Administración de Oracle® Solaris 11.1: sistemas de archivos ZFS

Referencia: E36655–05
Septiembre de 2013

ORACLE®

130926@25097

Contenido

Prefacio

La Guía de administración de ZFS de Oracle Solaris 11.1 proporciona información acerca de la configuración y la gestión de los sistemas de archivos ZFS de Oracle Solaris.

Esta guía contiene información para los sistemas basados en SPARC y x86.

Nota – Esta versión de Oracle Solaris es compatible con sistemas que usan arquitecturas de las familias de procesadores SPARC y x86. Los sistemas compatibles aparecen en la *Lista de compatibilidad de hardware de Oracle Solaris* en http://www.oracle.com/webfolder/technetwork/hcl/index.html. Este documento indica las diferencias de implementación entre los tipos de plataforma.

Quién debe utilizar este manual

Esta guía va dirigida a los usuarios interesados en la configuración y administración de los sistemas de archivos ZFS de Oracle Solaris. Es aconsejable tener experiencia previa con el sistema operativo (SO) Oracle Solaris u otra versión de UNIX.

Organización de esta guía

En la tabla siguiente se describen los capítulos de este manual.

Capítulo	Descripción
Capítulo 1, "Sistema de archivos ZFS de Oracle Solaris (introducción)"	Ofrece una descripción general de ZFS, sus características y ventajas. También abarca la terminología y algunos conceptos básicos.
Capítulo 2, "Procedimientos iniciales con Oracle Solaris ZFS "	Ofrece instrucciones paso a paso para configuraciones ZFS sencillas con sistemas de archivos y agrupaciones simples. Este capítulo también brinda instrucciones de hardware y software necesarias para crear sistemas de archivos ZFS.

Capítulo	Descripción
Capítulo 3, "Administración de agrupaciones de almacenamiento de Oracle Solaris ZFS"	Proporciona instrucciones detalladas para crear y administrar agrupaciones de almacenamiento de ZFS.
Capítulo 4, "Gestión de componentes de la agrupación raíz ZFS"	Describe cómo gestionar los componentes de la agrupación raíz ZFS, como configurar una agrupación raíz reflejada, actualizar los entornos de inicio ZFS y ajustar el tamaño de los dispositivos de intercambio y volcado.
Capítulo 5, "Administración de sistemas de archivos ZFS de Oracle Solaris"	Ofrece información detallada sobre la administración de sistemas de archivos ZFS. Abarca conceptos como la disposición jerárquica del sistema de archivos, la herencia de propiedades, la administración de puntos de montaje automático y el modo de compartir interacciones.
Capítulo 6, "Uso de clones e instantáneas de Oracle Solaris ZFS"	Describe cómo crear y administrar clones e instantáneas de ZFS.
Capítulo 7, "Uso de listas de control de acceso y atributos para proteger archivos Oracle Solaris ZFS"	Describe cómo utilizar las listas de control de acceso (ACL) para proteger los archivos ZFS ofreciendo más permisos granulares que los UNIX estándar.
Capítulo 8, "Administración delegada de ZFS Oracle Solaris"	Describe la forma de utilizar la administración delegada de ZFS para permitir que los usuarios sin privilegios puedan efectuar tareas de administración de ZFS.
Capítulo 9, "Temas avanzados de Oracle Solaris ZFS"	Ofrece información sobre el uso de volúmenes de ZFS, el uso de ZFS en un sistema Oracle Solaris con zonas instaladas y agrupaciones raíz alternativas.
Capítulo 10, "Recuperación de agrupaciones y solución de problemas de Oracle Solaris ZFS"	Describe cómo identificar errores de ZFS y cómo resolverlos. También se proporciona pasos para evitar errores.
Capítulo 11, "Archivado de instantáneas y recuperación de agrupaciones raíz"	Describe cómo archivar las instantáneas de la agrupación raíz y cómo recuperar la agrupación raíz.
Capítulo 12, "Prácticas de ZFS recomendadas por Oracle Solaris"	Describe las prácticas recomendadas para crear, supervisar y mantener las agrupaciones de almacenamiento y los sistemas de archivos de ZFS.
Apéndice A, "Descripciones de versiones de Oracle Solaris ZFS"	Describe versiones de ZFS disponibles, las características de cada versión y el sistema operativo Solaris pertinente.

