endereços de fencing do BMC	<code>eseries_lustre_pacemaker_fencing_agents.fence_redfish.ip</code>	" ha_cluster.yml#L211 ", " #L214 ", " #L217 ", " #L220 "	Repita para cada servidor Lustre	Endereço IP do BMC para cada nó do servidor.
Destinatários do e-mail de alerta	<code>eseries_lustre_alert_email_list</code>	" ha_cluster.yml#L233 "	Em todo o cluster	Destinatários para notificações de recursos de HA e falhas.
Domínio de e-mail	<code>eseries_lustre_mail_service_overrides.conf.mydomain</code>	" ha_cluster.yml#L240 "	Em todo o cluster	Domínio Postfix usado para enviar alertas por e-mail.
Fontes de tempo NTP	<code>eseries_lustre_time_sync_overrides.server_pools</code>	" ha_cluster.yml#L250 "	Em todo o cluster	Fontes de tempo para os nós do cluster. Especifique um pool NTP ou um ou mais servidores NTP individuais. Inclua a diretiva obrigatória <code>pool</code> ou <code>server</code> e opções como <code>iburst</code> .

Campo	Variável Ansible	Arquivo de modelo (release/1.0.0)	Escopo	Notas
Mapeamento udev de interface PCI para nome	<code>eseries_ip_udev_rules</code>	" ha_cluster.yml#L172 "	Repita por layout de slot HCA	Mapeie cada slot PCI para um nome de interface persistente (valores em L182). Colete com <code>lspci -nnvmm</code> .
Repositório de pacotes local (opcional)	<code>eseries_common_yum_repos</code>	" ha_cluster.yml#L256 ", " #L261 "	Em todo o cluster	Para instalações isoladas da internet, remova o comentário do bloco e defina o repositório url (L264).
IP de gerenciamento de servidor	<code>ansible_host</code>	" host_vars/lustre_01.yml#L10 "	Repita para cada servidor Lustre	Um <code>host_vars/lustre_NN.yml</code> arquivo por servidor.
Interfaces de backend NVMe-oF	<code>eseries_nvme_roce_interfaces</code> (RoCE) / <code>eseries_nvme_ib_interfaces</code> (InfiniBand)	" RoCE host_vars/lustre_01.yml#L21 ", " InfiniBand host_vars/lustre_01.yml#L17 "	Repita para cada servidor Lustre	Endereços da interface de backend usados para o tráfego de armazenamento NVMe-oF para os E-Series arrays.
Interfaces de cluster LNet	<code>eseries_roce_interfaces</code> (RoCE) / <code>eseries_ipoib_interfaces</code> (InfiniBand)	" RoCE host_vars/lustre_01.yml#L47 ", " InfiniBand host_vars/lustre_01.yml#L38 "	Repita para cada servidor Lustre	Endereços da interface LNet de front-end (valores RoCE na linha 56; InfiniBand valores na linha 47).
Endereços IP dos nós Corosync	<code>eseries_lustre_corosync_node_ips</code>	" host_vars/lustre_01.yml#L58 "	Repita para cada servidor Lustre	Liste todos os endereços IP dos nós no cluster de HA (valores na linha 62).
NIDs do serviço de destino Lustre	<code>lustre_target_service_nodes</code>	" host_vars/lustre_01.yml#L64 "	Repita para cada servidor Lustre	Inclua os NIDs de LNet de front-end (<code>@o2ib</code> de ambos os nós no par de HA para que os serviços de destino permaneçam acessíveis após o failover (valores na linha 67).
Endereço IP de gerenciamento do array	<code>ansible_host</code>	" host_vars/netapp_01.yml#L13 "	Repita para cada matriz E-Series	IP de gerenciamento de um controlador; a URL da API deriva dele.

Campo	Variável Ansible	Arquivo de modelo (release/1.0.0)	Escopo	Notas
Associação ao grupo de blocos de construção	mgs / mds_NN / oss_NN	"lustre_inventory.yml"	Por bloco de construção	Cada nome de host de inventário deve corresponder ao nome base do seu arquivo host_vars correspondente. Por exemplo, o host lustre_01 usa host_vars/lustre_01.yml, e netapp_01 usa host_vars/netapp_01.yml. Inclua o grupo MGS apenas no primeiro bloco de construção. Inclua os grupos MDS no bloco de construção base e em cada bloco de construção de expansão MDT+OST; omita os grupos MDS apenas dos blocos de construção de expansão somente OST. Registre os alvos de expansão no MGS existente com mgsnode.

