Revisar el diseño de la solución

BeeGFS on NetApp with E-Series Storage

NetApp
March 21, 2024

Tabla de contenidos

Revisar el diseño de la solución

Descripción general del diseño

Se requieren equipos, cables y configuraciones específicos para admitir BeeGFS en la solución de NetApp, que combina el sistema de archivos en paralelo BeeGFS con los sistemas de almacenamiento EF600 de NetApp.

Obtenga más información:

- "Configuración de hardware"
- "Configuración de software"
- "Verificación del diseño"
- "Directrices de tamaño"
- "Ajuste del rendimiento"

Arquitecturas derivadas con variaciones en el diseño y el rendimiento:

- "Elementos básicos de alta capacidad"

Configuración de hardware

La configuración de hardware para BeeGFS en NetApp incluye nodos de archivo y cableado de red.

Configuración de nodos de archivos

Los nodos de archivos tienen dos sockets de CPU configurados como zonas NUMA independientes, que incluyen acceso local a un mismo número de ranuras PCIe y memoria.

Los adaptadores InfiniBand deben llenarse en las ranuras o elevadores PCI adecuados, por lo que la carga de trabajo se equilibrará entre los canales PCIe y los canales de memoria disponibles. Equilibre la carga de trabajo aislando completamente el trabajo de los servicios BeeGFS individuales a un nodo NUMA en particular. El objetivo es lograr un rendimiento similar en cada nodo de archivo como si se tratara de dos servidores de un único socket independientes.

En la figura siguiente se muestra la configuración NUMA del nodo de archivo.

[]

Los procesos BeeGFS se anclan a una zona NUMA en particular para garantizar que las interfaces utilizadas se encuentren en la misma zona. Esta configuración evita la necesidad de acceso remoto a través de la conexión entre sockets. La conexión entre zócalos se conoce a veces como el enlace QPI o GMI2; incluso en arquitecturas de procesador modernas, pueden crear un cuello de botella al utilizar redes de alta velocidad como HDR InfiniBand.

Configuración del cableado de red

Dentro de un elemento básico, cada nodo de archivo está conectado a dos nodos de bloques mediante un

total de cuatro conexiones InfiniBand redundantes. Además, cada nodo de archivo tiene cuatro conexiones redundantes a la red de almacenamiento de InfiniBand.

En la siguiente figura, observe que:

- Todos los puertos de nodos de archivos delineados en verde se utilizan para conectarse al entramado de almacenamiento; todos los demás puertos de nodos de archivos son los puertos de nodos de bloques.
- Dos puertos InfiniBand en una zona NUMA específica se conectan a las controladoras A y B del mismo nodo de bloque.
- Los puertos del nodo NUMA 0 siempre se conectan al primer nodo de bloque.
- Los puertos del nodo NUMA 1 se conectan al segundo nodo de bloque.

[]

> Para las redes de almacenamiento con switches redundantes, los puertos descritos en verde claro deben conectarse a un switch y los puertos en verde oscuro a otro switch.

La configuración del cableado que se muestra en la figura permite a cada servicio BeeGFS:

- Se ejecuta en la misma zona NUMA independientemente del nodo de archivo que esté ejecutando el servicio BeeGFS.
- Tener rutas secundarias óptimas a la red de almacenamiento front-end y a los nodos de bloques de back-end, independientemente de dónde se produzca un fallo.
- Minimice los efectos en el rendimiento si un nodo de archivo o una controladora de un nodo de bloque requiere mantenimiento.

Cableado para aprovechar el ancho de banda

Para aprovechar todo el ancho de banda bidireccional de PCIe, asegúrese de que un puerto de cada adaptador InfiniBand se conecta a la estructura de almacenamiento y el otro puerto se conecta a un nodo de bloque. La velocidad máxima teórica de un puerto HDR InfiniBand es de 25 Gbps (sin tener en cuenta la señalización y otros gastos generales). El ancho de banda máximo de dirección única de una ranura PCIe 4.0 x16 es de 32 GB/s, lo que crea un posible cuello de botella al implementar nodos de archivos que incorporen adaptadores InfiniBand de puerto doble que en teoría puedan manejar 50 GB/s de ancho de banda.

La siguiente figura muestra el diseño del cableado utilizado para aprovechar todo el ancho de banda bidireccional de PCIe.

[]

Para cada servicio BeeGFS, utilice el mismo adaptador para conectar el puerto preferido utilizado para el tráfico de cliente con la ruta al controlador de nodos de bloque que es el propietario principal de dichos volúmenes de servicios. Para obtener más información, consulte "Configuración de software".