Manuales relacionados

Se puede encontrar información relacionada con temas generales de administración del sistema Oracle Solaris en los manuales siguientes:

- *Gestión del rendimiento, los procesos y la información del sistema en Oracle Solaris 11.1*
- *Gestión de las cuentas de usuario y los entornos de usuario en Oracle Solaris 11.1*
- *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*
- *Administración de Oracle Solaris 11.1: servicios de seguridad*

Acceso a My Oracle Support

Los clientes de Oracle tienen acceso a soporte electrónico por medio de My Oracle Support. Para obtener más información, visite `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` o, si tiene alguna discapacidad auditiva, visite `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs`.

Convenciones tipográficas

La siguiente tabla describe las convenciones tipográficas utilizadas en este manual.

TABLA P–1 Convenciones tipográficas

Tipos de letra	Descripción	Ejemplo
`AaBbCc123`	Los nombres de los comandos, los archivos, los directorios y los resultados que el equipo muestra en pantalla	Edite el archivo `.login`. Utilice el comando `ls -a` para mostrar todos los archivos. `nombre_sistema% tiene correo.`
`AaBbCc123`	Lo que se escribe, en contraposición con la salida del equipo en pantalla	`nombre_sistema% su` `Contraseña:`
aabbcc123	Marcador de posición: sustituir por un valor o nombre real	El comando necesario para eliminar un archivo es `rm` *filename*.

TABLA P–1 Convenciones tipográficas *(Continuación)*

Tipos de letra	Descripción	Ejemplo
AaBbCc123	Títulos de los manuales, términos nuevos y palabras destacables	Consulte el capítulo 6 de la *Guía del usuario*. Una *copia en caché* es aquella que se almacena localmente. *No* guarde el archivo. **Nota:** algunos elementos destacados aparecen en negrita en línea.

Indicadores de los shells en los ejemplos de comandos

La tabla siguiente muestra los indicadores de sistema UNIX y los indicadores de superusuario para shells incluidos en el sistema operativo Oracle Solaris. En los ejemplos de comandos, el indicador de shell muestra si el comando debe ser ejecutado por un usuario normal o un usuario con privilegios.

TABLA P–2 Indicadores de shell

Shell	Indicador
Shell Bash, shell Korn y shell Bourne	`$`
Shell Bash, shell Korn y shell Bourne para superusuario	`#`
Shell C	`machine_name%`
Shell C para superusuario	`machine_name#`

◆ ◆ ◆ CAPÍTULO 1

Sistema de archivos ZFS de Oracle Solaris (introducción)

Este capítulo ofrece una visión general del sistema de archivos ZFS de Oracle Solaris, así como de sus funciones y ventajas. También aborda terminología básica utilizada en el resto del manual.

Este capítulo se divide en las secciones siguientes:

Novedades de ZFS

Esta sección resume las funciones nuevas del sistema de archivos ZFS.

