



定义 **BeeGFS** 文件系统

BeeGFS on NetApp with E-Series Storage

NetApp

January 27, 2026

目录

- 定义BeeGFS文件系统 1
 - Ansible清单概述 1
 - 概述 1
 - 步骤 1
 - 规划文件系统 2
 - 概述 2
 - 步骤 2
 - 定义文件和块节点 3
 - 配置单个文件节点 3
 - 配置单个块节点 6
 - 指定通用文件节点配置 10
 - 指定通用块节点配置 17
 - 定义BeeGFS服务 19
 - 定义BeeGFS管理服务 19
 - 定义BeeGFS元数据服务 21
 - 定义BeeGFS存储服务 23
 - 将BeeGFS服务映射到文件节点 25
 - 概述 25
 - 步骤 25

定义BeeGFS文件系统

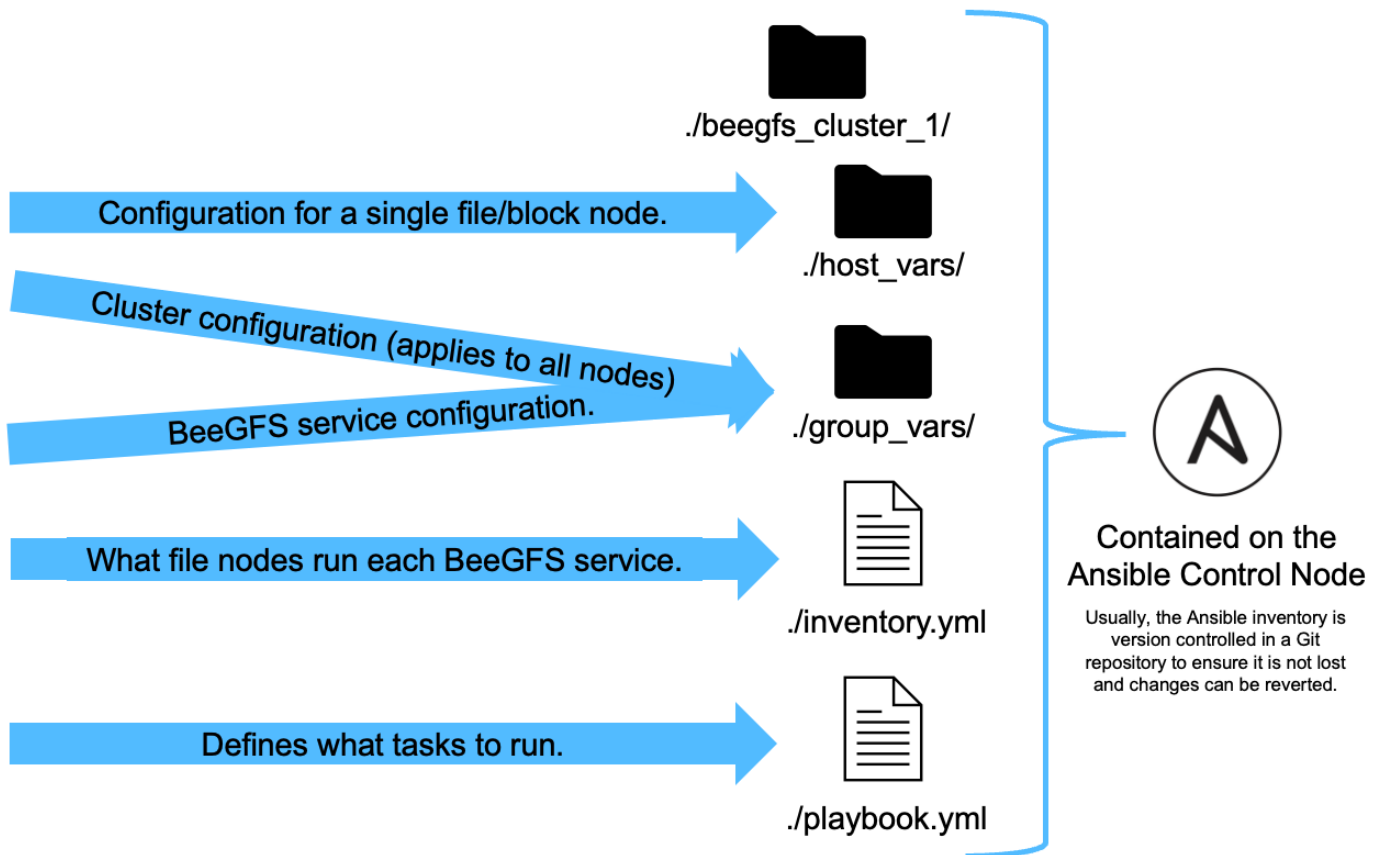
Ansible清单概述

Ansible清单是一组配置文件、用于定义所需的BeeGFS HA集群。

概述

建议遵循标准的Ansible实践来组织 "清单"、包括使用 "子目录/文件" 而不是将整个清单存储在一个文件中。

单个BeeGFS HA集群的Ansible清单组织如下：



由于一个BeeGFS文件系统可以跨越多个HA集群、因此大型安装可能具有多个Ansible清单。通常、不建议尝试将多个HA集群定义为一个Ansible清单以避免出现问题。

步骤

1. 在Ansible控制节点上、创建一个空目录、该目录将包含要部署的BeeGFS集群的Ansible清单。
 - a. 如果您的文件系统最终将包含/可能包含多个HA集群、建议先为文件系统创建一个目录、然后为表示每个HA集群的清单创建子目录。例如：

```
beegfs_file_system_1/
  beegfs_cluster_1/
  beegfs_cluster_2/
  beegfs_cluster_N/
```

2. 在包含要部署的HA集群清单的目录中、创建两个目录 `group_vars` 和 `host_vars` 和两个文件 `inventory.yml` 和 `playbook.yml`。

以下各节将逐步介绍如何定义每个文件的内容。

规划文件系统

在构建Ansible清单之前规划文件系统部署。

概述

在部署文件系统之前、您应定义集群中运行的所有文件节点、块节点和BeeGFS服务需要哪些IP地址、端口和其他配置。虽然具体配置会因集群架构而异、但本节定义了一般适用的最佳实践和应遵循的步骤。

步骤

1. 如果您使用基于IP的存储协议(例如iSER、iSCSI、NVMe/IB或NVMe/RoCE)将文件节点连接到块节点、请为每个组件填写以下工作表。一个构建块中的每个直接连接都应具有一个唯一的子网、并且不应与用于客户端-服务器连接的子网重叠。

文件节点	IB端口	IP 地址	块节点	IB端口	物理IP	虚拟IP (仅适用于具有HDR IB的EF600)
<HOSTNAME >	<PORT>	<IP/SUBNET >	<HOSTNAME >	<PORT>	<IP/SUBNET >	<IP/SUBNET >



