



Implemente el sistema de archivos BeeGFS

BeeGFS on NetApp with E-Series Storage

NetApp
January 27, 2026

Tabla de contenidos

| | |
|---|----|
| Implemente el sistema de archivos BeeGFS | 1 |
| Descripción general del libro de estrategia de Ansible | 1 |
| Descripción general | 1 |
| Ansible: Conceptos clave | 1 |
| Rol de ha de BeeGFS para Ansible: Conceptos clave | 1 |
| Ponga en marcha el clúster de alta disponibilidad de BeeGFS | 2 |
| Descripción general | 2 |
| Pasos | 2 |
| Implemente clientes BeeGFS | 6 |
| Descripción general | 6 |
| Pasos | 6 |
| Verifique la implementación de BeeGFS | 11 |
| Descripción general | 11 |

Implemente el sistema de archivos BeeGFS

Descripción general del libro de estrategia de Ansible

Puesta en marcha y gestión de clústeres de alta disponibilidad de BeeGFS mediante Ansible.

Descripción general

En las secciones anteriores se han realizado los pasos necesarios para crear un inventario de Ansible que represente un clúster de alta disponibilidad de BeeGFS. En esta sección se presenta la automatización de Ansible desarrollada por NetApp para poner en marcha y gestionar el clúster.

Ansible: Conceptos clave

Antes de continuar, es útil estar familiarizado con algunos conceptos clave de Ansible:

- Las tareas que se deben ejecutar con un inventario de Ansible se definen en lo que se conoce como **playbook**.
 - La mayoría de las tareas en Ansible están diseñadas para ser **idempotente**, lo que significa que pueden ejecutarse varias veces para verificar que la configuración/estado deseada todavía se aplica sin romper las cosas ni hacer actualizaciones innecesarias.
- La unidad más pequeña de ejecución en Ansible es un **módulo**.
 - Los libros de estrategia habituales utilizan varios módulos.
 - Ejemplos: Descargue un paquete, actualice un archivo de configuración, inicie/habilite un servicio.
 - NetApp distribuye módulos para automatizar los sistemas E-Series de NetApp.
- La automatización compleja se empaquetará mejor como rol.
 - Básicamente, un formato estándar para distribuir un libro de aplicaciones reutilizable.
 - NetApp distribuye roles para hosts Linux y sistemas de archivos BeeGFS.

Rol de ha de BeeGFS para Ansible: Conceptos clave

Toda la automatización necesaria para poner en marcha y gestionar cada versión de BeeGFS en NetApp se presenta como un rol de Ansible y se distribuye como parte de la "[Colección de Ansible E-Series de NetApp para BeeGFS](#)":

- Este papel se puede considerar como un lugar entre un motor de **instalador** y un motor de **implementación/administración** moderno para BeeGFS.
 - Aplica una infraestructura moderna como prácticas de código y filosofías para simplificar la gestión de infraestructura de almacenamiento a cualquier escala.
 - De forma similar a cómo el "[Kubespray](#)" proyecto permite a los usuarios implementar y mantener toda una distribución de Kubernetes para una infraestructura informática de escalado horizontal.
- Este rol es el formato **definido por software** que utiliza NetApp para empaquetar, distribuir y mantener BeeGFS en soluciones NetApp.
 - Esforzarse por crear una experiencia "similar a un dispositivo" sin necesidad de distribuir una distribución entera de Linux o una imagen grande.

- Incluye agentes de recursos en clúster compatibles con Open Cluster Framework (OCF) de NetApp para objetivos BeeGFS personalizados, direcciones IP y supervisión que proporcionan una integración inteligente con Pacemaker/BeeGFS.
- Esta función no se limita a la puesta en marcha de la «automatización», sino que está destinada a gestionar todo el ciclo de vida del sistema de archivos, incluidos:
 - Aplicar cambios y actualizaciones de configuración por servicio o para todo el clúster.
 - Automatizar la reparación y recuperación del clúster después de resolver problemas de hardware.
 - Simplificación del ajuste del rendimiento con valores predeterminados establecidos en función de amplias pruebas con BeeGFS y volúmenes de NetApp.
 - Verificación y corrección de deriva de configuración.

NetApp también proporciona un rol de Ansible para "["Clientes de BeeGFS"](#)", Que se puede utilizar opcionalmente para instalar BeeGFS y montar sistemas de archivos en nodos de cálculo/GPU/inicio de sesión.

