



# Automatización mediante API de REST

## SnapCenter Software 4.5

NetApp  
September 29, 2025

This PDF was generated from [https://docs.netapp.com/es-es/snapcenter-45/sc-automation/overview\\_rest\\_apis.html](https://docs.netapp.com/es-es/snapcenter-45/sc-automation/overview_rest_apis.html) on September 29, 2025. Always check [docs.netapp.com](https://docs.netapp.com) for the latest.

# Tabla de contenidos

Automatización mediante API de REST .....	1
Información general de las API de REST .....	1
Cómo acceder a la API DE REST de SnapCenter de forma nativa .....	1
Base de servicios web DE REST .....	1
Recursos y representación estatal .....	1
Extremos de URI .....	2
Mensajes HTTP .....	2
Formato JSON .....	2
Variables de entrada que controlan una solicitud API .....	2
Métodos HTTP .....	2
Solicitar encabezados .....	3
Solicitar el cuerpo .....	3
Filtrando objetos .....	3
Solicitando campos de objeto específicos .....	4
Ordenar objetos del conjunto de resultados .....	4
Paginación al recuperar objetos de una colección .....	4
Propiedades de tamaño .....	5
Interpretación de una respuesta API .....	6
Código de estado HTTP .....	6
Encabezados de respuesta .....	6
Cuerpo de respuesta .....	7
Errores .....	7
API de REST compatibles .....	8
Autor .....	8
Dominios .....	8
Trabajos .....	8
Configuración .....	9
Hosts .....	9
Recursos .....	9
Completos .....	9
Clones .....	9
Clanesplit .....	9
Grupos de recursos .....	10
Normativas .....	10
Reducida .....	10
Share .....	10
Complementos .....	10
Leídos .....	10
Alertas .....	10
RBAC .....	11
Configuración .....	11
CertificadosConfiguración .....	11
Repositorio .....	11

Cómo acceder a las API de REST a través de la página web de API de Swagger . . . . .	11
Comience con la API DE REST . . . . .	12
Hola Mundo . . . . .	12

# Automatización mediante API de REST

## Información general de las API de REST

Las API DE REST pueden utilizarse para realizar varias operaciones de gestión de SnapCenter. Las API DE REST se exponen a través de la página web de Swagger.

Es posible acceder a la página web de Swagger disponible en [https://<SnapCenter\\_IP\\_address\\_or\\_name>:<SnapCenter\\_port>/swagger/](https://<SnapCenter_IP_address_or_name>:<SnapCenter_port>/swagger/) para mostrar la documentación de API DE REST, y también para emitir manualmente una llamada API.

Los plugins compatibles con las API DE REST son:

- Plugin para Microsoft SQL Server
- Plugin para base de datos SAP HANA
- Plugins personalizados

## Cómo acceder a la API DE REST de SnapCenter de forma nativa

Puede acceder a la API DE REST de SnapCenter directamente desde cualquier lenguaje de programación que admita un cliente REST. Entre las opciones de idiomas populares se incluyen Python, PowerShell y Java.

## Base de servicios web DE REST

La transferencia de estado representacional (REST) es un estilo para crear aplicaciones web distribuidas. Cuando se aplica al diseño de una API de servicios web, establece un conjunto de tecnologías y prácticas recomendadas para exponer recursos basados en servidor y administrar sus estados. Utiliza estándares y protocolos más utilizados para proporcionar una base flexible para la gestión de SnapCenter.

### Recursos y representación estatal

Los recursos son los componentes básicos de un sistema basado en la Web. Al crear una aplicación DE SERVICIOS web DE REST, las tareas de diseño más tempranas incluyen:

#### Identificación de recursos basados en sistemas o servidores

Cada sistema utiliza y mantiene los recursos. Un recurso puede ser un archivo, una transacción comercial, un proceso o una entidad administrativa. Una de las primeras tareas en el diseño de una aplicación basada en servicios web DE REST es identificar los recursos.

#### Definición de estados de recursos y operaciones estatales asociadas

Los recursos siempre se encuentran en uno de un número limitado de estados. Los estados, así como las operaciones asociadas utilizadas para influir en los cambios de estado, deben definirse claramente.