如果每个构建块中的文件和块节点直接连接、您通常可以对多个构建块重复使用相同的IP/方案。

2. 无论您是在存储网络中使用InfiniBand还是基于融合以太网的RDMA (RoCE)、请填写以下工作表以确定将用于HA集群服务、BeeGFS文件服务和客户端进行通信的IP范围：

目的	InfiniBand端口	IP地址或范围
BeeGFS集群IP	<INTERFACE(s)>	<RANGE>
BeeGFS管理	<INTERFACE(s)>	<IP(s)>
BeeGFS元数据	<INTERFACE(s)>	<RANGE>
BeeGFS存储	<INTERFACE(s)>	<RANGE>
BeeGFS客户端	<INTERFACE(s)>	<RANGE>

- a. 如果您使用的是一个IP子网、则只需要一个工作表、否则还需要填写第二个子网的工作表。
3. 根据上述内容、为集群中的每个组件填写以下工作表、以定义要运行的BeeGFS服务。对于每个服务、请指定首选/二级文件节点、网络端口、浮动IP、NUMA分区分配(如果需要)以及要用于其目标的块节点。填写工作表时、请参见以下准则：
 - a. 将BeeGFS服务指定为 `mgmt.yml`，`meta_<ID>.yml` 或 `storage_<ID>.yml` 其中ID表示此文件系统中此类型的所有BeeGFS服务的唯一编号。此约定将简化在创建用于配置每个服务的文件时在后续章节中引用此工作表的过程。
 - b. 用于BeeGFS服务的端口只需在特定组件中是唯一的。确保具有相同端口号的服务不能在同一文件节点上运行、以避免端口冲突。
 - c. 如果需要、服务可以使用多个块节点和/或存储池中的卷(并非所有卷都需要归同一控制器所有)。多个服务也可以共享同一个块节点和/或存储池配置(各个卷将在后面的章节中进行定义)。

BeeGFS服务(文件名)	文件节点	Port	浮动IP	NUMA区域	块节点	存储池	所属控制器
<SERVICE TYPE> <ID>	<PREFERRED FILE NODE> <SECONDARY FILE NODE(s)>	<PORT>	<INTERFACE> : <IP/SUBNET> <INTERFACE> : <IP/SUBNET>	<NUMA NODE/ZONE>	<BLOCK NODE>	<STORAGE POOL/VOLUME GROUP>	<A OR B>

有关标准约定、最佳实践和填写的示例工作表的详细信息、请参阅["最佳实践"定义BeeGFS组件](#)NetApp经验证的架构上BeeGFS的和部分。

定义文件和块节点

配置单个文件节点

使用主机变量(host_vars)指定各个文件节点的配置。

概述

本节将逐步介绍填充 `host_vars/<FILE_NODE_HOSTNAME>.yml` 集群中每个文件节点的文件。这些文件只能包含特定文件节点独有的配置。这通常包括：

- 定义Ansible连接到节点时应使用的IP或主机名。
- 配置用于HA集群服务(Pacemaker和Corosync)的其他接口和集群IP、以便与其他文件节点进行通信。默认情况下、这些服务与管理接口使用相同的网络、但应提供更多接口以实现冗余。通常的做法是、在存储网络上定义其他IP、从而避免需要额外的集群或管理网络。
 - 用于集群通信的任何网络的性能对于文件系统性能并不重要。使用默认集群配置时、通常至少1 Gb/秒的网络可为集群操作(例如同步节点状态和协调集群资源状态更改)提供足够的性能。慢速/繁忙网络可能会使发生原因 资源状态发生变化、所需时间比平常要长、在极端情况下、如果节点无法在合理的时间范围内发送检测信号、可能会导致节点被逐出集群。
- 配置用于通过所需协议连接到块节点的接口(例如：iSCSI/iSER、NVMe/IB、NVMe/RoCE、FCP等)

步骤

请参考["规划文件系统"](#)本节中定义的IP寻址方案、为集群中的每个文件节点创建一个文件`host\_vars/<FILE\_NODE\_HOSTNAME>/yaml`并按如下所示进行填充：

1. 在顶部、指定Ansible通过SSH连接到节点并对其进行管理时应使用的IP或主机名：

```
ansible_host: "<MANAGEMENT_IP>"
```

2. 配置可用于集群流量的其他IP：

- a. 网络类型为 ["InfiniBand \(使用IPoIB\)"](#)：

```
eseries_ipoib_interfaces:
- name: <INTERFACE> # Example: ib0 or ilb
  address: <IP/SUBNET> # Example: 100.127.100.1/16
- name: <INTERFACE> # Additional interfaces as needed.
  address: <IP/SUBNET>
```

- b. 网络类型为 ["基于融合以太网的RDMA \(RoCE\)"](#)：

```
eseries_roce_interfaces:
- name: <INTERFACE> # Example: eth0.
  address: <IP/SUBNET> # Example: 100.127.100.1/16
- name: <INTERFACE> # Additional interfaces as needed.
  address: <IP/SUBNET>
```

- c. 网络类型为 ["以太网\(仅限TCP、无RDMA\)"](#)：

```
eseries_ip_interfaces:
- name: <INTERFACE> # Example: eth0.
  address: <IP/SUBNET> # Example: 100.127.100.1/16
- name: <INTERFACE> # Additional interfaces as needed.
  address: <IP/SUBNET>
```

3. 指示哪些IP应用于集群流量、优先IP列在上面：

```
beegfs_ha_cluster_node_ips:
- <MANAGEMENT_IP> # Including the management IP is typically but not
  required.
- <IP_ADDRESS> # Ex: 100.127.100.1
- <IP_ADDRESS> # Additional IPs as needed.
```



步骤2中配置的IP不会用作集群IP、除非它们包含在中 `beegfs_ha_cluster_node_ips` 列表这样、您就可以使用Ansible配置其他IP /接口、如果需要、这些IP /接口可用于其他目的。

4. 如果文件节点需要通过基于IP的协议与块节点进行通信、则需要在相应的接口上配置IP、并安装/配置该协议所需的任何软件包。

- a. 如果使用 "iSCSI":

```
eseries_iscsi_interfaces:
- name: <INTERFACE> # Example: eth0.
  address: <IP/SUBNET> # Example: 100.127.100.1/16
```

- b. 如果使用 "iSER":

```
eseries_ib_iser_interfaces:
- name: <INTERFACE> # Example: ib0.
  address: <IP/SUBNET> # Example: 100.127.100.1/16
  configure: true # If the file node is directly connected to the
block node set to true to setup OpenSM.
```