Ponga en marcha el clúster de alta disponibilidad de BeeGFS

Especifique qué tareas se deben ejecutar para poner en marcha el clúster de alta disponibilidad de BeeGFS mediante un libro de aplicaciones.

Descripción general

En esta sección se describe cómo montar el libro de estrategia estándar utilizado para poner en marcha/gestionar BeeGFS en NetApp.

Pasos

Cree el libro de aplicaciones de Ansible

Cree el archivo `playbook.yml` y rellenarlo de la siguiente manera:

1. Defina primero un conjunto de tareas (comúnmente denominado a) "["repr"](#)") Que solo se debe ejecutar en los nodos de bloques E-Series de NetApp. Utilizamos una tarea de pausa para preguntar antes de ejecutar la instalación (para evitar la ejecución accidental de playbook) y, a continuación, importar la `nar_santricity_management` función. Esta función se ocupa de aplicar cualquier configuración general del sistema definida en `group_vars/eseries_storage_systems.yml` o individual `host_vars/<BLOCK NODE>.yml` archivos.

```

- hosts: eseries_storage_systems
  gather_facts: false
  collections:
    - netapp_eseries.santricity
  tasks:
    - name: Verify before proceeding.
      pause:
        prompt: "Are you ready to proceed with running the BeeGFS HA role? Depending on the size of the deployment and network performance between the Ansible control node and BeeGFS file and block nodes this can take awhile (10+ minutes) to complete."
    - name: Configure NetApp E-Series block nodes.
      import_role:
        name: nar_santricity_management

```

2. Defina la reproducción que se ejecutará en todos los nodos de archivos y bloques:

```

- hosts: all
  any_errors_fatal: true
  gather_facts: false
  collections:
    - netapp_eseries.beegfs

```

3. En esta aplicación podemos definir opcionalmente un conjunto de «tareas previas» que se deben ejecutar antes de poner en marcha el clúster de alta disponibilidad. Esto puede ser útil para verificar/installar requisitos previos como Python. También podemos inyectar comprobaciones previas al vuelo, por ejemplo, verificar que las etiquetas de Ansible proporcionadas son compatibles:

```

pre_tasks:
  - name: Ensure a supported version of Python is available on all file nodes.
    block:
      - name: Check if python is installed.
        failed_when: false
        changed_when: false
        raw: python --version
        register: python_version

      - name: Check if python3 is installed.
        raw: python3 --version
        failed_when: false
        changed_when: false
        register: python3_version
        when: 'python_version["rc"] != 0 or (python_version["stdout"]'

```

```

| regex_replace("Python ", "")) is not version("3.0", ">=")

- name: Install python3 if needed.
  raw: |
    id=$(grep "^ID=" /etc/*release* | cut -d= -f 2 | tr -d '')
    case $id in
      ubuntu) sudo apt install python3 ;;
      rhel|centos) sudo yum -y install python3 ;;
      sles) sudo zypper install python3 ;;
    esac
  args:
    executable: /bin/bash
  register: python3_install
  when: python_version['rc'] != 0 and python3_version['rc'] != 0
  become: true

- name: Create a symbolic link to python from python3.
  raw: ln -s /usr/bin/python3 /usr/bin/python
  become: true
  when: python_version['rc'] != 0
  when: inventory_hostname not in
groups[beegfs_ha_ansible_storage_group]

- name: Verify any provided tags are supported.
  fail:
    msg: "{{ item }} tag is not a supported BeeGFS HA tag. Rerun
your playbook command with --list-tags to see all valid playbook tags."
    when: 'item not in ["all", "storage", "beegfs_ha",
"beegfs_ha_package", "beegfs_ha_configure",
"beegfs_ha_configure_resource", "beegfs_ha_performance_tuning",
"beegfs_ha_backup", "beegfs_ha_client"]'
  loop: "{{ ansible_run_tags }}"

```

4. Por último, este juego importa el rol de ha de BeeGFS para la versión de BeeGFS que desea implementar:

```

tasks:
- name: Verify the BeeGFS HA cluster is properly deployed.
  import_role:
    name: beegfs_ha_7_4 # Alternatively specify: beegfs_ha_7_3.