## Extremos de URI

Todos los recursos REST deben definirse y ponerse a disposición mediante un esquema de direccionamiento bien definido. Los extremos en los que se encuentran e identifican los recursos utilizan un identificador uniforme de recursos (URI).

El URI proporciona un marco general para crear un nombre único para cada recurso de la red. El Localizador uniforme de recursos (URL) es un tipo de URI que se utiliza con los servicios web para identificar y acceder a los recursos. Los recursos normalmente se exponen en una estructura jerárquica similar a un directorio de archivos.

## Mensajes HTTP

El Protocolo de transferencia de hipertexto (HTTP) es el protocolo utilizado por el cliente y servidor de servicios web para intercambiar mensajes de solicitud y respuesta sobre los recursos.

Como parte del diseño de una aplicación de servicios web, los métodos HTTP se asignan a los recursos y a las correspondientes acciones de administración del estado. HTTP no tiene estado. Por lo tanto, para asociar un conjunto de solicitudes y respuestas relacionadas como parte de una transacción, se debe incluir información adicional en los encabezados HTTP transportados con los flujos de datos de solicitud y respuesta.

## Formato JSON

Aunque la información se puede estructurar y transferir entre un cliente de servicios web y un servidor de varias maneras, la opción más popular es la notación de objetos JavaScript (JSON).

JSON es un estándar del sector para representar estructuras de datos simples en texto sin formato y se utiliza para transferir información de estado que describe los recursos. La API REST de SnapCenter utiliza JSON para formatear los datos transportados en el cuerpo de cada solicitud y respuesta de HTTP.

## Variables de entrada que controlan una solicitud API

Puede controlar cómo se procesa una llamada API mediante parámetros y variables definidas en la solicitud HTTP.

## Métodos HTTP

En la siguiente tabla, se muestran los métodos HTTP compatibles con la API DE REST de SnapCenter.



No todos los métodos HTTP están disponibles en cada extremo DE REST.

Método HTTP	Descripción
OBTENGA	Recupera propiedades de objeto en una instancia o colección de recursos.
PUBLICAR	Crea una nueva instancia de recurso basada en la entrada proporcionada.
ELIMINAR	Elimina una instancia de recurso existente.
PUESTO	Modifica una instancia de recurso existente.

## Solicitar encabezados

Debe incluir varios encabezados en la solicitud HTTP.

### Tipo de contenido

Si el cuerpo de la solicitud incluye JSON, este encabezado debe establecerse en *Application/json*.

### Acepte

Este encabezado debe establecerse en *Application/json*.

### Autorización

La autenticación básica se debe establecer con el nombre de usuario y la contraseña codificados como una cadena base64.

## Solicitar el cuerpo

El contenido del cuerpo de la solicitud varía en función de la llamada específica. El cuerpo de la solicitud HTTP consta de uno de los siguientes elementos:

- Objeto JSON con variables de entrada
- Vacío

## Filtrando objetos

Al emitir una llamada API que utilice GET, puede limitar o filtrar los objetos devueltos en función de cualquier atributo. Por ejemplo, puede especificar un valor exacto para que coincida:

```
<field>=<query value>
```

Además de una coincidencia exacta, hay otros operadores disponibles para devolver un conjunto de objetos sobre un rango de valores. La API REST de SnapCenter es compatible con los operadores de filtrado que se muestran en la tabla siguiente.

Operador	Descripción
=	Igual a.
<	Menor que
>	Mayor que
≤	Menor o igual que
≥	Mayor o igual que
ACTUALIZAR	O.
!	No es igual a.
*	Comodín codicioso

También puede devolver una colección de objetos basándose en si se establece o no un campo específico utilizando la palabra clave **null** o su negación **!null** como parte de la consulta.



Los campos que no están configurados generalmente se excluyen de consultas coincidentes.