Repita as configurações para cada host e bloco de construção. Crie um `host\_vars/lustre\_NN.yml` arquivo por servidor Lustre e um `host\_vars/netapp\_NN.yml` arquivo por array, e repita os grupos de inventário para cada bloco de construção adicional.

Os playbooks de implantação carregam as credenciais do array, Pacemaker e BMC a partir do arquivo `building_blocks/passwords.yml` compartilhado. Preencha este arquivo e criptografe-o com o Ansible Vault antes da implantação ou substitua os valores com segurança em tempo de execução com `--extra-vars`. Não inclua credenciais em texto simples no controle de versão.



Os endereços IP, nomes de host, nomes de interface e o dimensionamento de volumes nos modelos de implantação são exemplos. Modifique todos os arquivos de inventário do ambiente antes de executar qualquer playbook.

Depois que você terminar

"Implante o cluster Lustre HA e os clientes".

Implante o cluster Lustre HA e os clientes

Prepare o sistema operacional, instale NVIDIA DOCA-OFED, implemente o cluster Lustre HA e configure os clientes Lustre usando os playbooks do Ansible da NetApp.

Antes de começar, complete e assegure o inventário descrito em "[Personalize o inventário do Lustre Ansible](#)".

Implante o cluster Lustre HA

Passos

1. A partir do `building_blocks/playbooks` diretório na sua cópia de trabalho, prepare o sistema operacional nos servidores Lustre. Substitua `<inventory_path>` pelo caminho para o seu diretório de inventário personalizado:

```
ansible-playbook -i <inventory_path>/lustre_inventory.yml
deploy_lustre_os/deploy_lustre_os_playbook.yml
```

2. Instale o NVIDIA DOCA-OFED em cada servidor Lustre. Compile os módulos do kernel usando o kernel que foi instalado na etapa anterior. Siga o procedimento para o seu kernel em ["Instalação e atualização do NVIDIA DOCA-Host"](#). O exemplo a seguir mostra uma instalação do DOCA-OFED no RHEL ou Rocky Linux.

- a. Instale o pacote do repositório do host DOCA e atualize o cache de pacotes. Substitua `<doca_host_package>` pelo pacote do host DOCA correspondente ao seu sistema operacional e arquitetura:

```
dnf install -y wget tar
wget https://www.mellanox.com/downloads/DOCA/<doca_host_package>.rpm
rpm -i <doca_host_package>.rpm
dnf clean all
dnf makecache
```

- b. Compile os módulos do kernel DOCA para o kernel em execução. O script imprime o caminho do pacote que cria:

```
dnf install -y doca-extra
/opt/mellanox/doca/tools/doca-kernel-support
```

- c. Instale o pacote do repositório de módulos do kernel criado pelo script e, em seguida, atualize o cache de pacotes. Substitua `<doca_kernel_repo>` pelo caminho obtido na etapa anterior:

```
rpm -Uvh <doca_kernel_repo>.rpm
dnf makecache
```

- d. Instale os pacotes DOCA-OFED para espaço de usuário, kernel e NVMe. Substitua `<kernel_version>` pela versão do kernel em execução:

```
dnf install -y doca-Ofed-userspace
dnf install -y --disablerepo=doca doca-kernel-<kernel_version>
dnf install -y kmod-mlnx-nvme
```

3. Execute o playbook de implantação principal:

```
ansible-playbook -i <inventory_path>/lustre_inventory.yml  
lustre_playbook.yml
```



Ajuste `--forks` de acordo com o tamanho da implantação para controlar quantos hosts o Ansible configura em paralelo. Por exemplo, adicione `--forks 20` para um cluster maior.

O playbook provisiona grupos de volumes E-Series ou pools e volumes DDP, configura a conectividade de host NVMe-oF, formata e monta os targets Lustre, configura o LNet, aplica ajuste de desempenho e configura os recursos HA do Pacemaker.