```



Se mantiene un rol de ha de BeeGFS para cada versión principal.secundaria de BeeGFS admitida. Esto permite a los usuarios elegir cuándo desean actualizar versiones principales o secundarias. Actualmente (beegfs_7_3('beegfs_7_2' se admiten BeeGFS 7,3.x) o BeeGFS 7,2.x). De forma predeterminada, ambos roles implementarán la versión más reciente del parche de BeeGFS en el momento de su publicación, aunque los usuarios pueden optar por anular este parche e implementar el último parche si lo desean. Consulte la última "["guía de actualización"](#)"información para obtener más información.

5. Opcional: Si desea definir tareas adicionales, tenga en cuenta si las tareas deben ser dirigidas a11 Los hosts (incluidos los sistemas de almacenamiento E-Series) o solo los nodos de archivos. Si es necesario, defina una nueva reproducción dirigida específicamente a los nodos de archivo mediante - hosts: ha_cluster.

Haga clic en "["aqui"](#)" por ejemplo, un archivo de libro de estrategia completo.

Instale las colecciones Ansible de NetApp

Se mantiene la colección BeeGFS para Ansible y todas las dependencias "["Galaxia de ansible"](#)". En su nodo de control de Ansible, ejecute el siguiente comando para instalar la versión más reciente:

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

Aunque normalmente no se recomienda, también es posible instalar una versión específica de la colección:

```
ansible-galaxy collection install netapp_eseries.beegfs:  
==<MAJOR>.<MINOR>.<PATCH>
```

Ejecute el libro de aplicaciones

Desde el directorio del nodo de control de Ansible que contiene el `inventory.yml` y.. `playbook.yml` ejecute el libro de estrategia de la siguiente forma:

```
ansible-playbook -i inventory.yml playbook.yml
```

Según el tamaño del clúster, la puesta en marcha inicial puede tardar más de 20 minutos. Si se produce un error en la puesta en marcha por algún motivo, solo tiene que corregir los problemas (p. ej., un cableado incorrecto, nodo no se inició, etc.) y reiniciar el libro de estrategia de Ansible.

Al especificar "["configuración común de nodos de archivos"](#)", si selecciona la opción predeterminada para que Ansible gestione automáticamente la autenticación basada en conexión, `connAuthFile` ahora se puede encontrar un secreto utilizado como secreto compartido en `<playbook_dir>/files/beegfs/<sysMgmtdHost>_connAuthFile` (de forma predeterminada). Cualquier cliente que necesite acceder al sistema de archivos tendrá que utilizar este secreto compartido. Esto se gestiona automáticamente si los clientes se configuran mediante el "["Función de cliente de BeeGFS"](#)".

Implemente clientes BeeGFS

Opcionalmente, Ansible se puede usar para configurar los clientes de BeeGFS y montar el sistema de archivos.

Descripción general

Para acceder a los sistemas de archivos BeeGFS es necesario instalar y configurar el cliente BeeGFS en cada nodo que necesite montar el sistema de archivos. En esta sección se documenta cómo realizar estas tareas mediante el útil disponible "["Rol de Ansible"](#)".

Pasos

Cree el archivo de inventario de cliente

1. Si es necesario, configure SSH sin contraseñas desde el nodo de control de Ansible a cada uno de los hosts que desea configurar como clientes BeeGFS:

```
ssh-copy-id <user>@<HOSTNAME_OR_IP>
```

2. Inferior `host_vars/`, Cree un archivo para cada cliente BeeGFS denominado `<HOSTNAME>.yml` con el siguiente contenido, rellene el texto del marcador de posición con la información correcta para su entorno:

```
# BeeGFS Client
ansible_host: <MANAGEMENT_IP>
```

3. Incluya de manera opcional uno de los siguientes elementos si desea utilizar los roles de recogida de hosts E-Series de NetApp para configurar las interfaces InfiniBand o Ethernet para que los clientes se conecten a los nodos de archivos BeeGFS:

- a. Si el tipo de red es "["InfiniBand \(uso de IPoIB\)"](#):

```
eseries_ipoib_interfaces:
- name: <INTERFACE> # Example: ib0 or i1b
  address: <IP/SUBNET> # Example: 100.127.100.1/16
- name: <INTERFACE> # Additional interfaces as needed.
  address: <IP/SUBNET>
```

- b. Si el tipo de red es "["RDMA sobre Ethernet convergente \(roce\)"](#):

```
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. Si el tipo de red es "Ethernet (solo TCP, sin 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>
```

4. Cree un archivo nuevo `client_inventory.yml` Y especifique el usuario que Ansible debe usar para conectarse a cada cliente y la contraseña que Ansible debe usar para el escalado de privilegios (esto requiere `ansible_ssh_user` sea raíz o tenga privilegios sudo):

```
# BeeGFS client inventory.  
all:  
  vars:  
    ansible_ssh_user: <USER>  
    ansible_become_password: <PASSWORD>
```



No almacene contraseñas en texto sin formato. En su lugar, utilice Ansible Vault (consulte "["Documentación de Ansible"](#) Para cifrar contenido con Ansible Vault) o usar el --ask-become-pass al ejecutar el libro de estrategia.