## Solicitando campos de objeto específicos

De forma predeterminada, al emitir una llamada API mediante GET, sólo se devuelven los atributos que identifican de forma exclusiva el objeto o los objetos. Este conjunto mínimo de campos actúa como clave para cada objeto y varía según el tipo de objeto. Puede seleccionar propiedades de objeto adicionales mediante la `fields` parámetro de consulta de las siguientes formas:

### Campos comunes o estándar

Especifique `Fields=*` para recuperar los campos de objeto más utilizados. Estos campos normalmente se mantienen en la memoria del servidor local o requieren poco procesamiento para acceder. Estas son las mismas propiedades que se devuelven para un objeto después de utilizar GET con una clave de ruta de URL (UUID).

### Todos los campos

Especifique `Fields=**` para recuperar todos los campos de objeto, incluidos los que requieren procesamiento de servidor adicional para tener acceso.

### Selección de campo personalizado

Utilice `Fields=<field_name>` para especificar el campo exacto que desea. Al solicitar varios campos, los valores deben separarse con comas sin espacios.



Como práctica recomendada, siempre debe identificar los campos específicos que desea. Sólo debe recuperar el conjunto de campos comunes o todos los campos cuando sea necesario. NetApp determina qué campos se clasifican como comunes y se devuelven con `fields=*` en función del análisis de rendimiento interno. La clasificación de un campo puede cambiar en versiones futuras.

## Ordenar objetos del conjunto de resultados

Los registros de una colección de recursos se devuelven en el orden predeterminado definido por el objeto. Puede cambiar el pedido mediante `order_by` consulte el parámetro con el nombre del campo y la dirección de ordenación de la siguiente manera:

```
order_by=<field name> asc|desc
```

Por ejemplo, puede ordenar el campo de tipo en orden descendente seguido de id en orden ascendente:

```
order_by=type desc, id asc
```

- Si especifica un campo de ordenación pero no proporciona una dirección, los valores se ordenan en orden ascendente.
- Cuando se incluyan varios parámetros, los campos deben separarse con una coma.

## Paginación al recuperar objetos de una colección

Al emitir una llamada API mediante GET para acceder a una colección de objetos del mismo tipo, SnapCenter intenta devolver tantos objetos como sea posible basándose en dos restricciones. Puede controlar cada una

de estas restricciones utilizando parámetros de consulta adicionales en la solicitud. La primera restricción alcanzada para una solicitud GET específica termina la solicitud y, por lo tanto, limita el número de registros devueltos.



Si una solicitud finaliza antes de iterar todos los objetos, la respuesta contiene el vínculo necesario para recuperar el siguiente lote de registros.

### **LIMITAR EL NÚMERO DE OBJETOS**

De forma predeterminada, SnapCenter devuelve un máximo de 10,000 objetos para UNA solicitud GET. Puede cambiar este límite utilizando el parámetro de consulta *max\_Records*. Por ejemplo:

```
max_records=20
```

El número de objetos realmente devueltos puede ser menor que el máximo en efecto, basándose en la restricción de tiempo relacionada, así como en el número total de objetos del sistema.

### **LIMITAR EL TIEMPO UTILIZADO PARA RECUPERAR LOS OBJETOS**

De forma predeterminada, SnapCenter devuelve tantos objetos como sea posible dentro del tiempo permitido para LA solicitud GET. El tiempo de espera predeterminado es 15 segundos. Puede cambiar este límite utilizando el parámetro de consulta *return\_TIMEOUT*. Por ejemplo:

```
return_timeout=5
```

El número de objetos realmente devueltos puede ser menor que el máximo en efecto, basándose en la restricción relacionada en el número de objetos así como en el número total de objetos del sistema.

### **REDUCCIÓN DEL CONJUNTO DE RESULTADOS**

Si es necesario, puede combinar estos dos parámetros con parámetros de consulta adicionales para restringir el conjunto de resultados. Por ejemplo, el siguiente devuelve hasta 10 eventos de EMS generados después de la hora especificada:

```
time=> 2018-04-04T15:41:29.140265Z&max_records=10
```

Puede emitir varias solicitudes para desplazarse por los objetos. Cada llamada API posterior debe utilizar un nuevo valor de tiempo basado en el último evento del último conjunto de resultados.