Configuración de software

La configuración de software de BeeGFS en NetApp incluye componentes de red BeeGFS, nodos de bloque EF600, nodos de archivos BeeGFS, grupos de recursos y servicios BeeGFS.

Configuración de red BeeGFS

La configuración de red BeeGFS consta de los siguientes componentes.

- **IP flotantes** las IP flotantes son un tipo de dirección IP virtual que se puede enrutar dinámicamente a cualquier servidor de la misma red. Varios servidores pueden tener la misma dirección IP flotante, pero sólo puede estar activa en un servidor en un momento dado.

  Cada servicio de servidor BeeGFS tiene su propia dirección IP que puede moverse entre nodos de archivo en función de la ubicación de ejecución del servicio de servidor BeeGFS. Esta configuración de IP flotante permite que cada servicio conmute por error de manera independiente al otro nodo de archivo. El cliente simplemente necesita conocer la dirección IP de un servicio BeeGFS concreto; no necesita saber qué nodo de archivo está ejecutando ese servicio en ese momento.

- **Configuración de hosts múltiples del servidor BeeGFS** para aumentar la densidad de la solución, cada nodo de archivo tiene varias interfaces de almacenamiento con IP configuradas en la misma subred IP.

  Es necesario configurar más para asegurarse de que esta configuración funciona de la forma esperada con el paquete de redes de Linux, ya que, de forma predeterminada, es posible responder las solicitudes a una interfaz en otra interfaz si sus IP se encuentran en la misma subred. Además de otros inconvenientes, este comportamiento predeterminado hace que sea imposible establecer o mantener correctamente las conexiones RDMA.

  La puesta en marcha basada en Ansible gestiona el apriete del comportamiento de la ruta inversa (RP) y del protocolo de resolución de direcciones (ARP), junto con la garantía de cuándo se inician y se detienen las IP flotantes; las reglas y rutas IP correspondientes se crean de forma dinámica para permitir que la configuración de red de múltiples hosts funcione correctamente.

- **Configuración multi-rail de cliente BeeGFS** *Multi-rail* se refiere a la capacidad de una aplicación para utilizar múltiples “rieles” de red independientes para aumentar el rendimiento.

  Si bien BeeGFS puede utilizar RDMA para la conectividad, BeeGFS utiliza IPoIB para simplificar la detección y establecimiento de conexiones RDMA. Para permitir que los clientes BeeGFS utilicen varias interfaces InfiniBand, puede configurar cada cliente con una dirección IP ubicada en una subred diferente y, a continuación, configurar las interfaces preferidas para la mitad de los servicios de servidor BeeGFS en cada subred.

  En el siguiente diagrama, las interfaces resaltadas en verde claro se encuentran en una subred IP (por ejemplo, `100.127.0.0/16`) y las interfaces de color verde oscuro se encuentran en otra subred (por ejemplo, `100.128.0.0/16`).

  En la siguiente figura se muestra el equilibrio del tráfico en varias interfaces de cliente BeeGFS.

  []

  Debido a que cada archivo de BeeGFS suele estar segado en múltiples servicios de almacenamiento, la configuración multicanal permite al cliente conseguir un mayor rendimiento del que es posible con un único puerto InfiniBand. Por ejemplo, el siguiente ejemplo de código muestra una configuración común de segmentación de archivos que permite al cliente equilibrar el tráfico entre ambas interfaces:

```
root@ictad21h01:/mnt/beegfs# beegfs-ctl --getentryinfo myfile
Entry type: file
EntryID: 11D-624759A9-65
Metadata node: meta_01_tgt_0101 [ID: 101]
Stripe pattern details:
+ Type: RAID0
+ Chunksize: 1M
+ Number of storage targets: desired: 4; actual: 4
+ Storage targets:
  + 101 @ stor_01_tgt_0101 [ID: 101]
  + 102 @ stor_01_tgt_0101 [ID: 101]
  + 201 @ stor_02_tgt_0201 [ID: 201]
  + 202 @ stor_02_tgt_0201 [ID: 201]
```

El uso de dos subredes IPoIB es una distinción lógica. Puede usar una sola subred InfiniBand física (red de almacenamiento) si lo desea.

> Se ha añadido soporte multi-rail en BeeGFS 7.3.0 para permitir el uso de múltiples interfaces IB en una única subred de IPoIB. El diseño de BeeGFS en la solución de NetApp se desarrolló antes de la disponibilidad general de BeeGFS 7.3.0, por lo que se demuestra el uso de dos subredes IP para usar dos interfaces IB en los clientes BeeGFS. Una ventaja del método de subred IP múltiple es que se elimina la necesidad de configurar el modo de multicentrado en los nodos cliente de BeeGFS (para obtener más información, consulte "Soporte para RDMA BeeGFS").