- c. 如果使用 "NVMe/IB":

```
eseries_nvme_ib_interfaces:
- name: <INTERFACE> # Example: ib0.
  address: <IP/SUBNET> # Example: 100.127.100.1/16
  configure: true # If the file node is directly connected to the
block node set to true to setup OpenSM.
```

- d. 如果使用 "NVMe/RoCE":

```
eseries_nvme_roce_interfaces:
- name: <INTERFACE> # Example: eth0.
  address: <IP/SUBNET> # Example: 100.127.100.1/16
```

- e. 其他协议:

- i. 如果使用 "NVMe/FC"、不需要配置各个接口。BeeGFS集群部署将根据需要自动检测协议和安装/配置要求。如果您使用网络结构连接文件和块节点、请确保按照NetApp和交换机供应商的最佳实践正确对交换机进行分区。
- ii. 使用FCP或SAS不需要安装或配置其他软件。如果使用FCP、请确保交换机已正确分区、如下所示 "NetApp" 以及交换机供应商的最佳实践。
- iii. 目前不建议使用IB SRP。根据E系列块节点支持的内容、使用NVMe/IB或iSER。

单击 ["此处"](#) 有关表示单个文件节点的完整清单文件的示例。

高级：在以太网模式和InfiniBand模式之间切换NVIDIA ConnectX VPI适配器

NVIDIA ConnectX-Virtual Protocol Interconnect & reg; (VPI)适配器既支持InfiniBand、也支持以太网作为传输层。不会自动协商模式之间的切换，必须使用中包含的工具进行配置 `mstconfig`，该工具 `mstflint` 是一个开放源代码软件包 ["NVIDIA软件工具 \(MFT\)"](#)。只需更改一次适配器的模式即可。这可以手动完成、也可以作为使用清单部分配置的任何接口的一部分包含在AndS得以 自动检查/应用的清单中 ``eseries-[ib|ib_iser|ipoib|nvme_ib|nvme_roce|roce]_interfaces:。`

例如、要将InfiniBand模式下的接口电流更改为以太网、以便用于RoCE：

1. 对于要配置的每个接口、请指定 `mstconfig` 作为指定的映射(或词典) `LINK_TYPE_P<N>` 其中：<N> 由接口的HCA端口号决定。。<N> 可以通过运行来确定值 `grep PCI_SLOT_NAME /sys/class/net/<INTERFACE_NAME>/device/uevent` 并从PCI插槽名称中将1添加到最后一个数字、然后转换为十进制值。
 - a. 例如给定 `PCI_SLOT_NAME=0000:2f:00.2 (2 + 1 → HCA端口3)→ LINK_TYPE_P3: eth:`

```
eseries_roce_interfaces:
- name: <INTERFACE>
  address: <IP/SUBNET>
  mstconfig:
    LINK_TYPE_P3: eth
```

有关更多详细信息、请参见 ["NetApp E系列主机集合的文档"](#) 所使用的接口类型/协议。

配置单个块节点

使用主机变量(`host_vars`)指定各个块节点的配置。

概述

本节将逐步介绍填充 `host_vars/<BLOCK_NODE_HOSTNAME>.yaml` 集群中每个块节点的文件。这些文件只能包含特定块节点特有的配置。这通常包括：

- 系统名称(如System Manager中所示)。
- 其中一个控制器的HTTPS URL (用于使用其REST API管理系统)。
- 用于连接到此块节点的存储协议文件节点。
- 配置主机接口卡(HIC)端口、例如IP地址(如果需要)。

步骤

请参考["规划文件系统"](#)本节中定义的IP寻址方案、为集群中的每个块节点创建一个文件 ``host_vars/<BLOCK_NODE_HOSTNAME>.yaml`` 并按如下所示进行填充：

1. 在顶部指定其中一个控制器的系统名称和HTTPS URL：


```
eseries_system_name: <SYSTEM_NAME>
eseries_system_api_url:
https://<MANAGEMENT_HOSTNAME_OR_IP>:8443/devmgr/v2/
```

2. 选择 "协议" 文件节点将用于连接到此块节点:

- a. 支持的协议: auto, iscsi, fc, sas, ib_srp, ib_iser, nvme_ib, nvme_fc, nvme_roce。

```
eseries_initiator_protocol: <PROTOCOL>
```

3. 根据所使用的协议、HIC端口可能需要额外配置。如果需要、应定义HIC端口配置、以便每个控制器配置中的顶部条目与每个控制器上最左侧的物理端口相对应、底部端口对应最右侧的端口。所有端口都需要有效配置、即使它们当前未在使用中也是如此。



如果要将HDR (200 GB) InfiniBand或200 GB RoCE与EF600块节点结合使用、另请参见以下部分。

- a. 对于iSCSI:

```

eseries_controller_iscsi_port:
  controller_a:      # Ordered list of controller A channel
definition.
    - state:          # Whether the port should be enabled.
Choices: enabled, disabled
    config_method:    # Port configuration method Choices: static,
dhcp
    address:          # Port IPv4 address
    gateway:          # Port IPv4 gateway
    subnet_mask:      # Port IPv4 subnet_mask
    mtu:              # Port IPv4 mtu
    - (...)           # Additional ports as needed.
  controller_b:      # Ordered list of controller B channel
definition.
    - (...)           # Same as controller A but for controller B

# Alternatively the following common port configuration can be
defined for all ports and omitted above:
eseries_controller_iscsi_port_state: enabled      # Generally
specifies whether a controller port definition should be applied
Choices: enabled, disabled
eseries_controller_iscsi_port_config_method: dhcp # General port
configuration method definition for both controllers. Choices:
static, dhcp
eseries_controller_iscsi_port_gateway:             # General port
IPv4 gateway for both controllers.
eseries_controller_iscsi_port_subnet_mask:         # General port
IPv4 subnet mask for both controllers.
eseries_controller_iscsi_port_mtu: 9000           # General port
maximum transfer units (MTU) for both controllers. Any value greater
than 1500 (bytes).

```

b. 对于iSER:

```

eseries_controller_ib_iser_port:
  controller_a:      # Ordered list of controller A channel address
definition.
    -                # Port IPv4 address for channel 1
    - (...)          # So on and so forth
  controller_b:      # Ordered list of controller B channel address
definition.

```

c. 对于NVMe/IB:

```

eseries_controller_nvme_ib_port:
  controller_a:      # Ordered list of controller A channel address
definition.
  -                  # Port IPv4 address for channel 1
  - (...)            # So on and so forth
  controller_b:      # Ordered list of controller B channel address
definition.