### **PROPIEDADES DE TAMAÑO**

Los valores de entrada utilizados con algunas llamadas API, así como ciertos parámetros de consulta son numéricos. En lugar de proporcionar un entero en bytes, puede usar de manera opcional un sufijo como se muestra en la siguiente tabla.

Sufijo	Descripción
KB	Kilobytes de KB (1024 bytes) o kibibytes
MB	MB megabytes (KB x 1024 bytes) o mebibbytes
GB	GB Gigabytes (MB x 1024 bytes) o gibibbytes
TB	TB terabytes (GB x 1024 bytes) o tebibbytes

Sufijo	Descripción
PB	Petabytes de PB (TB x 1024 byes) o gibibytes

## Interpretación de una respuesta API

Cada solicitud de API genera una respuesta al cliente. Debe examinar la respuesta para determinar si ha tenido éxito y recuperar datos adicionales según sea necesario.

### Código de estado HTTP

A continuación se describen los códigos de estado HTTP utilizados por la API DE REST de SnapCenter.

Codificación	Descripción
200	OK indica éxito para las llamadas que no crean un nuevo objeto.
201	Se ha creado correctamente un objeto. El encabezado de ubicación de la respuesta incluye el identificador único del objeto.
202	Aceptado se ha iniciado Un trabajo en segundo plano para realizar la solicitud, pero aún no se ha completado.
400	Solicitud incorrecta la entrada de la solicitud no se reconoce o no es apropiada.
401	Se ha producido un error en la autenticación de usuario no autorizada.
403	El acceso prohibido se rechaza debido a un error de autorización (RBAC).
404	No se encuentra el recurso al que se hace referencia en la solicitud no existe.
405	Método no permitido el método HTTP en la solicitud no es compatible con el recurso.
409	Conflicto error al intentar crear un objeto porque primero se debe crear otro objeto o ya existe el objeto solicitado.
500	Error interno se ha producido un error interno general en el servidor.

### Encabezados de respuesta

Se incluyen varios encabezados en la respuesta HTTP generada por SnapCenter.

#### Ubicación

Cuando se crea un objeto, el encabezado de ubicación incluye la dirección URL completa del nuevo objeto, incluido el identificador único asignado al objeto.

## **Tipo de contenido**

Esto será normalmente application/json.

## **Cuerpo de respuesta**

El contenido del cuerpo de respuesta que resulta de una solicitud API varía en función del objeto, el tipo de procesamiento y el éxito o el fallo de la solicitud. La respuesta siempre se representa en JSON.

### **Un solo objeto**

Un solo objeto se puede devolver con un conjunto de campos basados en la solicitud. Por ejemplo, se puede usar GET para recuperar las propiedades seleccionadas de un clúster mediante el identificador único.

### **Varios objetos**

Se pueden devolver varios objetos de una colección de recursos. En todos los casos, existe un formato coherente utilizado, con num\_records indica el número de registros y registros que contienen una matriz de las instancias de objeto. Por ejemplo, puede recuperar los nodos definidos en un clúster específico.

### **Objeto de trabajo**

Si una llamada API se procesa de forma asíncrona, se devuelve un objeto Job que ancla la tarea en segundo plano. Por ejemplo, la solicitud DE REVISIÓN utilizada para actualizar la configuración del clúster se procesa de forma asíncrona y devuelve un objeto Job.

### **Objeto de error**

Si se produce un error, siempre se devuelve un objeto error. Por ejemplo, recibirá un error al intentar cambiar un campo no definido para un clúster.

### **Vacio**

En ciertos casos, no se devuelven datos y el cuerpo de respuesta incluye un objeto JSON vacío.

## **Errores**

Si se produce un error, se devuelve un objeto de error en el cuerpo de respuesta.

### **Formato**

Un objeto de error tiene el siguiente formato:

```
"error": {  
  "message": "<string>",  
  "code": <integer>[,  
  "target": "<string>"]  
}
```

Puede utilizar el valor del código para determinar el tipo o la categoría de error general y el mensaje para determinar el error específico. Si está disponible, el campo de destino incluye la entrada de usuario específica

asociada al error.