Configuración de nodos de bloques de EF600

Los nodos de bloques constan de dos controladoras RAID activo/activo con acceso compartido al mismo conjunto de unidades. Por lo general, cada controladora tiene la mitad de los volúmenes configurados en el sistema, pero puede sustituir la otra controladora según sea necesario.

El software multivía en los nodos de archivos determina la ruta activa y optimizada para cada volumen y se mueve automáticamente a la ruta alternativa en caso de que se produzca un fallo en un cable, un adaptador o una controladora.

El siguiente diagrama muestra el diseño de la controladora en los nodos de bloques de EF600.

[]

Para facilitar la solución de alta disponibilidad de disco compartido, los volúmenes se asignan a los dos nodos de archivo para que puedan hacerse cargo entre sí según sea necesario. En el siguiente diagrama se muestra un ejemplo de cómo se configura el servicio BeeGFS y la propiedad de volumen preferida para obtener el máximo rendimiento. La interfaz a la izquierda de cada servicio BeeGFS indica la interfaz preferida que los clientes y otros servicios utilizan para ponerse en contacto con él.

[]

En el ejemplo anterior, los clientes y los servicios de servidor prefieren comunicarse con el servicio de almacenamiento 1 mediante la interfaz i1b. El servicio de almacenamiento 1 utiliza la interfaz i1a como ruta preferida para comunicarse con sus volúmenes (Storage_tgt_101, 102) en la controladora A del primer nodo

de bloque. Esta configuración hace uso del ancho de banda PCIe bidireccional completo disponible para el adaptador InfiniBand y logra un mejor rendimiento a partir de un adaptador HDR InfiniBand de doble puerto del que sería posible con PCIe 4.0 de otro modo.

Configuración de nodos de archivos BeeGFS

Los nodos de archivos BeeGFS se configuran en un clúster de alta disponibilidad para facilitar la conmutación por error de los servicios BeeGFS entre varios nodos de archivos.

El diseño de clúster de alta disponibilidad se basa en dos proyectos de alta disponibilidad de Linux ampliamente utilizados: Corosync para la pertenencia a clústeres y Pacemaker para la administración de recursos de clúster. Para obtener más información, consulte "Formación de Red Hat para complementos de alta disponibilidad".

NetApp es autor y creó varios agentes de recursos de marco de clúster abierto (OCF) ampliados para permitir que el clúster inicie y supervise de forma inteligente los recursos de BeeGFS.

Clústeres de alta disponibilidad de BeeGFS

Normalmente, cuando se inicia un servicio BeeGFS (con o sin ha), deben existir algunos recursos:

- Direcciones IP donde se puede acceder al servicio, generalmente configuradas por Network Manager.
- Sistemas de archivos subyacentes utilizados como objetivos para BeeGFS para almacenar datos.

  Normalmente se definen en `/etc/fstab` Y montado por systemd.

- Servicio de sistema responsable de iniciar los procesos de BeeGFS cuando los otros recursos están listos.

  Sin software adicional, estos recursos solo comienzan en un único nodo de archivo. Por lo tanto, si el nodo de archivo se desconecta, una parte del sistema de archivos BeeGFS no está accesible.

Debido a que varios nodos pueden iniciar cada servicio BeeGFS, Pacemaker debe asegurarse de que cada servicio y los recursos dependientes sólo se ejecutan en un nodo cada vez. Por ejemplo, si dos nodos intentan iniciar el mismo servicio BeeGFS, existe el riesgo de que se dañen los datos si ambos intentan escribir en los mismos archivos en el destino subyacente. Para evitar esta situación, Pacemaker confía en Corosync para mantener de forma fiable el estado general del clúster sincronizado entre todos los nodos y establecer quórum.

Si se produce un fallo en el clúster, Pacemaker reacciona y reinicia los recursos de BeeGFS en otro nodo. En algunos casos, es posible que Pacemaker no pueda comunicarse con el nodo defectuoso original para confirmar que los recursos están detenidos. Para verificar que el nodo está inactivo antes de reiniciar los recursos de BeeGFS en otra parte, Pacemaker apaga el nodo defectuoso, lo que es ideal para eliminar la alimentación.

Hay muchos agentes de esgrima de código abierto disponibles que permiten a Pacemaker cercar un nodo con una unidad de distribución de energía (PDU) o utilizando el controlador de administración de placa base del servidor (BMC) con API como Redfish.

Cuando BeeGFS se ejecuta en un clúster ha, Pacemaker gestiona todos los servicios BeeGFS y los recursos subyacentes en grupos de recursos. Cada servicio BeeGFS y los recursos de los que depende, se configuran en un grupo de recursos, que garantiza que los recursos se inician y se detienen en el orden correcto y se encuentran en el mismo nodo.