5. En la `client_inventory.yml` File, enumera todos los hosts que deben configurarse como clientes BeeGFS en `beegfs_clients` Agrupe y, a continuación, consulte los comentarios en línea y elimine los comentarios de cualquier configuración adicional necesaria para crear el módulo de kernel de cliente BeeGFS en su sistema:

```

children:
    # Ansible group representing all BeeGFS clients:
beegfs_clients:
    hosts:
        <CLIENT HOSTNAME>:
            # Additional clients as needed.

    vars:
        # OPTION 1: If you're using the NVIDIA OFED drivers and they are
        already installed:
        #eseries_ib_skip: True # Skip installing inbox drivers when
        using the IPoIB role.
        #beegfs_client_ofed_enable: True
        #beegfs_client_ofed_include_path:
        "/usr/src/ofa_kernel/default/include"

        # OPTION 2: If you're using inbox IB/RDMA drivers and they are
        already installed:
        #eseries_ib_skip: True # Skip installing inbox drivers when
        using the IPoIB role.

        # OPTION 3: If you want to use inbox IB/RDMA drivers and need
        them installed/configured.
        #eseries_ib_skip: False # Default value.
        #beegfs_client_ofed_enable: False # Default value.

```



Cuando utilice los controladores OFED de NVIDIA, asegúrese de que `beegfs_CLIENT_ofed_INCLUDE_PATH` apunte a la ruta de acceso de inclusión de encabezado correcta para la instalación de Linux. Para obtener más información, consulte la documentación de BeeGFS para "["Compatibilidad con RDMA"](#)".

6. En la `client_inventory.yml` Archivo, enumere los sistemas de archivos BeeGFS que desea montar en cualquiera de los definidos previamente `vars`:

```

beegfs_client_mounts:
  - sysMgtdHost: <IP ADDRESS> # Primary IP of the BeeGFS
management service.
    mount_point: /mnt/beegfs      # Path to mount BeeGFS on the
client.

  connInterfaces:
    - <INTERFACE> # Example: ibs4f1
    - <INTERFACE>

  beegfs_client_config:
    # Maximum number of simultaneous connections to the same
node.
    connMaxInternodeNum: 128 # BeeGFS Client Default: 12
    # Allocates the number of buffers for transferring IO.
    connRDMABufNum: 36 # BeeGFS Client Default: 70
    # Size of each allocated RDMA buffer
    connRDMABufSize: 65536 # BeeGFS Client Default: 8192
    # Required when using the BeeGFS client with the shared-
disk HA solution.
    # This does require BeeGFS targets be mounted in the
default "sync" mode.
    # See the documentation included with the BeeGFS client
role for full details.
    sysSessionChecksEnabled: false
    # Specify additional file system mounts for this or other file
systems.

```

7. A partir de BeeGFS 7.2.7 y 7.3.1 "[autenticación de conexión](#)" se deben configurar o desactivar explícitamente. Dependiendo de cómo decida configurar la autenticación basada en conexión al especificar "[configuración común de nodos de archivos](#)", puede que necesite ajustar la configuración del cliente:

- De forma predeterminada, la puesta en marcha del clúster de alta disponibilidad configurará automáticamente la autenticación de conexiones y generará un `connauthfile` Que se colocará/mantendrá en el nodo de control de Ansible en `<INVENTORY>/files/beegfs/<sysMgtdHost>_connAuthFile`. De forma predeterminada, la función de cliente BeeGFS está configurada para leer/distribuir este archivo a los clientes definidos en `client_inventory.yml`, y no se necesita ninguna acción adicional.
 - Para obtener información sobre las opciones avanzadas, consulte la lista completa de valores predeterminados que se incluyen con la "[Función de cliente de BeeGFS](#)".
- Si decide especificar un secreto personalizado con `beegfs_ha_conn_auth_secret` especifique en la `client_inventory.yml` también archivo:

```
beegfs_ha_conn_auth_secret: <SECRET>
```