### códigos de error comunes

Los códigos de error comunes se describen en la siguiente tabla. Las llamadas API específicas pueden incluir códigos de error adicionales.

Codificación	Descripción
409	Ya existe un objeto con el mismo identificador.
400	El valor de un campo no es válido o falta, o se ha proporcionado un campo adicional.
400	La operación no es compatible.
405	No se puede encontrar un objeto con el identificador especificado.
403	Se deniega el permiso para realizar la solicitud.
409	El recurso está en uso.

## API de REST compatibles

Los recursos disponibles a través de la API DE REST de SnapCenter están organizados en categorías, como se muestra en la página de documentación de API SnapCenter. A continuación se presenta una breve descripción de cada uno de los recursos con las rutas de recursos base, junto con otras consideraciones de uso adicionales si procede.

### Autor

Puede utilizar esta llamada API para iniciar sesión en el servidor SnapCenter. Esta API devuelve un token de autorización de usuario que se utiliza para autenticar solicitudes posteriores.

### Dominios

Puede usar estas llamadas API para realizar lo siguiente:

- recupere todos los dominios
- recuperar detalles de un dominio específico
- registre o cancele el registro de un dominio
- modificar un dominio

### Trabajos

Puede usar estas llamadas API para realizar lo siguiente:

- recuperar todos los trabajos
- recuperar el estado de un trabajo
- cancelar o detener un trabajo

## **Configuración**

Puede usar estas llamadas API para realizar lo siguiente:

- registre, vea, modifique o quite una credencial
- configurar los ajustes de notificación

## **Hosts**

Puede usar estas llamadas API para realizar lo siguiente:

- recuperar los detalles del host
- recupere el plugin instalado y sus detalles de recursos
- añada, quite o modifique un host de plugin
- instale o actualice los plugins

## **Recursos**

Puede usar estas llamadas API para realizar lo siguiente:

- recuperar recursos
- crear, modificar o quitar recursos
- proteger un recurso
- realice un backup, restaure o clone un recurso

## **Completos**

Puede usar estas llamadas API para realizar lo siguiente:

- recupere los detalles de la copia de seguridad
- cambiar el nombre o eliminar backups

## **Clones**

Puede usar estas llamadas API para realizar lo siguiente:

- recuperar detalles de clones
- eliminar clones

## **Clanesplit**

Puede usar estas llamadas API para realizar lo siguiente:

- recupere el estado de una operación de división de clones
- inicie o detenga una operación de división de clones

## **Grupos de recursos**

Puede usar estas llamadas API para realizar lo siguiente:

- recuperar los detalles de un grupo de recursos
- crear, modificar o eliminar un grupo de recursos
- realice un backup de un grupo de recursos

## **Normativas**

Puede usar estas llamadas API para realizar lo siguiente:

- recuperar los detalles de la política
- crear, modificar o eliminar políticas

## **Reducida**

Puede usar estas llamadas API para realizar lo siguiente:

- recupere los detalles de almacenamiento
- crear, modificar o eliminar un almacenamiento
- detectar recursos en un almacenamiento de
- cree o elimine un recurso compartido en el almacenamiento

## **Share**

Puede usar estas llamadas API para realizar lo siguiente:

- recuperar los detalles de un recurso compartido
- cree o elimine un recurso compartido en el almacenamiento

## **Complementos**

Estas llamadas API se pueden usar para recuperar todos los plugins de un host y realizar diferentes operaciones.