Para cada grupo de recursos BeeGFS, Pacemaker ejecuta un recurso de supervisión BeeGFS personalizado

que es responsable de detectar condiciones de fallo y de activar de forma inteligente recuperaciones tras fallos cuando un servicio BeeGFS ya no está accesible en un nodo concreto.

La siguiente figura muestra los servicios y dependencias de BeeGFS controlados por marcapasos.

[]

De modo que se inician varios servicios BeeGFS del mismo tipo en el mismo nodo, Pacemaker se configura para iniciar servicios BeeGFS mediante el método de configuración Multi Mode. Para obtener más información, consulte "Documentación de BeeGFS sobre modo múltiple".

Debido a que los servicios BeeGFS deben poder iniciarse en varios nodos, el archivo de configuración de cada servicio (normalmente ubicado en `/etc/beegfs`) Se almacena en uno de los volúmenes E-Series utilizados como objetivo BeeGFS para ese servicio. Esto hace que la configuración junto con los datos de un servicio BeeGFS en particular sea accesible para todos los nodos que puedan necesitar ejecutar el servicio.

```
# tree stor_01_tgt_0101/ -L 2
stor_01_tgt_0101/
├── data
│   ├── benchmark
│   ├── buddymir
│   ├── chunks
│   ├── format.conf
│   ├── lock.pid
│   ├── nodeID
│   ├── nodeNumID
│   ├── originalNodeID
│   ├── targetID
│   └── targetNumID
└── storage_config
    ├── beegfs-storage.conf
    ├── connInterfacesFile.conf
    └── connNetFilterFile.conf
```

Verificación del diseño

El diseño de segunda generación de BeeGFS en la solución de NetApp se verificó mediante tres perfiles de configuración de bloques básicos.

Los perfiles de configuración incluyen lo siguiente:

- Un único elemento básico, incluidos los servicios de gestión, metadatos y almacenamiento de BeeGFS.
- Metadatos BeeGFS más un elemento básico de almacenamiento.
- Un elemento básico de sólo almacenamiento BeeGFS.

Los elementos básicos se adjuntaron a dos switches Mellanox Quantum InfiniBand (MQM8700). También se adjuntaron diez clientes BeeGFS a los switches InfiniBand y se utilizaron para ejecutar utilidades de análisis de rendimiento sintéticos.

En la siguiente figura, se muestra la configuración de BeeGFS que se utiliza para validar BeeGFS en la solución de NetApp.

[]

Segmentación de archivos BeeGFS

Una ventaja de los sistemas de archivos paralelos es la capacidad de resegmentar archivos individuales en múltiples destinos de almacenamiento, lo que podría representar volúmenes en los mismos sistemas de almacenamiento subyacentes o en diferentes.

En BeeGFS, puede configurar la segmentación por directorio y por archivo para controlar el número de destinos utilizados para cada archivo y para controlar el tamaño de bloque (o el tamaño de bloque) utilizado para cada franja de archivo. Esta configuración permite al sistema de archivos admitir distintos tipos de cargas de trabajo y perfiles de I/o sin necesidad de reconfigurar o reiniciar los servicios. Puede aplicar la configuración de franja mediante `beegfs-ctl` Herramienta de línea de comandos o con aplicaciones que usan la API de segmentación. Para obtener más información, consulte la documentación de BeeGFS para "Segmentación" y.. "API de segmentación".

Para lograr el mejor rendimiento, los patrones de franjas se ajustaron durante la prueba, y se señalan los parámetros utilizados para cada prueba.

Pruebas de ancho de banda IOR: Múltiples clientes

Las pruebas de ancho de banda IOR utilizaban OpenMPI para ejecutar trabajos paralelos de la herramienta de generador de E/S sintético IOR (disponible en "GitHub de HPC") A través de los 10 nodos de cliente a uno o más bloques de creación de BeeGFS. A menos que se indique lo contrario:

- Todas las pruebas utilizaron E/S directa con un tamaño de transferencia 1MiB.
- La segmentación de archivos BeeGFS se ha establecido en un tamaño de archivo de 1 MB y un objetivo por archivo.

Se utilizaron los siguientes parámetros para IOR con el recuento de segmentos ajustado para mantener el tamaño del archivo agregado a 5 TIB para un bloque básico y 40 TIB para tres bloques básicos.

```
mpirun --allow-run-as-root --mca btl tcp -np 48 -map-by node -hostfile
10xnodes ior -b 1024k --posix.odirect -e -t 1024k -s 54613 -z -C -F -E -k
```

Un elemento básico de BeeGFS (gestión, metadatos y almacenamiento)

En la siguiente figura, se muestran los resultados de la prueba IOR con un solo elemento básico de BeeGFS (gestión, metadatos y almacenamiento).