```

d. 对于NVMe/RoCE:

```

eseries_controller_nvme_roce_port:
  controller_a:      # Ordered list of controller A channel
definition.
  - state:           # Whether the port should be enabled.
  config_method:     # Port configuration method Choices: static,
dhcp
  address:           # Port IPv4 address
  subnet_mask:       # Port IPv4 subnet_mask
  gateway:           # Port IPv4 gateway
  mtu:               # Port IPv4 mtu
  speed:             # Port IPv4 speed
  controller_b:      # Ordered list of controller B channel
definition.
  - (...)            # Same as controller A but for controller B

# Alternatively the following common port configuration can be
defined for all ports and omitted above:
eseries_controller_nvme_roce_port_state: enabled          # Generally
specifies whether a controller port definition should be applied
Choices: enabled, disabled
eseries_controller_nvme_roce_port_config_method: dhcp      # General
port configuration method definition for both controllers. Choices:
static, dhcp
eseries_controller_nvme_roce_port_gateway:                 # General
port IPv4 gateway for both controllers.
eseries_controller_nvme_roce_port_subnet_mask:             # General
port IPv4 subnet mask for both controllers.
eseries_controller_nvme_roce_port_mtu: 4200                # General
port maximum transfer units (MTU). Any value greater than 1500
(bytes).
eseries_controller_nvme_roce_port_speed: auto              # General
interface speed. Value must be a supported speed or auto for
automatically negotiating the speed with the port.

```

- e. FC和SAS协议不需要额外配置。不正确建议使用SRP。

有关配置HIC端口和主机协议的其他选项、包括配置iSCSI CHAP的功能、请参见 ["文档。"](#) 包含在SANtricity 集合中。注意在部署BeeGFS时、存储池、卷配置以及配置存储的其他方面将在其他位置进行配置、不应在此文件中定义。

单击 ["此处"](#) 表示单个块节点的完整清单文件示例。

将**HDR (200 GB) InfiniBand**或**200 GB RoCE**与**NetApp EF600**块节点结合使用：

要将HDR (200 GB) InfiniBand与EF600结合使用、必须为每个物理端口配置第二个"虚拟" IP。下面是配置配备双端口InfiniBand HDR HIC的EF600的正确方法示例：

```
eseries_controller_nvme_ib_port:
  controller_a:
    - 192.168.1.101      # Port 2a (virtual)
    - 192.168.2.101      # Port 2b (virtual)
    - 192.168.1.100      # Port 2a (physical)
    - 192.168.2.100      # Port 2b (physical)
  controller_b:
    - 192.168.3.101      # Port 2a (virtual)
    - 192.168.4.101      # Port 2b (virtual)
    - 192.168.3.100      # Port 2a (physical)
    - 192.168.4.100      # Port 2b (physical)
```

指定通用文件节点配置

使用组变量(`group_vars`)指定通用文件节点配置。

概述

应连接到所有文件节点的配置在中进行定义 `group_vars/ha_cluster.yml`。通常包括：

- 有关如何连接和登录到每个文件节点的详细信息。
- 通用网络配置。
- 是否允许自动重新启动。
- 应如何配置防火墙和SELinux状态。
- 集群配置、包括警报和隔离。
- 性能调整。
- 常见BeeGFS服务配置。



此外、还可以在各个文件节点上定义此文件中设置的选项、例如、如果使用的是混合硬件型号、或者每个节点的密码不同。单个文件节点上的配置将优先于此文件中的配置。

步骤

创建文件 `group_vars/ha_cluster.yml` 并按如下所示进行填充：

1. 指示Ansible Control节点应如何向远程主机进行身份验证：

```
ansible_ssh_user: root
ansible_become_password: <PASSWORD>
```



尤其是在生产环境中、不要以纯文本格式存储密码。请改用Ansible Vault (请参见 ["使用Ansible Vault加密内容"](#))或 `--ask-become-pass` 运行攻略手册时的选项。如果 `ansible_ssh_user` 已为root用户、则可以选择省略 `ansible_become_password`。

2. 如果要在以太网或InfiniBand接口上配置静态IP (例如集群IP)、并且多个接口位于同一IP子网中(例如、ib0使用192.168.1.10/24、ib1使用192.168.1.11/24)、要使多主机支持正常工作、必须设置其他IP路由表和规则。只需按如下所示启用提供的网络接口配置挂钩：

```
eseries_ip_default_hook_templates:
- 99-multihoming.j2
```

3. 部署集群时、根据存储协议的不同、可能需要重新启动节点、以便于发现远程块设备(E系列卷)或应用配置的其他方面。默认情况下、节点将在重新启动之前进行提示、但您可以通过指定以下内容来允许节点自动重新启动：

```
eseries_common_allow_host_reboot: true
```

- a. 默认情况下、重新启动后、为确保块设备和其他服务准备就绪、Ansible将等待系统完成 `default.target` 将在继续部署之前访问。在某些使用NVMe/IB的情况下、此过程可能不够长、无法初始化、发现和连接到远程设备。这可能会导致自动部署过早继续并失败。为避免在使用NVMe/IB时出现这种情况、还应定义以下内容：

```
eseries_common_reboot_test_command: "! systemctl status
eseries_nvme_ib.service || systemctl --state=exited | grep
eseries_nvme_ib.service"
```

4. 要使BeeGFS和HA集群服务进行通信、需要使用多个防火墙端口。除非您要手动配置固件(不建议)、否则请指定以下内容以创建所需的防火墙区域并自动打开端口：

```
beegfs_ha_firewall_configure: True
```

5. 目前不支持SELinux、建议将状态设置为disabled以避免冲突(尤其是在使用RDMA时)。设置以下内容以确保SELinux已禁用：

```
eseries_beeufs_ha_disable_selinux: True
eseries_selinux_state: disabled
```

6. 配置身份验证以使文件节点能够进行通信、并根据组织策略根据需要调整默认值:

```
beegfs_ha_cluster_name: hacluster # BeeGFS HA cluster
name.
beegfs_ha_cluster_username: hacluster # BeeGFS HA cluster
username.
beegfs_ha_cluster_password: hapassword # BeeGFS HA cluster
username's password.
beegfs_ha_cluster_password_sha512_salt: randomSalt # BeeGFS HA cluster
username's password salt.
```

7. 根据"规划文件系统"指定此文件系统的BeeGFS管理IP部分:

```
beegfs_ha_mgmtd_floating_ip: <IP ADDRESS>
```



虽然`冗余、但当您将BeeGFS文件系统扩展到单个HA集群之外时、`beegfs_ha_mgmtd_floating_ip非常重要。部署后续HA集群时无需额外的BeeGFS管理服务、并指向第一个集群提供的管理服务。