Mensajes mejorados sobre dispositivos de agrupación ZFS

Oracle Solaris 11.1: el comando `zpool status` se ha mejorado para proporcionar información más detallada sobre errores de dispositivos. En este ejemplo, la salida de `zpool status` identifica un dispositivo de agrupación (`c0t5000C500335F907Fd0`) que tiene el estado `UNAVAIL` debido a errores persistentes, y debe reemplazarse.

```
# zpool status -v pond
  pool: pond
 state: DEGRADED
status: One or more devices are unavailable in response to persistent errors.
        Sufficient replicas exist for the pool to continue functioning in a
        degraded state.
action: Determine if the device needs to be replaced, and clear the errors
        using 'zpool clear' or 'fmadm repaired', or replace the device
        with 'zpool replace'.
  scan: scrub repaired 0 in 0h0m with 0 errors on Wed Jun 20 15:38:08 2012
config:

        NAME                       STATE     READ WRITE CKSUM
        pond                       DEGRADED     0     0     0
          mirror-0                 DEGRADED     0     0     0
            c0t5000C500335F95E3d0  ONLINE       0     0     0
            c0t5000C500335F907Fd0  UNAVAIL      0     0     0
          mirror-1                 ONLINE       0     0     0
            c0t5000C500335BD117d0  ONLINE       0     0     0
            c0t5000C500335DC60Fd0  ONLINE       0     0     0

device details:

        c0t5000C500335F907Fd0    UNAVAIL          cannot open
        status: ZFS detected errors on this device.
                The device was missing.
           see: http://support.oracle.com/msg/ZFS-8000-LR for recovery


errors: No known data errors
```

Mejoras en el uso compartido de archivos ZFS

Oracle Solaris 11.1: el uso compartido de sistemas de archivos ZFS es más eficaz gracias a las siguientes mejoras principales:

- Se simplificó la sintaxis del recurso compartido. Puede compartir un sistema de archivos configurando la nueva propiedad `share.nfs` o `share.smb`
- Mejor herencia de propiedades de recursos compartidos para sistemas de archivos descendentes

Las mejoras de uso compartido de archivos están asociadas con la versión de agrupación 34.

Para obtener más información, consulte "Cómo compartir y anular la compartición de sistemas de archivos ZFS" en la página 188.

Sistema de archivos var compartido

Oracle Solaris 11.1: la instalación de Oracle Solaris 11.1 crea automáticamente un sistema de archivos `rpool/VARSHARE` montado en `/var/share`. La finalidad de este sistema de archivos es compartir sistemas de archivos entre entornos de inicio de modo de reducir la cantidad de espacio necesaria en el directorio `/var` para todos los entornos de inicio. Por ejemplo:

```
# ls /var/share
audit  cores  crash  mail
```

Se crean automáticamente enlaces simbólicos de `/var` a los componentes `/var/share` antes enumerados por motivos de compatibilidad. Este sistema de archivos, por lo general, no requiere administración, excepto para garantizar que los componentes de `/var` no completen el sistema de archivos raíz.

Si un sistema Oracle Solaris 11 se actualiza a Oracle Solaris 11.1, la migración de datos del directorio `/var` original al directorio `/var/share` posiblemente tarde un tiempo.

Compatibilidad de inicio para discos con etiqueta EFI (GPT)

Oracle Solaris 11.1: esta versión instala una etiqueta de disco EFI (GPT) en un disco de agrupación raíz ZFS para un sistema basado en x86, en la mayoría de los casos. Por ejemplo:

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: none requested
config:
```

```
    NAME      STATE     READ WRITE CKSUM
    rpool     ONLINE       0     0     0
      c2t0d0  ONLINE       0     0     0

errors: No known data errors
```

- La instalación de Oracle Solaris 11.1 en un sistema basado en SPARC con firmware compatible con GPT o en un sistema basado en x86 aplica una etiqueta de disco GPT en el disco de agrupación raíz que usa todo el disco. Para sistemas basados en SPARC que admiten un disco de inicio con etiqueta GPT, consulte las notas de la versión de Oracle Solaris 11.1 para obtener información sobre la aplicación de firmware compatible con GPT. De lo contrario, la instalación de Oracle Solaris 11.1 en un sistema SPARC aplica una etiqueta SMI (VTOC) al disco de agrupación raíz con un único segmento 0.
- En la mayoría de los casos, un sistema basado en x86 se instala con un disco con etiqueta EFI (GPT)
- El comando `zpool` se ha mejorado para admitir una etiqueta de disco EFI (GPT), de modo que si necesita volver a crear una agrupación root después de la instalación del sistema, puede hacerlo mediante el comando `zpool create -B`. Esta nueva opción del comando crea las particiones requeridas y la información necesaria para el inicio. Para obtener más información sobre la creación de una agrupación raíz después de la instalación, consulte "Cómo crear un entorno de inicio en otra agrupación raíz (SPARC o x86/VTOC)" en la página 132.

  Si necesita reemplazar un disco en una agrupación raíz con una etiqueta EFI (GPT), deberá ejecutar este comando después de la operación `zpool replace`.