[]

Metadatos BeeGFS + elemento básico de almacenamiento

En la siguiente figura se muestran los resultados de la prueba IOR con un único bloque de creación de almacenamiento y metadatos BeeGFS.

[]

Elemento básico de sólo almacenamiento BeeGFS

En la siguiente figura se muestran los resultados de la prueba IOR con un solo elemento básico de almacenamiento BeeGFS.

[]

Tres elementos básicos de BeeGFS

En la siguiente figura se muestran los resultados de la prueba IOR con tres bloques de construcción BeeGFS.

[]

Según lo esperado, la diferencia de rendimiento entre el bloque básico y el bloque básico de metadatos + almacenamiento posterior es mínima. Si comparamos el elemento básico de metadatos + almacenamiento con un elemento básico exclusivo del almacenamiento, el rendimiento de lectura se aprecia un ligero aumento en el rendimiento de lectura debido a las unidades adicionales utilizadas como destino del almacenamiento. Sin embargo, no existe una diferencia significativa en el rendimiento de escritura. Para lograr un mayor rendimiento, puede añadir varios elementos básicos juntos para escalar el rendimiento de forma lineal.

Pruebas de ancho de banda IOR: Un único cliente

La prueba de ancho de banda IOR utilizó OpenMPI para ejecutar varios procesos IOR utilizando un único servidor GPU de alto rendimiento para explorar el rendimiento que se puede obtener en un único cliente.

En esta prueba también se compara el comportamiento y el rendimiento de BeeGFS cuando el cliente está configurado para utilizar la caché de páginas del kernel de Linux (`tuneFileCacheType = native`) frente al valor predeterminado `buffered` ajuste.

El modo de almacenamiento en caché nativo utiliza la memoria caché de página del kernel de Linux en el cliente, lo que permite que las operaciones de nueva lectura provengan de la memoria local en lugar de retransmitirse a través de la red.

En el siguiente diagrama se muestran los resultados de las pruebas IOR con tres bloques de creación BeeGFS y un único cliente.

[]

> La segmentación de BeeGFS para estas pruebas se estableció en un tamaño de archivo de 1 MB con ocho objetivos por archivo.

Aunque el rendimiento de escritura y lectura inicial es superior mediante el modo de búfer predeterminado, en el caso de cargas de trabajo que releer los mismos datos varias veces, se produce un aumento significativo del rendimiento en el modo de almacenamiento en caché nativo. Este rendimiento mejorado de nueva obtención es importante para cargas de trabajo como el aprendizaje profundo que relecan el mismo conjunto de datos varias veces a lo largo de muchas épocas.

Prueba de rendimiento de metadatos

En las pruebas de rendimiento de metadatos se utilizó la herramienta MDTest (incluida como parte de IOR) para medir el rendimiento de los metadatos de BeeGFS. Las pruebas utilizaron OpenMPI para ejecutar trabajos paralelos en los diez nodos cliente.

Se utilizaron los siguientes parámetros para ejecutar la prueba de referencia con el número total de procesos escalados de 10 a 320 en el paso del doble y con un tamaño de archivo de 4k.

```
mpirun -h 10xnodes –map-by node np $processes mdtest -e 4k -w 4k -i 3 -I
16 -z 3 -b 8 -u
```

El rendimiento de los metadatos se midió primero con uno entonces dos metadatos + elementos básicos del almacenamiento, para mostrar cómo se escala el rendimiento añadiendo elementos básicos adicionales.

Un elemento básico de metadatos BeeGFS + almacenamiento

En el siguiente diagrama se muestran los resultados de MDTest con un bloque de creación de almacenamiento y metadatos BeeGFS.

[]

Dos metadatos BeeGFS + elementos básicos de almacenamiento

El siguiente diagrama muestra los resultados de MDTest con dos metadatos BeeGFS + bloques de almacenamiento.

[]

Validación funcional

Como parte de la validación de esta arquitectura, NetApp ejecutó varias pruebas funcionales incluyendo las siguientes:

- Al producirse un fallo en un puerto InfiniBand de un único cliente, se deshabilita el puerto del switch.
- Al producirse un fallo en un puerto InfiniBand de un único servidor, se deshabilita el puerto del switch.
- Activación de un apagado inmediato del servidor mediante el BMC.
- Colocación dignidad de un nodo en espera y conmutación por error al servicio en otro nodo.
- Con dignidad, volver a colocar un nodo en línea y devolver servicios al nodo original.
- Apague uno de los switches InfiniBand mediante la PDU. Todas las pruebas se realizaron mientras las pruebas de estrés estaban en curso con el `sysSessionChecksEnabled: false` Parámetro definido en los clientes BeeGFS. No se han observado errores ni interrupciones en I/O.