Uma implantação com um único bloco de construção forma um cluster Pacemaker de dois nós. O nó A recebe um voto extra em caso de failover. Para garantir quorum em um cluster de dois nós, considere configurar um qdevice. Quando uma implantação contém vários blocos de construção, cada bloco contribui com um par de servidores de dois nós para o cluster Pacemaker maior, até o limite documentado do cluster. Revise a configuração de fencing e quorum no ["Guia do usuário de administração de HA"](#) para que o cluster possa recuperar os alvos com segurança durante uma falha de nó.

Implantar clientes Lustre

Implante clientes Lustre em qualquer sistema que exija acesso ao sistema de arquivos, personalizando o modelo de implantação do cliente e executando o playbook do cliente.

Passos

1. Selecione um sistema operacional de cliente e kernel compatíveis no ["Matriz de suporte Lustre"](#). Compile e instale os pacotes de cliente Lustre para esse kernel a partir de ["netapp-lustre"](#) caso ainda não estejam instalados.
2. Crie uma cópia de trabalho do ``clients`` diretório de template, incluindo seu ``passwords.yml`` arquivo compartilhado, e selecione o template que corresponde à sua fabric:
 - **InfiniBand:** ["IB_lustre_cluster_clients"](#)
 - **RoCE:** ["Clientes do cluster ROCE_lustre"](#)
3. Personalize `lustre_client_inventory.yml`, `host_vars` e `group_vars` para seus clientes. As referências de linha na tabela apontam para os arquivos de modelo de cliente RoCE na tag `ansible-lustre release/1.0.0`; o modelo de cliente InfiniBand usa as mesmas variáveis, exceto quando indicado.

Campo	Variável Ansible	Arquivo de modelo (release/1.0.0)	Escopo	Notas
Hosts do cliente	<code>lustre_clients</code>	"lustre_client_inventory.yml#L4"	Por cliente	Liste o nome do host de cada cliente.
Usuário SSH do cliente e senha become	<code>ssh_client_user</code> / <code>ssh_client_become_pass</code>	"passwords.yml#L2"	Em todo o cluster	Defina a senha "become" somente se o usuário SSH não for root.

Campo	Variável Ansible	Arquivo de modelo (release/1.0.0)	Escopo	Notas
Interfaces LNet	<code>eseries_lustre_lnet</code>	" group_vars/all.yml#L7 "	Em todo o cluster	Nomes de interface LNet do lado do cliente (valores na L13).
Montar NIDs do MGS	<code>lustre_client_mounts.mgsnode</code>	" group_vars/all.yml#L17 "	Em todo o cluster	NIDs do MGS usados para montar o sistema de arquivos (valores em L20).
Nome do sistema de arquivos	<code>lustre_client_mounts.fsnome</code>	" group_vars/all.yml#L21 "	Em todo o cluster	Deve corresponder ao cluster <code>fsname</code> .
Ponto de montagem	<code>lustre_client_mounts.mount_point</code>	" group_vars/all.yml#L22 "	Em todo o cluster	Diretório de montagem do cliente.
IP de gerenciamento do cliente	<code>ansible_host</code>	" host_vars/client_01.yml#L4 "	Repita para cada cliente	Um <code>host_vars/client_NN.yml</code> a rquivo por cliente.
Interfaces de malha de armazenamento	<code>eseries_roce_interfaces</code>	" host_vars/client_01.yml#L10 "	Repita para cada cliente	Endereços da interface de front-end (valores na linha 15). O InfiniBand template usa <code>eseries_ipoib_interfaces</code> aqui.

Repita as configurações para cada cliente. Crie um arquivo `host_vars/client_NN.yml` para cada cliente e adicione o host correspondente a `lustre_client_inventory.yml`. Preencha o arquivo compartilhado `clients/passwords.yml` e criptografe-o com o Ansible Vault antes de executar o playbook do cliente.

4. Execute o playbook do cliente a partir do diretório de modelos do cliente:

```
ansible-playbook -i lustre_client_inventory.yml
lustre_client_playbook.yml
```

O playbook do cliente configura as interfaces de rede do cliente e o LNet, e pode criar uma unidade de montagem `systemd` persistente para o sistema de arquivos Lustre.

Depois que você terminar

"[Verifique a implantação](#)".