8. 根据需要启用电子邮件警报:

```
beegfs_ha_enable_alerts: True
# E-mail recipient list for notifications when BeeGFS HA resources
change or fail.
beegfs_ha_alert_email_list: ["<EMAIL>"]
# This dictionary is used to configure postfix service
(/etc/postfix/main.cf) which is required to set email alerts.
beegfs_ha_alert_conf_ha_group_options:
    # This parameter specifies the local internet domain name. This is
    optional when the cluster nodes have fully qualified hostnames (i.e.
    host.example.com)
    mydomain: <MY_DOMAIN>
beegfs_ha_alert_verbosity: 3
# 1) high-level node activity
# 3) high-level node activity + fencing action information + resources
(filter on X-monitor)
# 5) high-level node activity + fencing action information + resources
```

9. 强烈建议启用隔离、否则、在主节点发生故障时、可能会阻止服务在二级节点上启动。

- a. 通过指定以下内容全局启用隔离：

```
beegfs_ha_cluster_crm_config_options:
  stonith-enabled: True
```

- i. 注意如果需要、也可以在此处指定任何受支持的 **"集群属性"**。通常不需要调整这些值，因为BeeGFS HA角色附带了许多经过充分测试的 **"默认值"**。

- b. 接下来、选择并配置隔离代理：

- i. 选项1：要使用APC配电单元(PDU)启用隔离、请执行以下操作：

```
beegfs_ha_fencing_agents:
  fence_apc:
    - ipaddr: <PDU_IP_ADDRESS>
      login: <PDU_USERNAME>
      passwd: <PDU_PASSWORD>
      pcmk_host_map:
        "<HOSTNAME>:<PDU_PORT>,<PDU_PORT>;<HOSTNAME>:<PDU_PORT>,<PDU_PORT>"
        "
```

- ii. 选项2：要使用联想XCC (及其他BMC)提供的Redfish API启用隔离：

```
redfish: &redfish
  username: <BMC_USERNAME>
  password: <BMC_PASSWORD>
  ssl_insecure: 1 # If a valid SSL certificate is not available
                  specify "1".

beegfs_ha_fencing_agents:
  fence_redfish:
    - pcmk_host_list: <HOSTNAME>
      ip: <BMC_IP>
      <<: *redfish
    - pcmk_host_list: <HOSTNAME>
      ip: <BMC_IP>
      <<: *redfish
```

- iii. 有关配置其他隔离代理的详细信息，请参见 **"Red Hat 文档"**。

10. BeeGFS HA角色可以应用多种不同的调整参数来帮助进一步优化性能。其中包括优化内核内存利用率和块设备I/O等参数。根据对NetApp E-Series块节点的测试、此角色会附带一组合理的 **"默认值"**、但默认情况下、除非您指定以下内容、否则不会应用这些功能：

```
beegfs_ha_enable_performance_tuning: True
```

- a. 如果需要、还可以在此处指定对默认性能调整所做的任何更改。有关更多详细信息、请参见完整 ["性能调整参数"](#) 文档。

11. 为了确保用于BeeGFS服务的浮动IP地址(有时称为逻辑接口)可以在文件节点之间进行故障转移、所有网络接口的名称必须一致。默认情况下、网络接口名称由内核生成、即使在安装了相同PCIe插槽中的网络适配器的相同服务器型号上、也不能保证生成一致的名称。在部署设备之前创建清单并知道生成的接口名称时、这一点也很有用。以确保设备名称一致、具体取决于服务器或的方框图 `lshw -class network -businfo` 输出中、按如下所示指定所需的PCIe地址到逻辑接口映射：

- a. 对于InfiniBand (IPoIB)网络接口：

```
eseries_ipoib_udev_rules:  
  "<PCIe ADDRESS>": <NAME> # Ex: 0000:01:00.0: i1a
```

- b. 对于以太网网络接口：

```
eseries_ip_udev_rules:  
  "<PCIe ADDRESS>": <NAME> # Ex: 0000:01:00.0: e1a
```



为避免重命名接口时发生冲突(防止重命名接口)、不应使用任何可能的默认名称、例如eth0、ens9f0、ib0或ibs4f0。一种常见的命名约定是、对以太网或InfiniBand使用"e"或"i"、后跟PCIe插槽编号和一个字母以指示端口。例如、插槽3中安装的InfiniBand适配器的第二个端口为：i3b。



如果您使用的是经验证的文件节点型号、请单击 ["此处"](#) PCIe地址到逻辑端口映射示例。

12. (可选)指定应应用于集群中所有BeeGFS服务的配置。可以找到默认配置值 ["此处"](#)，并在其他位置指定每项服务配置：

- a. BeeGFS管理服务：

```
beegfs_ha_beegfs_mgmtd_conf_ha_group_options:  
  <OPTION>: <VALUE>
```

- b. BeeGFS元数据服务：

```
beegfs_ha_beegfs_meta_conf_ha_group_options:  
  <OPTION>: <VALUE>
```

- c. BeeGFS存储服务：


```
beegfs_ha_beegfs_storage_conf_ha_group_options:  
  <OPTION>: <VALUE>
```

13. 截至BeeGFS 7.2.7和7.3.1 "连接身份验证" 必须配置或显式禁用。使用基于Ansible的部署可以通过以下几种方式进行配置：

- a. 默认情况下、部署将自动配置连接身份验证并生成 `connauthfile` 该文件将分发到所有文件节点、并与BeeGFS服务结合使用。此文件还将放置/维护在位于的Ansible控制节点上
`<INVENTORY>/files/beegfs/<sysMgmdHost>_connAuthFile` 应(安全)维护的位置、以便在需要访问此文件系统的客户端中重复使用。
 - i. 要生成新密钥、请指定 `-e "beegfs_ha_conn_auth_force_new=True` 运行Ansible攻略手册时。请注意、如果出现、则会忽略此问题 `beegfs_ha_conn_auth_secret` 已定义。
 - ii. 有关高级选项，请参阅附带的完整默认值列表 "[BeeGFS HA角色](#)"。
- b. 可以通过在中定义以下内容来使用自定义密钥 `ha_cluster.yml`：

```
beegfs_ha_conn_auth_secret: <SECRET>
```

- c. 可以完全禁用连接身份验证(不建议使用)：

```
beegfs_ha_conn_auth_enabled: false
```

单击 "[此处](#)" 有关表示通用文件节点配置的完整清单文件的示例。

将**HDR (200 GB) InfiniBand**与**NetApp EF600**块节点结合使用：

要将HDR (200 GB) InfiniBand与EF600结合使用、子网管理器必须支持虚拟化。如果使用交换机连接文件和块节点、则需要在子网管理器管理器上为整个网络结构启用此功能。

如果块节点和文件节点使用InfiniBand直接连接、`opensm`则必须在每个文件节点上为直接连接到块节点的每个接口配置一个实例。通过指定 `configure: true` 时间来完成此"[配置文件节点存储接口](#)"操作。