> Hay un problema conocido (consulte "Cambios") Cuando las conexiones RDMA cliente/servidor BeeGFS se interrumpen inesperadamente, ya sea a través de la pérdida de la interfaz primaria (como se define en `connInterfacesFile`) O un servidor BeeGFS falla; la E/S de cliente activa se puede bloquear durante un máximo de diez minutos antes de continuar. Este problema no ocurre cuando los nodos BeeGFS se colocan correctamente dentro y fuera del modo de espera para el mantenimiento planificado o si TCP está en uso.

Validación NVIDIA DGX A100 SuperPOD y BasePOD

NetApp validó una solución de almacenamiento para nVIDIAs DGX A100 SuperPOD que utiliza un sistema de archivos BeeGFS similar que consiste en tres elementos básicos con los metadatos más el perfil de configuración de almacenamiento aplicado. El esfuerzo de cualificación incluyó probar la solución descrita en este NVA con veinte servidores DGX A100 GPU que ejecutan una gran variedad de pruebas de rendimiento de almacenamiento, aprendizaje automático y aprendizaje profundo. Todo el almacenamiento certificado para su uso en DGX A100 SuperPOD de NVIDIA está certificado automáticamente para su uso en las arquitecturas NVIDIA BasePOD.

Para obtener más información, consulte "NVIDIA DGX SuperPOD con NetApp" y.. "DGX BasePOD de NVIDIA".

Directrices de tamaño

La solución BeeGFS incluye recomendaciones sobre el rendimiento y el ajuste de la capacidad basadas en pruebas de verificación.

El objetivo de una arquitectura de elementos básicos es crear una solución de tamaño sencillo mediante la adición de varios elementos básicos para satisfacer los requisitos de un sistema BeeGFS concreto. Con las siguientes pautas, puede estimar la cantidad y los tipos de bloques de construcción de BeeGFS que se necesitan para cumplir los requisitos de su entorno.

Tenga en cuenta que estas estimaciones representan el mejor caso de rendimiento. Las aplicaciones de pruebas de rendimiento sintéticas se escriben y se utilizan para optimizar el uso de sistemas de archivos subyacentes de formas que las aplicaciones del mundo real podrían no.

Ajuste de tamaño del rendimiento

La siguiente tabla proporciona un ajuste del tamaño del rendimiento recomendado.

Perfil de configuración	1MiB lee	1MiB escribe
Metadatos + almacenamiento	62GiBps	21 GiBps
Solo almacenamiento	64 GiBps	21 GiBps

Las estimaciones de tamaño de la capacidad de metadatos se basan en la "regla general" según la cual 500 GB de capacidad son suficientes para aproximadamente 150 millones de archivos en BeeGFS. (Para obtener más información, consulte la documentación de BeeGFS para "Requisitos del sistema".)

El uso de funciones como las listas de control de acceso y el número de directorios y archivos por directorio también afecta a la rapidez con la que se consume el espacio de metadatos. Las estimaciones de la capacidad de almacenamiento dan cuenta de la capacidad de unidad utilizable junto con la sobrecarga de RAID 6 y XFS.

Configuración de la capacidad para metadatos + elementos básicos de almacenamiento

La siguiente tabla proporciona un tamaño de capacidad recomendado para metadatos, además de los elementos básicos de almacenamiento.

Tamaño de la unidad (2+2 RAID 1) grupos de volúmenes de metadatos	Capacidad de metadatos (cantidad de archivos)	Grupos de volúmenes de almacenamiento de tamaño de unidad (8+2 RAID 6)	Capacidad de almacenamiento (contenido de archivos)
1,92 TB	1,938,577,200	1,92 TB	51,77 TB
3,84 TB	3,880,388,400	3,84 TB	103,55 TB
7,68 TB	8,125,278,000	7,68 TB	216.74 TB
15,3 TB	17,269,854,000	15,3 TB	460 TB

Al ajustar el tamaño de los metadatos más los elementos básicos de almacenamiento, puede reducir los costes usando unidades más pequeñas para los grupos de volúmenes de metadatos frente a los grupos de volúmenes de almacenamiento.

Configuración de la capacidad para los elementos básicos solo del almacenamiento

La siguiente tabla proporciona ajuste de tamaño de capacidad de regla general para elementos básicos de solo almacenamiento.