## **Leídos**

Puede usar estas llamadas API para realizar lo siguiente:

- generar informes de backup, restauración, clonado y plugin
- agregar, ejecutar, eliminar o modificar programaciones

## **Alertas**

Puede usar estas llamadas API para realizar lo siguiente:

- recupere todas las alertas
- eliminar alertas

## **RBAC**

Puede usar estas llamadas API para realizar lo siguiente:

- recuperar detalles de usuarios, grupos y roles
- añadir usuarios
- crear, modificar o eliminar roles
- asignar o anular la asignación de roles y grupos

## **Configuración**

Puede usar estas llamadas API para realizar lo siguiente:

- ver los ajustes de configuración
- modifique las opciones de configuración

## **CertificadosConfiguración**

Puede usar estas llamadas API para realizar lo siguiente:

- vea el estado del certificado
- modifique la configuración del certificado

## **Repositorio**

Puede usar estas llamadas API para realizar lo siguiente:

- Realice un backup y restaure el repositorio de NSM
- Proteja y desproteja el repositorio NSM
- comutación al respaldo
- Vuelva a generar el repositorio NSM

# **Cómo acceder a las API de REST a través de la página web de API de Swagger**

Las API DE REST se exponen a través de la página web de Swagger. Es posible acceder a la página web de Swagger para mostrar las API DE REST de SnapCenter Server, y también para emitir manualmente una llamada API. Es posible usar la API DE REST para ayudar a gestionar SnapCenter Server o para realizar operaciones de protección de datos.

Debe conocer la dirección IP o el nombre de dominio de gestión de la instancia de SnapCenter Server donde desea ejecutar las API de REST.

No se necesitan permisos especiales para ejecutar el cliente API de REST. Cualquier usuario puede acceder a la página web de Swagger. Los permisos respectivos en los objetos a los que se accede a través de la API DE REST se basan en el usuario que genera el token para iniciar sesión en la API DE REST.

- Pasos\*

1. Desde un explorador, introduzca la URL para acceder a la página web de Swagger con el formato [https://<SnapCenter\\_IP\\_address\\_or\\_name>:<SnapCenter\\_port>/swagger/](https://<SnapCenter_IP_address_or_name>:<SnapCenter_port>/swagger/).



Asegúrese de que la dirección URL de la API DE REST no tenga los caracteres siguientes: +, ., % y &.

2. En el campo **Explore** de Swagger, si la documentación de API de Swagger no se muestra automáticamente, escriba:  
[https://<SnapCenter\\_IP\\_address\\_or\\_name>:<SnapCenter\\_port>/Content/swagger/SnapCenter.yaml](https://<SnapCenter_IP_address_or_name>:<SnapCenter_port>/Content/swagger/SnapCenter.yaml)
3. Haga clic en **explorar**.

Se muestra una lista de los tipos de recursos o categorías de la API.

4. Haga clic en un tipo de recurso de la API para mostrar las API en ese tipo de recurso.

Si se produce un comportamiento inesperado al ejecutar las API DE REST de SnapCenter, puede usar los archivos de registro para identificar la causa del problema y resolverlo. Puede descargar los archivos de registro desde la interfaz de usuario de SnapCenter haciendo clic en **Monitor > Logs > Download**.

## Comience con la API DE REST

Es posible comenzar a usar rápidamente la API de REST de SnapCenter. El acceso a la API ofrece una cierta perspectiva antes de comenzar a utilizarla con los procesos de flujos de trabajo más complejos en una configuración en directo.

### Hola Mundo

Puede ejecutar un comando simple en su sistema para comenzar a utilizar la API DE REST de SnapCenter y confirmar su disponibilidad.

#### Lo que necesitará

- Asegúrese de que la utilidad Curl está disponible en el sistema.
- La dirección IP o el nombre de host del servidor SnapCenter
- Nombre de usuario y contraseña para una cuenta con autoridad para acceder a la API DE REST de SnapCenter.



Si sus credenciales incluyen caracteres especiales, debe formatearlos de una forma que sea aceptable para Curl en función del shell que esté utilizando. Por ejemplo, puede insertar una barra diagonal inversa antes de cada carácter especial o envolver la totalidad `username :password` cadena entre comillas simples.

#### Paso

En la interfaz de línea de comandos de, ejecute lo siguiente para recuperar la información del plugin:

```
curl -X GET -u username:password -k  
"https://<ip_address>/api/hosts?fields=IncludePluginInfo"
```

Ejemplo:

```
curl -X GET -u admin:password -k  
"https://10.225.87.97/api/hosts?fields=IncludePluginInfo"
```

## **Información de copyright**

Copyright © 2025 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.