- Si decide deshabilitar la autenticación basada en conexión completamente con `beegfs_ha_conn_auth_enabled`, especifique que en la `client_inventory.yml` también

archivo:

```
beegfs_ha_conn_auth_enabled: false
```

Para obtener una lista completa de los parámetros admitidos y detalles adicionales, consulte la ["Documentación completa del cliente de BeeGFS"](#). Para ver un ejemplo completo de un inventario de cliente, haga clic en ["aquí"](#).

Cree el archivo del libro de aplicaciones del cliente BeeGFS

1. Cree un archivo nuevo `client_playbook.yml`

```
# BeeGFS client playbook.
- hosts: beegfs_clients
  any_errors_fatal: true
  gather_facts: true
  collections:
    - netapp_eseries.beegfs
    - netapp_eseries.host
  tasks:
```

2. Opcional: Si desea utilizar los roles de la recogida de hosts de E-Series de NetApp para configurar interfaces para que los clientes se conecten a sistemas de archivos BeeGFS, importe el rol correspondiente al tipo de interfaz que está configurando:

- a. Si utiliza InfiniBand (IPoIB):

```
- name: Ensure IPoIB is configured
  import_role:
    name: ipoib
```

- b. Si utiliza RDMA over Converged Ethernet (roce):

```
- name: Ensure IPoIB is configured
  import_role:
    name: roce
```

- c. Si utiliza Ethernet (solo TCP, no RDMA):

```
- name: Ensure IPoIB is configured
  import_role:
    name: ip
```

3. Por último, importe la función de cliente de BeeGFS para instalar el software cliente y configurar los montajes del sistema de archivos:

```
# REQUIRED: Install the BeeGFS client and mount the BeeGFS file system.  
- name: Verify the BeeGFS clients are configured.  
  import_role:  
    name: beegfs_client
```

Para ver un ejemplo completo de un libro de aplicaciones del cliente, haga clic en "[aquí](#)".

Ejecute el libro de aplicaciones del cliente BeeGFS

Para instalar/crear el cliente y montar BeeGFS, ejecute el siguiente comando:

```
ansible-playbook -i client_inventory.yml client_playbook.yml
```

Verifique la implementación de BeeGFS

Compruebe la implementación del sistema de archivos antes de colocar el sistema en producción.

Descripción general

Antes de poner el sistema de archivos BeeGFS en producción, realice algunas comprobaciones de verificación.

Pasos

1. Inicie sesión en cualquier cliente y ejecute lo siguiente para garantizar que todos los nodos esperados estén presentes o sean accesibles, no se han notificado inconsistencias ni otros problemas:

```
beegfs-fsck --checkfs
```

2. Apague todo el clúster y, a continuación, reinícielo. Desde cualquier nodo de archivo ejecute lo siguiente:

```
pcs cluster stop --all # Stop the cluster on all file nodes.  
pcs cluster start --all # Start the cluster on all file nodes.  
pcs status # Verify all nodes and services are started and no failures  
are reported (the command may need to be reran a few times to allow time  
for all services to start).
```

3. Coloque cada nodo en espera y compruebe que los servicios de BeeGFS pueden comutar por error a nodos secundarios. Para realizar este inicio de sesión en cualquiera de los nodos de archivo y ejecute lo

siguiente:

```
pcs status # Verify the cluster is healthy at the start.  
pcs node standby <FILE NODE HOSTNAME> # Place the node under test in  
standby.  
pcs status # Verify services are started on a secondary node and no  
failures are reported.  
pcs node unstandby <FILE NODE HOSTNAME> # Take the node under test out  
of standby.  
pcs status # Verify the file node is back online and no failures are  
reported.  
pcs resource relocate run # Move all services back to their preferred  
nodes.  
pcs status # Verify services have moved back to the preferred node.
```

4. Utilizar herramientas de evaluación del rendimiento como IOR y MDTest para verificar que el rendimiento del sistema de archivos cumple las expectativas. Se pueden encontrar ejemplos de pruebas y parámetros comunes utilizados con BeeGFS "["Verificación del diseño"](#)" en la sección de BeeGFS on NetApp Verified Architecture.