Grupos de volúmenes de almacenamiento de tamaño de unidad (10+2 RAID 6)	Capacidad de almacenamiento (contenido de archivos)
1,92 TB	59,89 TB
3,84 TB	1319,80 TB
7,68 TB	251.89TB
15,3 TB	538,55 TB

La sobrecarga de rendimiento y capacidad de incluir el servicio de gestión en el elemento básico (primero) es mínima, a menos que se habilite el bloqueo global de archivos.

Ajuste del rendimiento

La solución BeeGFS incluye recomendaciones para el ajuste del rendimiento basadas en pruebas de verificación.

Si bien BeeGFS proporciona un rendimiento razonable desde el momento de su instalación, NetApp ha desarrollado un conjunto de parámetros de ajuste recomendados para maximizar el rendimiento. Estos parámetros tienen en cuenta las funcionalidades de los nodos de bloque E-Series subyacentes y todos los requisitos especiales necesarios para ejecutar BeeGFS en una arquitectura de alta disponibilidad de disco compartido.

Ajuste del rendimiento de los nodos de archivos

Los parámetros de ajuste disponibles que puede configurar son los siguientes:

1. **Configuración del sistema en el UEFI/BIOS de nodos de archivos.** para maximizar el rendimiento, recomendamos configurar los ajustes del sistema en el modelo de servidor que utilice como nodos de archivos. Los ajustes del sistema se configuran cuando se configuran los nodos de archivos mediante la configuración del sistema (UEFI/BIOS) o las API Redfish proporcionadas por el controlador de administración de la placa base (BMC).

   La configuración del sistema varía en función del modelo de servidor que utilice como nodo de archivos. Los ajustes deben configurarse manualmente en función del modelo de servidor que se esté utilizando. Para aprender a configurar los ajustes del sistema para los nodos de archivo Lenovo SR665 validados, consulte "Ajuste la configuración del sistema del nodo de archivos para aumentar el rendimiento".

2. **Configuración predeterminada de los parámetros de configuración necesarios.** los parámetros de configuración necesarios afectan a la forma en que se configuran los servicios BeeGFS y cómo los volúmenes E-Series (dispositivos de bloques) se formatean y montan mediante Pacemaker. Entre estos

parámetros de configuración necesarios se incluyen los siguientes:

- Parámetros de configuración del servicio BeeGFS

  Es posible anular la configuración predeterminada para los parámetros de configuración según sea necesario. Para ver los parámetros que se pueden ajustar para sus cargas de trabajo específicas o sus casos de uso, consulte "Parámetros de configuración del servicio BeeGFS".

- El formato de los volúmenes y los parámetros de montaje se establecen en los valores predeterminados recomendados y solo se deben ajustar en casos prácticos avanzados. Los valores predeterminados harán lo siguiente:
  - Optimización del formato de volumen inicial basado en el tipo de destino (como la gestión, los metadatos o el almacenamiento), junto con la configuración de RAID y el tamaño de segmentos del volumen subyacente.
  - Ajuste cómo monta Pacemaker cada volumen para garantizar que los cambios se vacíen inmediatamente a los nodos de bloque E-Series. De este modo se evita la pérdida de datos cuando fallan nodos de archivos con las escrituras activas en curso.

    Para ver los parámetros que se pueden ajustar para sus cargas de trabajo específicas o sus casos de uso, consulte "parámetros de configuración de formato de volumen y montaje".

3. **Configuración del sistema en el sistema operativo Linux instalado en los nodos de archivo.** puede anular la configuración predeterminada del sistema operativo Linux cuando cree el inventario de Ansible en el paso 4 de "Cree el inventario de Ansible".

   La configuración predeterminada se utilizó para validar BeeGFS en la solución de NetApp, pero es posible modificarla para adaptarla a sus cargas de trabajo o casos de uso específicos. Algunos ejemplos de la configuración del sistema operativo Linux que puede cambiar son los siguientes:

   - Las colas de I/o en dispositivos de bloques E-Series.

     Se pueden configurar colas de I/o en los dispositivos de bloque E-Series que se utilizan como destinos BeeGFS para:

     - Ajuste el algoritmo de programación en función del tipo de dispositivo (NVMe, HDD, etc.).
     - Aumentar el número de solicitudes pendientes.
     - Ajustar los tamaños de las solicitudes.
     - Optimice el comportamiento de lectura anticipada.
   - Ajustes de memoria virtual.

     Puede ajustar la configuración de memoria virtual para obtener un rendimiento de transmisión sostenido óptimo.

   - Configuración de CPU.

     Puede ajustar el regulador de frecuencia de la CPU y otras configuraciones de la CPU para obtener el máximo rendimiento.

   - Tamaño de solicitud de lectura.