目前、受支持的Linux分发版附带的收件箱版本 `opensm` 不支持虚拟化。而是需要从NVIDIA OpenFabFabric企业分发版(OFED)安装和配置版本 `opensm`。尽管仍支持使用Ansible进行部署、但还需要执行一些额外步骤：

1. 使用cURL或所需工具、将NVIDIA网站上部分中列出的OpenSM版本的软件包下载到目录中 "[技术要求](#)"
`<INVENTORY>/packages/`。例如：

```
curl -o packages/opensm-5.17.2.MLNX20240610.dc7c2998-  
0.1.2310322.x86_64.rpm  
https://linux.mellanox.com/public/repo/mlnx_ofed/23.10-  
3.2.2.0/rhel9.4/x86_64/opensm-5.17.2.MLNX20240610.dc7c2998-  
0.1.2310322.x86_64.rpm  
curl -o packages/opensm-libs-5.17.2.MLNX20240610.dc7c2998-  
0.1.2310322.x86_64.rpm  
https://linux.mellanox.com/public/repo/mlnx_ofed/23.10-  
3.2.2.0/rhel9.4/x86_64/opensm-libs-5.17.2.MLNX20240610.dc7c2998-  
0.1.2310322.x86_64.rpm
```

2. 下 group_vars/ha_cluster.yml 定义以下配置:

```

### OpenSM package and configuration information
eseries_ib_opensm_allow_upgrades: true
eseries_ib_opensm_skip_package_validation: true
eseries_ib_opensm_rhel_packages: []
eseries_ib_opensm_custom_packages:
  install:
    - files:
        add:
          "packages/opensm-5.17.2.MLNX20240610.dc7c2998-
0.1.2310322.x86_64.rpm": "/tmp/"
          "packages/opensm-libs-5.17.2.MLNX20240610.dc7c2998-
0.1.2310322.x86_64.rpm": "/tmp/"
    - packages:
        add:
          - /tmp/opensm-5.17.2.MLNX20240610.dc7c2998-
0.1.2310322.x86_64.rpm
          - /tmp/opensm-libs-5.17.2.MLNX20240610.dc7c2998-
0.1.2310322.x86_64.rpm
  uninstall:
    - packages:
        remove:
          - opensm
          - opensm-libs
    files:
        remove:
          - /tmp/opensm-5.17.2.MLNX20240610.dc7c2998-
0.1.2310322.x86_64.rpm
          - /tmp/opensm-libs-5.17.2.MLNX20240610.dc7c2998-
0.1.2310322.x86_64.rpm

eseries_ib_opensm_options:
  virt_enabled: "2"

```

指定通用块节点配置

使用组变量(group_vars)指定通用块节点配置。

概述

应通过定义适用于所有块节点的配置 group_vars/eseries_storage_systems.yml。通常包括：

- 有关Ansible控制节点应如何连接到用作块节点的E系列存储系统的详细信息。
- 节点应运行的固件、NVSRAM和驱动器固件版本。
- 全局配置、包括缓存设置、主机配置以及应如何配置卷的设置。



此外、还可以在各个块节点上定义此文件中设置的选项、例如、如果使用的是混合硬件型号、或者每个节点的密码不同。单个块节点上的配置将优先于此文件中的配置。

步骤

创建文件 `group_vars/eseries_storage_systems.yml` 并按如下所示进行填充：

1. Ansible不使用SSH连接到块节点、而是使用REST API。为此、我们必须设置：

```
ansible_connection: local
```

2. 指定用于管理每个节点的用户名和密码。用户名可以选择省略(并默认为admin)、否则您可以指定具有管理员权限的任何帐户。此外、还应指定是验证SSL证书还是忽略SSL证书：

```
eseries_system_username: admin
eseries_system_password: <PASSWORD>
eseries_validate_certs: false
```



不建议以纯文本格式列出任何密码。使用Ansible存储或提供 `eseries_system_password` 使用-extra vars运行Ansible时。

3. 也可以指定应在节点上安装的控制器固件、NVSRAM和驱动器固件。需要将这些文件下载到 `packages/` 目录、然后运行Ansible。可以下载E系列控制器固件和NVSRAM ["此处"](#) 和驱动器固件 ["此处"](#)：

```
eseries_firmware_firmware: "packages/<FILENAME>.dlp" # Ex.
"packages/RCB_11.80GA_6000_64cc0ee3.dlp"
eseries_firmware_nvram: "packages/<FILENAME>.dlp" # Ex.
"packages/N6000-880834-D08.dlp"
eseries_drive_firmware_firmware_list:
  - "packages/<FILENAME>.dlp"
  # Additional firmware versions as needed.
eseries_drive_firmware_upgrade_drives_online: true # Recommended unless
BeeGFS hasn't been deployed yet, as it will disrupt host access if set
to "false".
```



如果指定了此配置、Ansible将自动更新所有固件、包括重新启动控制器(如有必要)、而不会出现其他提示。这对BeeGFS/主机I/O预期不会造成中断、但发生原因 性能可能会暂时下降。

4. 调整全局系统配置默认值。此处列出的选项和值通常建议用于NetApp上的BeeGFS、但可以根据需要进行调整：

```

eseries_system_cache_block_size: 32768
eseries_system_cache_flush_threshold: 80
eseries_system_default_host_type: linux dm-mp
eseries_system_autoload_balance: disabled
eseries_system_host_connectivity_reporting: disabled
eseries_system_controller_shelf_id: 99 # Required by default.

```

5. 配置全局卷配置默认值。此处列出的选项和值通常建议用于NetApp上的BeeGFS、但可以根据需要进行调整：

```

eseries_volume_size_unit: pct # Required by default. This allows volume
capacities to be specified as a percentage, simplifying putting together
the inventory.
eseries_volume_read_cache_enable: true
eseries_volume_read_ahead_enable: false
eseries_volume_write_cache_enable: true
eseries_volume_write_cache_mirror_enable: true
eseries_volume_cache_without_batteries: false

```

6. 如果需要、请根据以下最佳实践调整Ansible为存储池和卷组选择驱动器的顺序：
- 列出应首先用于管理和/或元数据卷、最后用于存储卷的任何(可能较小)驱动器。
 - 确保根据磁盘架/驱动器机箱型号在可用驱动器通道之间平衡驱动器选择顺序。例如、如果使用EF600且不进行扩展、则驱动器0-11位于驱动器通道1上、驱动器12-23位于驱动器通道上。因此、要选择一种平衡驱动器选择的策略 `disk shelf:drive 99: 0、99: 23、99: 1、99: 22`等如果存在多个机箱、则第一个数字表示驱动器架ID。

```

# Optimal/recommended order for the EF600 (no expansion):
eseries_storage_pool_usable_drives:
"99:0,99:23,99:1,99:22,99:2,99:21,99:3,99:20,99:4,99:19,99:5,99:18,99
:6,99:17,99:7,99:16,99:8,99:15,99:9,99:14,99:10,99:13,99:11,99:12"