  ```
  # bootadm install-bootloader
  ```
- Las instalaciones de Oracle Solaris ya no se limitan a los primeros 2 TiB del disco en un sistema basado en x86.

Mejoras en el uso de comandos de ZFS

Oracle Solaris 11: los comandos `zfs` y `zpool` tienen un subcomando `help` que se puede utilizar para proporcionar más información sobre los subcomandos `zfs` y `zpool`, y sus opciones admitidas. Por ejemplo:

```
# zfs help
The following commands are supported:
allow       clone       create      destroy     diff        get
groupspace  help        hold        holds       inherit     list
mount       promote     receive     release     rename      rollback
send        set         share       snapshot    unallow     unmount
unshare     upgrade     userspace
For more info, run: zfs help <command>
# zfs help create
usage:
        create [-p] [-o property=value] ... <filesystem>
        create [-ps] [-b blocksize] [-o property=value] ... -V <size> <volume>
```

```
# zpool help
The following commands are supported:
add      attach   clear    create   destroy  detach   export   get
help     history  import   iostat   list     offline  online   remove
replace  scrub    set      split    status   upgrade
For more info, run: zpool help <command>
# zpool help attach
usage:
        attach [-f] <pool> <device> <new-device>
```

Para obtener más información, consulte `zfs`(1M) and `zpool`(1M).

Mejoras de instantáneas de ZFS

Oracle Solaris 11: esta versión incluye las siguientes mejoras de instantáneas de ZFS:

- El comando `zfs snapshot` tiene un alias `snap` que proporciona una sintaxis abreviada para este comando. Por ejemplo:

  ```
  # zfs snap -r users/home@snap1
  ```

- El comando `zfs diff` proporciona una opción de enumeración, `-e`, para identificar todos los archivos agregados o modificados entre las dos instantáneas. La salida generada identifica todos los archivos agregados, pero no proporciona posibles supresiones. Por ejemplo:

  ```
  # zfs diff -e tank/cindy@yesterday tank/cindy@now
  +       /tank/cindy/
  +       /tank/cindy/file.1
  ```

  También puede utilizar la opción `-o` para identificar campos seleccionados para visualizar. Por ejemplo:

  ```
  # zfs diff -e -o size -o name tank/cindy@yesterday tank/cindy@now
  +       7       /tank/cindy/
  +       206695  /tank/cindy/file.1
  ```

Para obtener más información sobre la creación de instantáneas de ZFS, consulte el Capítulo 6, “Uso de clones e instantáneas de Oracle Solaris ZFS”.

Cambio de página de manual de ZFS (`zfs.1m`)

Oracle Solaris 11: la página del manual `zfs.1m` fue revisada para que las funciones principales del sistema de archivos ZFS permanezcan en la página `zfs.1m`, pero la administración delegada, el cifrado y los ejemplos y el uso compartido de sintaxis se tratan en las siguientes páginas:

- `zfs_allow`(1M)
- `zfs_encrypt`(1M)
- `zfs_share`(1M)

Propiedad aclmode mejorada

Oracle Solaris 11: la propiedad `aclmode` modifica el comportamiento de la lista de control de acceso (ACL) cuando se modifican los permisos de ACL de un archivo durante una operación `chmod`. La propiedad `aclmode` se ha vuelto a introducir con los siguientes valores de propiedad:

- `discard`: un sistema de archivos con una propiedad `aclmode` de `discard` suprime todas las entradas de ACL que no representan el modo del archivo. Éste es el valor predeterminado.
- `mask`: un sistema de archivos con una propiedad `aclmode` de `mask` reduce los permisos de usuario o de grupo. Se reducen los permisos para que no superen los bits de permisos de grupo, a menos que se trate de una entrada de usuario cuyo UID sea igual al del propietario del archivo o directorio. Así, los permisos de ACL se reducen para que no superen los bits de permisos del propietario. El valor de máscara también conserva la ACL cuando cambian los modos, siempre que no se haya realizado una operación de conjunto de ACL explícita.
- `passthrough`: un sistema de archivos con una propiedad `aclmode` de `passthrough` indica que no se realizaron más cambios en la ACL aparte de generar las entradas necesarias de ACL para representar el nuevo modo del archivo o del directorio.

Para obtener más información, consulte el Ejemplo 7–14.

Identificación de dispositivos de agrupaciones por ubicación física

Oracle Solaris 11: en esta versión de Solaris, utilice el comando `zpool status -l` para mostrar la información de ubicación física del disco para dispositivos de la agrupación, que se encuentra disponible desde el directorio `/dev/chassis`. Este directorio contiene valores de chasis, receptáculo y ocupante para los dispositivos del sistema.

Además, puede utilizar el comando `fmadm add-alias` para incluir un nombre de alias de disco que lo ayude a identificar la ubicación física de los discos en su entorno. Por ejemplo:

```
# fmadm add-alias SUN-Storage-J4400.0912QAJ001 SUN-Storage-J4400.rack22
```

Por ejemplo:

```
% zpool status -l export
  pool: export
 state: ONLINE
  scan: resilvered 492G in 8h22m with 0 errors on Wed Aug  1 17:22:11 2012
config:

 NAME                                                    STATE  READ WRITE CKSUM
   export                                                ONLINE    0     0     0
     mirror-0                                            ONLINE    0     0     0
       /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__2/disk  ONLINE    0     0     0
       /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__3/disk  ONLINE    0     0     0
```

```
    mirror-1                                                ONLINE    0     0     0
      /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__4/disk  ONLINE    0     0     0
      /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__5/disk  ONLINE    0     0     0
    mirror-2                                                ONLINE    0     0     0
      /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__6/disk  ONLINE    0     0     0
      /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__7/disk  ONLINE    0     0     0
    mirror-3                                                ONLINE    0     0     0
      /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__8/disk  ONLINE    0     0     0
      /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__9/disk  ONLINE    0     0     0
    mirror-4                                                ONLINE    0     0     0
      /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__10/disk ONLINE    0     0     0
      /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__11/disk ONLINE    0     0     0
    spares
      /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__0/disk  AVAIL
      /dev/chassis/SUN-Storage-J4400.rack22/SCSI_Device__1/disk  AVAIL

errors: No known data errors
```

El comando `zpool iostat` también se ha actualizado con el fin de proporcionar información de ubicación física para los dispositivos de una agrupación.

Además, los comandos `diskinfo`, `format` y `prtconf` también proporcionan información sobre la ubicación física del disco. Para obtener más información, consulte `diskinfo(1M)`.

Migración de ZFS shadow

Oracle Solaris 11: en esta versión, puede migrar datos de un sistema de archivos antiguo a un sistema de archivos nuevo, mientras permite el acceso y la modificación del sistema de archivos nuevo durante el proceso de migración.

La configuración de la propiedad shadow en un sistema de archivos ZFS nuevo activa la migración de los datos más antiguos. La propiedad shadow se puede establecer para migrar datos del sistema local o un sistema remoto con cualquiera de los siguientes valores:

`file:///`*path*
`nfs://`*host*:*path*

Para obtener más información, consulte "Migración de sistemas de archivos ZFS" en la página 213.

Cifrado del sistema de archivos ZFS

Oracle Solaris 11: en esta versión, puede cifrar un sistema de archivos ZFS.

Por ejemplo, el sistema de archivos `tank/cindy` se crea con la propiedad de cifrado activada. La política de cifrado predeterminada debe proporcionar una frase de contraseña, que debe tener un mínimo de 8 caracteres de longitud.