Las pruebas adicionales deben realizarse en función de los criterios de aceptación definidos para una instalación/emplazamiento particular.

Información de copyright

Copyright © 2026 NetApp, Inc. Todos los derechos reservados. Imprimido en EE. UU. No se puede reproducir este documento protegido por copyright ni parte del mismo de ninguna forma ni por ningún medio (gráfico, electrónico o mecánico, incluidas fotocopias, grabaciones o almacenamiento en un sistema de recuperación electrónico) sin la autorización previa y por escrito del propietario del copyright.

El software derivado del material de NetApp con copyright está sujeto a la siguiente licencia y exención de responsabilidad:

ESTE SOFTWARE LO PROPORCIONA NETAPP «TAL CUAL» Y SIN NINGUNA GARANTÍA EXPRESA O IMPLÍCITA, INCLUYENDO, SIN LIMITAR, LAS GARANTÍAS IMPLÍCITAS DE COMERCIALIZACIÓN O IDONEIDAD PARA UN FIN CONCRETO, CUYA RESPONSABILIDAD QUEDA EXIMIDA POR EL PRESENTE DOCUMENTO. EN NINGÚN CASO NETAPP SERÁ RESPONSABLE DE NINGÚN DAÑO DIRECTO, INDIRECTO, ESPECIAL, EJEMPLAR O RESULTANTE (INCLUYENDO, ENTRE OTROS, LA OBTENCIÓN DE BIENES O SERVICIOS SUSTITUTIVOS, PÉRDIDA DE USO, DE DATOS O DE BENEFICIOS, O INTERRUPCIÓN DE LA ACTIVIDAD EMPRESARIAL) CUALQUIERA SEA EL MODO EN EL QUE SE PRODUJERON Y LA TEORÍA DE RESPONSABILIDAD QUE SE APLIQUE, YA SEA EN CONTRATO, RESPONSABILIDAD OBJETIVA O AGRAVIO (INCLUIDA LA NEGLIGENCIA U OTRO TIPO), QUE SURJAN DE ALGÚN MODO DEL USO DE ESTE SOFTWARE, INCLUSO SI HUBIEREN SIDO ADVERTIDOS DE LA POSIBILIDAD DE TALES DAÑOS.

NetApp se reserva el derecho de modificar cualquiera de los productos aquí descritos en cualquier momento y sin aviso previo. NetApp no asume ningún tipo de responsabilidad que surja del uso de los productos aquí descritos, excepto aquello expresamente acordado por escrito por parte de NetApp. El uso o adquisición de este producto no lleva implícita ninguna licencia con derechos de patente, de marcas comerciales o cualquier otro derecho de propiedad intelectual de NetApp.

Es posible que el producto que se describe en este manual esté protegido por una o más patentes de EE. UU., patentes extranjeras o solicitudes pendientes.

LEYENDA DE DERECHOS LIMITADOS: el uso, la copia o la divulgación por parte del gobierno están sujetos a las restricciones establecidas en el subpárrafo (b)(3) de los derechos de datos técnicos y productos no comerciales de DFARS 252.227-7013 (FEB de 2014) y FAR 52.227-19 (DIC de 2007).

Los datos aquí contenidos pertenecen a un producto comercial o servicio comercial (como se define en FAR 2.101) y son propiedad de NetApp, Inc. Todos los datos técnicos y el software informático de NetApp que se proporcionan en este Acuerdo tienen una naturaleza comercial y se han desarrollado exclusivamente con fondos privados. El Gobierno de EE. UU. tiene una licencia limitada, irrevocable, no exclusiva, no transferible, no sublicenciable y de alcance mundial para utilizar los Datos en relación con el contrato del Gobierno de los Estados Unidos bajo el cual se proporcionaron los Datos. Excepto que aquí se disponga lo contrario, los Datos no se pueden utilizar, desvelar, reproducir, modificar, interpretar o mostrar sin la previa aprobación por escrito de NetApp, Inc. Los derechos de licencia del Gobierno de los Estados Unidos de América y su Departamento de Defensa se limitan a los derechos identificados en la cláusula 252.227-7015(b) de la sección DFARS (FEB de 2014).

Información de la marca comercial

NETAPP, el logotipo de NETAPP y las marcas que constan en <http://www.netapp.com/TM> son marcas comerciales de NetApp, Inc. El resto de nombres de empresa y de producto pueden ser marcas comerciales de sus respectivos propietarios.