```

单击 ["此处"](#) 有关表示通用块节点配置的完整清单文件的示例。

定义BeeGFS服务

定义BeeGFS管理服务

BeeGFS服务使用组变量(`group_vars`)进行配置。

概述

本节将介绍如何定义BeeGFS管理服务。对于特定文件系统、HA集群中只应存在一个此类型的服务。配置此服

务包括定义：

- 服务类型(管理)。
- 定义应仅适用于此BeeGFS服务的任何配置。
- 配置可访问此服务的一个或多个浮动IP (逻辑接口)。
- 指定卷存储此服务数据的位置/方式(BeeGFS管理目标)。

步骤

创建一个新文件 `group\_vars/mgmt.yml` 并引用该["规划文件系统"](#)部分、按如下所示填充该文件：

1. 指示此文件表示BeeGFS管理服务的配置：

```
beegfs_service: management
```

2. 定义应仅应用于此BeeGFS服务的任何配置。管理服务通常不需要执行此操作、除非您需要启用配额、无论您的任何受支持的配置参数如何 `beegfs-mgmtd.conf` 可以包括在内。请注意、以下参数是自动/在其他位置配置的、不应在此处指定：`storeMgmtdDirectory`，`connAuthFile`，`connDisableAuthentication`，`connInterfacesFile`，和 `connNetFilterFile`。

```
beegfs_ha_beegfs_mgmtd_conf_resource_group_options:  
  <beegfs-mgmt.conf:key>:<beegfs-mgmt.conf:value>
```

3. 配置一个或多个浮动IP、供其他服务和客户端连接到此服务时使用(此操作将自动设置BeeGFS `connInterfacesFile` 选项)：

```
floating_ips:  
  - <INTERFACE>:<IP/SUBNET> # Primary interface. Ex.  
  i1b:100.127.101.0/16  
  - <INTERFACE>:<IP/SUBNET> # Secondary interface(s) as needed.
```

4. 或者、指定一个或多个允许用于传出通信的IP子网(此操作将自动设置BeeGFS `connNetFilterFile` 选项)：

```
filter_ip_ranges:  
  - <SUBNET>/<MASK> # Ex. 192.168.10.0/24
```

5. 根据以下准则指定此服务将在其中存储数据的BeeGFS管理目标：
 - a. 同一存储池或卷组名称可用于多个BeeGFS服务/目标、只需确保使用相同的即可 `name`，`raid_level`，`criteria_*`，和 `common_*` 每个服务的配置(为每个服务列出的卷应不同)。
 - b. 卷大小应指定为存储池/卷组的百分比、并且使用特定存储池/卷组的所有服务/卷的总数不应超过100。注意使用SSD时、建议在卷组中保留一些可用空间、以最大程度地提高SSD性能和使用寿命(单击可["此处"](#)了解更多详细信息)。

- c. 单击 ["此处"](#) 以获取可用于的配置选项的完整列表 `eseries_storage_pool_configuration`。请注意一些选项、例如 `state`, `host`, `host_type`, `workload_name`, 和 `workload_metadata` 和卷名称将自动生成、不应在此指定。

```
beegfs_targets:
  <BLOCK_NODE>: # The name of the block node as found in the Ansible
inventory. Ex: netapp_01
  eseries_storage_pool_configuration:
    - name: <NAME> # Ex: beegfs_m1_m2_m5_m6
      raid_level: <LEVEL> # One of: raid1, raid5, raid6, raidDiskPool
      criteria_drive_count: <DRIVE COUNT> # Ex. 4
      common_volume_configuration:
        segment_size_kb: <SEGMENT SIZE> # Ex. 128
      volumes:
        - size: <PERCENT> # Percent of the pool or volume group to
allocate to this volume. Ex. 1
          owning_controller: <CONTROLLER> # One of: A, B
```

单击 ["此处"](#) 有关表示BeeGFS管理服务的完整清单文件的示例。

定义BeeGFS元数据服务

BeeGFS服务使用组变量(`group_vars`)进行配置。

概述

本节将介绍如何定义BeeGFS元数据服务。对于特定文件系统、HA集群中应至少存在一个此类型的服务。配置此服务包括定义：

- 服务类型(元数据)。
- 定义应仅适用于此BeeGFS服务的任何配置。
- 配置可访问此服务的一个或多个浮动IP (逻辑接口)。
- 指定卷存储此服务数据的位置/方式(BeeGFS元数据目标)。

步骤

参考["规划文件系统"](#)部分、``group_vars/meta_<ID>.yaml``为集群中的每个元数据服务创建一个位于的文件、然后按如下所示填充它们：

1. 指示此文件表示BeeGFS元数据服务的配置：

```
beegfs_service: metadata
```

2. 定义应仅应用于此BeeGFS服务的任何配置。您必须至少指定所需的TCP和UDP端口、无论这些端口来自哪个受支持的配置参数 `beegfs-meta.conf` 也可以包括在内。请注意、以下参数是自动/在其他位置配置

的、不应在此处指定：sysMgmtHost, storeMetaDirectory, connAuthFile, connDisableAuthentication, connInterfacesFile, 和 connNetFilterFile。

```
beegfs_ha_beegfs_meta_conf_resource_group_options:
  connMetaPortTCP: <TCP PORT>
  connMetaPortUDP: <UDP PORT>
  tuneBindToNumaZone: <NUMA ZONE> # Recommended if using file nodes with
multiple CPU sockets.
```

3. 配置一个或多个浮动IP、供其他服务和客户端连接到此服务时使用(此操作将自动设置BeeGFS connInterfacesFile 选项):

```
floating_ips:
  - <INTERFACE>:<IP/SUBNET> # Primary interface. Ex.
  ilb:100.127.101.1/16
  - <INTERFACE>:<IP/SUBNET> # Secondary interface(s) as needed.
```

4. 或者、指定一个或多个允许用于传出通信的IP子网(此操作将自动设置BeeGFS connNetFilterFile 选项):

```
filter_ip_ranges:
  - <SUBNET>/<MASK> # Ex. 192.168.10.0/24
```

5. 根据以下准则指定此服务将在其中存储数据的BeeGFS元数据目标(此操作也会自动配置) storeMetaDirectory 选项):