Verifique e expanda a implementação do Lustre

Confirme a integridade do cluster, as montagens do Lustre e a capacidade do sistema de arquivos antes de usar em produção. Em seguida, use o mesmo inventário para adicionar componentes conforme necessário.

Verifique a implantação

Passos

1. A partir de um servidor Lustre, confirme se todos os recursos do Pacemaker estão iniciados nos nós esperados:

```
pcs status
```

2. A partir de um cliente Lustre, confirme se o sistema de arquivos Lustre está montado:

```
mount -t lustre
```

3. A partir de um cliente Lustre, confirme a capacidade do sistema de arquivos e se os destinos MDT e OST estão visíveis:

```
lfs df -h
```

Para validação de desempenho com IOR, mdtest e fio, consulte ["Orientação de dimensionamento"](#).

Para migração controlada de destino ou teste de failover de HA, siga os procedimentos em ["Guia do usuário de administração de HA"](#).

Expandir o sistema de arquivos

Para adicionar capacidade ou desempenho de metadados, expanda seu inventário existente com um bloco de construção adicional. Não crie um inventário separado para o novo bloco de construção. O playbook atribui automaticamente índices MDT e OST exclusivos e registra os novos destinos no MGS existente.

Passos

1. Adicione os novos hosts, arrays e grupos de recursos (MDT+OST ou somente OST) ao mesmo inventário e `host_vars` e `group_vars` arquivos que você usou para a implantação original.
2. Execute novamente o playbook de implantação principal no inventário atualizado.

Consulte ["Arquitetura da solução"](#) as regras de dimensionamento e ["Orientação de dimensionamento"](#) as orientações sobre quando adicionar cada tipo de bloco de construção.

Lustre com o NetApp E-Series Storage - Recursos relacionados

Use esses links para acessar documentação, ferramentas e softwares relacionados ao planejar ou expandir uma implantação do Lustre com o NetApp E-Series Storage. Para informações sobre dimensionamento e etapas de implantação, consulte ["Orientação de dimensionamento"](#) e ["Implante a solução"](#).

NetApp recursos

- ["NetApp Ferramenta de Matriz de Interoperabilidade"](#). Pesquise por **E-Series Lustre**.
- ["NetApp EF-Series datasheet do array all-flash"](#)
- ["NetApp SANtricity System Manager documentação"](#)
- ["Coleção NetApp ansible-lustre"](#). Para orientações sobre o campo de inventário, consulte ["Personalize o inventário do Lustre Ansible"](#).
- ["NetApp Lustre repositório de origem"](#). NetApp fornece pacotes RPM pré-compilados do servidor Lustre para Rocky Linux 9.8 e Red Hat Enterprise Linux 9.8. Compile os pacotes de cliente Lustre para o sistema operacional e kernel do cliente a partir do código-fonte da NetApp.

Comunidade Lustre

- ["Sistema de arquivos Lustre"](#)
- ["Matriz de suporte Lustre"](#) para sistemas operacionais de cliente e kernels
- ["Wiki do Lustre: ajuste do Lustre"](#)

Soluções de infraestrutura de IA relacionadas da NetApp

- ["BeeGFS no NetApp com armazenamento da série E"](#)
- ["Implementando o IBM Spectrum Scale com NetApp E-Series Storage"](#)
- ["NVIDIA DGX SuperPOD com NetApp Série EF"](#)

Informações sobre direitos autorais

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

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

ESTE SOFTWARE É FORNECIDO PELA NETAPP "NO PRESENTE ESTADO" E SEM QUAISQUER GARANTIAS EXPRESSAS OU IMPLÍCITAS, INCLUINDO, SEM LIMITAÇÕES, GARANTIAS IMPLÍCITAS DE COMERCIALIZAÇÃO E ADEQUAÇÃO A UM DETERMINADO PROPÓSITO, CONFORME A ISENÇÃO DE RESPONSABILIDADE DESTES DOCUMENTOS. EM HIPÓTESE ALGUMA A NETAPP SERÁ RESPONSÁVEL POR QUALQUER DANO DIRETO, INDIRETO, INCIDENTAL, ESPECIAL, EXEMPLAR OU CONSEQUENCIAL (INCLUINDO, SEM LIMITAÇÕES, AQUISIÇÃO DE PRODUTOS OU SERVIÇOS SOBRESSALIENTES; 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.