     Puede aumentar el tamaño máximo de solicitud de lectura para los profesionales de Mellanox.

Ajuste del rendimiento para nodos de bloques

En función de los perfiles de configuración aplicados a un bloque de creación de BeeGFS en particular, los grupos de volúmenes configurados en los nodos de bloque cambian ligeramente. Por ejemplo, con un nodo de bloque EF600 de 24 unidades:

- Para el único elemento básico, incluidos los servicios de gestión, metadatos y almacenamiento de BeeGFS:
  - 1 grupo de volúmenes de 2+2 RAID 10 para servicios de metadatos y gestión de BeeGFS
  - 2 grupos de volúmenes RAID 6 de 8+2 para servicios de almacenamiento BeeGFS
- Para un bloque básico de metadatos BeeGFS + almacenamiento:
  - 1 grupo de volúmenes de 2+2 RAID 10 para servicios de metadatos BeeGFS
  - 2 grupos de volúmenes RAID 6 de 8+2 para servicios de almacenamiento BeeGFS
- Para el almacenamiento BeeGFS, solo elemento básico:
  - 2 grupos de volúmenes RAID 6 de 10+2 para servicios de almacenamiento BeeGFS

> Como BeeGFS necesita menos espacio de almacenamiento para la gestión y los metadatos en comparación con el almacenamiento, una opción es utilizar unidades más pequeñas para los grupos de volúmenes RAID 10. Las unidades más pequeñas deben llenarse en las ranuras de unidad más externas. Para obtener más información, consulte "instrucciones de puesta en funcionamiento".

Todos estos ajustes se configuran mediante la puesta en marcha basada en Ansible, junto con otros ajustes que suelen recomendarse para optimizar el rendimiento o el comportamiento, entre los que se incluyen:

- Ajustar el tamaño de bloque de caché global a 32 KiB y ajustar el vaciado de caché basado en demanda al 80 %.
- Al deshabilitar el equilibrio de carga automático (se garantiza que las asignaciones de volúmenes de la controladora permanezcan según la definición).
- Habilitar el almacenamiento en caché de lectura y deshabilitar el almacenamiento en caché de lectura anticipada.
- Habilitar el almacenamiento en caché de escritura con mirroring y requerir backup de batería, de modo que la caché se mantiene mediante el fallo de una controladora del nodo de bloque.
- Especificar el orden en que las unidades se asignan a grupos de volúmenes, equilibrando las operaciones de I/o en los canales de unidades disponibles.

Elemento básico de gran capacidad

El diseño de la solución BeeGFS estándar se ha diseñado teniendo en cuenta las cargas de trabajo de alto rendimiento. Los clientes que busquen casos de uso de gran capacidad deben observar las variaciones en las características de diseño y rendimiento descritas aquí.

Configuración de hardware y software

La configuración de hardware y software del elemento básico de alta capacidad es de serie, a excepción de que las controladoras EF600 deben sustituirse por una controladora EF300 con opción de conectarse entre 1

y 7 bandejas de expansión IOM con 60 unidades cada una para cada cabina de almacenamiento, un total de 2 a 14 bandejas de expansión por bloque de construcción.

Es probable que los clientes que implementan un diseño de elemento básico de gran capacidad utilicen solo la configuración del estilo de bloque de creación base compuesta por servicios de gestión, metadatos y almacenamiento de BeeGFS para cada nodo. Para reducir la rentabilidad, los nodos de almacenamiento de gran capacidad deben aprovisionar volúmenes de metadatos en las unidades NVMe en el compartimento de controladora EF300 y deben aprovisionar volúmenes de almacenamiento a las unidades NL-SAS de las bandejas de expansión.

[]

Directrices de tamaño

Estas directrices de tamaño suponen que los bloques básicos de gran capacidad se configuran con un grupo de volúmenes SSD de 2+2 NVMe para los metadatos en el compartimento EF300 básico y 6 grupos de volúmenes NL-SAS de 8+2 por bandeja de ampliación IOM para el almacenamiento.

Tamaño de unidad (HDD de capacidad)	Capacidad por BB (1 bandeja)	Capacidad por BB (2 bandejas)	Capacidad por BB (3 bandejas)	Capacidad por BB (4 bandejas)
4 TB	439 TB	878 TB	1317 TB	1756 TB
8 TB	878 TB	1756 TB	2634 TB	3512 TB
10 TB	1097 TB	2195 TB	3292 TB	4390 TB
12 TB	1317 TB	2634 TB	3951 TB	5268 TB
16 TB	1756 TB	3512 TB	5268 TB	7024 TB
18 TB	1975 TB	3951 TB	5927 TB	7902 TB