- a. 同一存储池或卷组名称可用于多个BeeGFS服务/目标、只需确保使用相同的即可 name, raid_level, criteria_*, 和 common_* 每个服务的配置(为每个服务列出的卷应不同)。
- b. 卷大小应指定为存储池/卷组的百分比、并且使用特定存储池/卷组的所有服务/卷的总数不应超过100。注意使用SSD时、建议在卷组中保留一些可用空间、以最大程度地提高SSD性能和使用寿命(单击可["此处"](#)了解更多详细信息)。
- c. 单击 ["此处"](#) 以获取可用于的配置选项的完整列表 eseries_storage_pool_configuration。请注意一些选项、例如 state, host, host_type, workload_name, 和 workload_metadata 和卷名称将自动生成、不应在此指定。


```

beegfs_targets:
  <BLOCK_NODE>: # The name of the block node as found in the Ansible
inventory. Ex: netapp_01
  eseries_storage_pool_configuration:
    - name: <NAME> # Ex: beegfs_m1_m2_m5_m6
      raid_level: <LEVEL> # One of: raid1, raid5, raid6, raidDiskPool
      criteria_drive_count: <DRIVE COUNT> # Ex. 4
      common_volume_configuration:
        segment_size_kb: <SEGMENT SIZE> # Ex. 128
      volumes:
        - size: <PERCENT> # Percent of the pool or volume group to
allocate to this volume. Ex. 1
          owning_controller: <CONTROLLER> # One of: A, B

```

单击 ["此处"](#) 有关表示BeeGFS元数据服务的完整清单文件的示例。

定义BeeGFS存储服务

BeeGFS服务使用组变量(group_vars)进行配置。

概述

本节将介绍如何定义BeeGFS存储服务。对于特定文件系统、HA集群中应至少存在一个此类型的服务。配置此服务包括定义：

- 服务类型(存储)。
- 定义应仅适用于此BeeGFS服务的任何配置。
- 配置可访问此服务的一个或多个浮动IP (逻辑接口)。
- 指定卷存储此服务数据的位置/方式(BeeGFS存储目标)。

步骤

参考["规划文件系统"](#)部分、`group\_vars/stor\_<ID>.yaml`为集群中的每个存储服务创建一个文件、然后按如下所示填充它们：

1. 指示此文件表示BeeGFS存储服务的配置：

```
beegfs_service: storage
```

2. 定义应仅应用于此BeeGFS服务的任何配置。您必须至少指定所需的TCP和UDP端口、无论这些端口来自哪个受支持的配置参数 `beegfs-storage.conf` 也可以包括在内。请注意、以下参数是自动/在其他位置配置的、不应在此处指定：`sysMgmdHost`，`storeStorageDirectory`，`connAuthFile`，`connDisableAuthentication`，`connInterfacesFile`，和 `connNetFilterFile`。

```
beegfs_ha_beegfs_storage_conf_resource_group_options:
  connStoragePortTCP: <TCP PORT>
  connStoragePortUDP: <UDP PORT>
  tuneBindToNumaZone: <NUMA ZONE> # Recommended if using file nodes with
multiple CPU sockets.
```

3. 配置一个或多个浮动IP、供其他服务和客户端连接到此服务时使用(此操作将自动设置BeeGFS connInterfacesFile 选项):

```
floating_ips:
  - <INTERFACE>:<IP/SUBNET> # Primary interface. Ex.
i1b:100.127.101.1/16
  - <INTERFACE>:<IP/SUBNET> # Secondary interface(s) as needed.
```

4. 或者、指定一个或多个允许用于传出通信的IP子网(此操作将自动设置BeeGFS connNetFilterFile 选项):

```
filter_ip_ranges:
  - <SUBNET>/<MASK> # Ex. 192.168.10.0/24
```

5. 根据以下准则指定此服务要存储数据的BeeGFS存储目标(此操作也会自动配置) storeStorageDirectory 选项):
 - a. 同一存储池或卷组名称可用于多个BeeGFS服务/目标、只需确保使用相同的即可 name, raid_level, criteria_*, 和 common_* 每个服务的配置(为每个服务列出的卷应不同)。
 - b. 卷大小应指定为存储池/卷组的百分比、并且使用特定存储池/卷组的所有服务/卷的总数不应超过100。注意使用SSD时、建议在卷组中保留一些可用空间、以最大程度地提高SSD性能和使用寿命(单击可["此处"](#)了解更多详细信息)。
 - c. 单击 ["此处"](#) 以获取可用于的配置选项的完整列表 eseries_storage_pool_configuration。请注意一些选项、例如 state, host, host_type, workload_name, 和 workload_metadata 和卷名称将自动生成、不应在此指定。

```

beegfs_targets:
  <BLOCK_NODE>: # The name of the block node as found in the Ansible
inventory. Ex: netapp_01
  eseries_storage_pool_configuration:
    - name: <NAME> # Ex: beegfs_s1_s2
      raid_level: <LEVEL> # One of: raid1, raid5, raid6,
raidDiskPool
      criteria_drive_count: <DRIVE COUNT> # Ex. 4
      common_volume_configuration:
        segment_size_kb: <SEGMENT SIZE> # Ex. 128
      volumes:
        - size: <PERCENT> # Percent of the pool or volume group to
allocate to this volume. Ex. 1
          owning_controller: <CONTROLLER> # One of: A, B
        # Multiple storage targets are supported / typical:
        - size: <PERCENT> # Percent of the pool or volume group to
allocate to this volume. Ex. 1
          owning_controller: <CONTROLLER> # One of: A, B

```

单击 ["此处"](#) 有关表示BeeGFS存储服务的完整清单文件的示例。

将BeeGFS服务映射到文件节点

使用指定可以运行每个BeeGFS服务的文件节点 `inventory.yml` 文件

概述

本节将介绍如何创建 `inventory.yml` 文件其中包括列出所有块节点并指定可以运行每个BeeGFS服务的文件节点。

步骤

创建文件 `inventory.yml` 并按如下所示进行填充：

1. 从文件顶部、创建标准Ansible清单结构：

```

# BeeGFS HA (High_Availability) cluster inventory.
all:
  children:

```

2. 创建一个包含参与此HA集群的所有块节点的组：

```
# Ansible group representing all block nodes:
eseries_storage_systems:
  hosts:
    <BLOCK NODE HOSTNAME>:
    <BLOCK NODE HOSTNAME>:
    # Additional block nodes as needed.
```

3. 创建一个组、其中包含集群中的所有BeeGFS服务以及要运行这些服务的文件节点:

```
# Ansible group representing all file nodes:
ha_cluster:
  children:
```

4. 对于集群中的每个BeeGFS服务、定义应运行该服务的首选文件节点和任何二级文件节点:

```
<SERVICE>: # Ex. "mgmt", "meta_01", or "stor_01".
  hosts:
    <FILE NODE HOSTNAME>:
    <FILE NODE HOSTNAME>:
    # Additional file nodes as needed.
```

单击 ["此处"](#) 有关完整清单文件的示例。

版权信息

版权所有 © 2026 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。