```
# zfs create -o encryption=on tank/cindy
Enter passphrase for 'tank/cindy': xxx
Enter again: xxx
```

Una política de cifrado se establece cuando se crea un sistema de archivos ZFS. La política de cifrado de un sistema de archivos es heredada por sistemas de archivos descendientes y no se puede eliminar.

Para obtener más información, consulte "Cifrado de sistemas de archivos ZFS" en la página 205.

Mejoras en el flujo de envío de ZFS

Oracle Solaris 11: en esta versión, se pueden establecer las propiedades del sistema de archivos que se envían y se reciben en un flujo de instantáneas. Estas mejoras proporcionan flexibilidad al aplicar las propiedades del sistema de archivos en un flujo de envío al sistema de archivos receptor o al determinar si las propiedades del sistema de archivos local, como el valor de propiedad `mountpoint`, se deben ignorar cuando se reciban.

Para obtener más información, consulte "Aplicación de valores de propiedad diferentes a un flujo de instantáneas de ZFS" en la página 234.

Diferencias entre instantáneas de ZFS (`zfs diff`)

Oracle Solaris 11: en esta versión, se pueden determinar las diferencias entre instantáneas de ZFS mediante el comando `zfs diff`.

Por ejemplo, considere que se crean las siguientes dos instantáneas:

```
$ ls /tank/cindy
fileA
$ zfs snapshot tank/cindy@0913
$ ls /tank/cindy
fileA  fileB
$ zfs snapshot tank/cindy@0914
```

Por ejemplo, para identificar las diferencias que existen entre dos instantáneas, utilice una sintaxis similar a la siguiente:

```
$ zfs diff tank/cindy@0913 tank/cindy@0914
M       /tank/cindy/
+       /tank/cindy/fileB
```

En la salida anterior, `M` indica que el directorio se ha modificado. El símbolo + indica que `fileB` existe en la instantánea posterior.

Para obtener más información, consulte "Identificación de diferencias entre instantáneas de ZFS (`zfs diff`)" en la página 225.

Mejoras en el rendimiento y la recuperación de agrupaciones de almacenamiento ZFS

Oracle Solaris 11: en esta versión, se proporcionan las siguientes funciones nuevas de agrupación de almacenamiento ZFS:

- Puede importar una agrupación con un registro faltante usando el comando `zpool import -m`. Para obtener más información, consulte "Importación de una agrupación a la que le falta un dispositivo de registro" en la página 111.
- Puede importar una agrupación en el modo de sólo lectura. Esta función está diseñada, principalmente, para la recuperación de agrupaciones. Si no se puede acceder a una agrupación dañada debido a que los dispositivos subyacentes están dañados, puede importar la agrupación de sólo lectura para recuperar los datos. Para obtener más información, consulte "Importación de una agrupación en modo de sólo lectura" en la página 113.
- Una agrupación de almacenamiento RAID-Z (`raidz1`, `raidz2` o `raidz3`) creada en esta versión tendrá algunos metadatos sensibles a la latencia reflejados automáticamente para mejorar el rendimiento del procesamiento de lectura de E/S. En el caso de las agrupaciones RAID-Z existentes que se actualicen, al menos, a la versión 29, se reflejarán algunos metadatos para todos los datos escritos recientemente.

  Los metadatos reflejados en una agrupación RAID-Z *no* ofrecen protección adicional contra fallos de hardware, algo similar a lo que ofrece una agrupación de almacenamiento reflejada. Los metadatos reflejados utilizan más espacio, pero la protección de RAID-Z sigue siendo la misma que en las versiones anteriores. Esta mejora sólo tiene como objetivo el rendimiento.

Ajuste del comportamiento síncrono de ZFS

Oracle Solaris 11: en esta versión, puede determinar el comportamiento síncrono de un sistema de archivos ZFS mediante la propiedad `sync`.

El comportamiento síncrono predeterminado consiste en escribir todas las transacciones síncronas del sistema de archivos en el registro de intención y vaciar todos los dispositivos para garantizar que los datos estén estables. No se recomienda la desactivación del comportamiento síncrono predeterminado. Es posible que las aplicaciones que dependen de la compatibilidad síncrona resulten afectadas y que los datos se pierdan.

La propiedad `sync` se puede establecer antes o después de la creación del sistema de archivos. En cualquier caso, el valor de la propiedad se aplica inmediatamente. Por ejemplo:

```
# zfs set sync=always tank/neil
```

El parámetro `zil_disable` ya no está disponible en las versiones de Oracle Solaris que incluyen la propiedad `sync`.

Para obtener más información, consulte la Tabla 5–1.

Mensajes de agrupación ZFS mejorados

Oracle Solaris 11: en esta versión, se puede utilizar la opción `-T` para asignar un intervalo y un valor de recuento para que los comandos `zpool list` y `zpool status` muestren información adicional.

Además, el comando `zpool status` proporciona información sobre la reconstrucción y la limpieza de datos de la agrupación de la siguiente manera:

- Informe de reconstrucción en curso. Por ejemplo:

```
scan: resilver in progress since Thu Jun  7 14:41:11 2012
    3.83G scanned out of 73.3G at 106M/s, 0h11m to go
    3.80G resilvered, 5.22% done
```

- Informe de limpieza en curso. Por ejemplo:

```
scan: scrub in progress since Thu Jun  7 14:59:25 2012
    1.95G scanned out of 73.3G at 118M/s, 0h10m to go
    0 repaired, 2.66% done
```

- Mensaje de reconstrucción finalizada. Por ejemplo:

```
resilvered 73.3G in 0h13m with 0 errors on Thu Jun  7 14:54:16 2012
```

- Mensaje de limpieza finalizada. Por ejemplo:

```
scan: scrub repaired 512B in 1h2m with 0 errors on Thu Jun  7 15:10:32 2012
```

- Mensaje de cancelación de limpieza en curso. Por ejemplo:

```
scan: scrub canceled on Thu Jun  7 15:19:20 MDT 2012
```

- Los mensajes de finalización de limpieza y reconstrucción se mantienen durante los reinicios del sistema.

La sintaxis siguiente utiliza el intervalo y la opción de recuento para mostrar la información de la reconstrucción de la agrupación en curso. Puede utilizar el valor `-T d` para mostrar la información en formato de fecha estándar o el valor `-T u` para mostrar la información en un formato interno.

```
# zpool status -T d tank 3 2
Thu Jun 14 14:08:21 MDT 2012

  pool: tank
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Thu Jun 14 14:08:05 2012
    2.96G scanned out of 4.19G at 189M/s, 0h0m to go
    1.48G resilvered, 70.60% done
```

```
config:

        NAME                      STATE     READ WRITE CKSUM
        tank                      DEGRADED     0     0     0
          mirror-0                ONLINE       0     0     0
            c0t5000C500335F95E3d0 ONLINE       0     0     0
            c0t5000C500335F907Fd0 ONLINE       0     0     0
          mirror-1                DEGRADED     0     0     0
            c0t5000C500335BD117d0 ONLINE       0     0     0
            c0t5000C500335DC60Fd0 DEGRADED     0     0     0  (resilvering)

errors: No known data errors
```

Mejoras en la interoperabilidad de las ACL de ZFS

Oracle Solaris 11: en esta versión, se ofrecen las siguientes mejoras en las ACL:

- Las ACL triviales no requieren entradas de control de acceso (ACE) `deny`, salvo los permisos poco comunes. Por ejemplo, un modo `0644`, `0755` o `0664` no requiere ACE `deny`, pero un modo como `0705`, `0060`, etc. requiere ACE `deny`.

  El comportamiento anterior incluye entradas de control de acceso `deny` en ACL triviales, como `644`. Por ejemplo:

  ```
  # ls -v file.1
  -rw-r--r--   1 root     root      206663 Jun 14 11:52 file.1
       0:owner@:execute:deny
       1:owner@:read_data/write_data/append_data/write_xattr/write_attributes
           /write_acl/write_owner:allow
       2:group@:write_data/append_data/execute:deny
       3:group@:read_data:allow
       4:everyone@:write_data/append_data/write_xattr/execute/write_attributes
           /write_acl/write_owner:deny
       5:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
           :allow
  ```

  El nuevo comportamiento para una ACL trivial, como `644`, no incluye la opción de entradas de control de acceso `deny`. Por ejemplo:

  ```
  # ls -v file.1
  -rw-r--r--   1 root     root      206663 Jun 22 14:30 file.1
       0:owner@:read_data/write_data/append_data/read_xattr/write_xattr
           /read_attributes/write_attributes/read_acl/write_acl/write_owner
           /synchronize:allow
       1:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
       2:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
           :allow
  ```

- Las ACL ya no se dividen en varias entradas de control de acceso durante la herencia para tratar de preservar el permiso original sin modificaciones. En cambio, los permisos se modifican según resulta necesario para aplicar el modo de creación de archivos.
- El comportamiento de la propiedad `aclinherit` incluye una reducción de los permisos cuando la propiedad se configura como `restricted`, lo que implica que las ACL ya no se dividen en varias entradas de control de acceso durante la herencia.

- Una nueva regla de cálculo del modo de permiso especifica que si una ACL tiene una entrada de control de acceso de usuario (*user*) que coincide con el propietario del archivo, dichos permisos se incluyen en el cálculo del modo de permiso. La misma regla se aplica si una entrada de control de acceso de grupo (*group*) coincide con el propietario del grupo del archivo.

Para obtener más información, consulte Capítulo 7, "Uso de listas de control de acceso y atributos para proteger archivos Oracle Solaris ZFS".

División de una agrupación de almacenamiento de ZFS refleja (`zpool split`)

Oracle Solaris 11: en esta versión, se puede utilizar el comando `zpool split` para dividir una agrupación de almacenamiento reflejada, que desconecta discos de la agrupación reflejada original para crear otra agrupación idéntica.

Para obtener más información, consulte "Creación de una nueva agrupación mediante la división de una agrupación de almacenamiento de ZFS reflejada" en la página 75.

Cambios de iSCSI de ZFS

Oracle Solaris 11: en esta versión, el daemon de destino iSCSI se sustituye mediante el uso del daemon de destino Common Multiprotocol SCSI Target (COMSTAR). Este cambio también significa que la propiedad `shareiscsi` que se utilizó para compartir un volumen de ZFS como un LUN de iSCSI ya no está disponible. Utilice el comando `stmfadm` para configurar y compartir un volumen de ZFS como un LUN de iSCSI.

Para obtener más información, consulte "Uso de un volumen de ZFS como un LUN iSCSI" en la página 287.

Nuevo proceso del sistema ZFS

Oracle Solaris 11: en esta versión, cada agrupación de almacenamiento de ZFS tiene un `zpool-`*nombredeagrupación* asociado con el proceso. Los subprocesos de este proceso son los del procesamiento de E/S de la agrupación para manejar las tareas de E/S, como la validación de la suma de comprobación y la compresión, que están asociadas con la agrupación. La finalidad de este proceso es proporcionar visibilidad en cada uso de la CPU del grupo de almacenamiento.

Mediante los comandos `ps` y `prstat` se puede obtener información sobre los procesos en ejecución. Dichos procesos sólo están disponibles en la zona global. Para obtener más información, consulte `SDC(7)`.

Propiedad de eliminación de datos duplicados de ZFS

Oracle Solaris 11: en esta versión, puede utilizar la propiedad de eliminación de datos duplicados (`dedup`) para eliminar datos redundantes de sus sistemas de archivos ZFS. Si un sistema de archivos tiene activada la propiedad `dedup`, los bloques de datos duplicados se eliminan de forma sincrónica. El resultado es que se almacenan solamente los datos exclusivos y los componentes comunes se comparten entre archivos.

Puede activar esta propiedad como se indica a continuación:

```
# zfs set dedup=on tank/home
```

Aunque la eliminación de datos duplicados se establece como una propiedad del sistema de archivos, el alcance se extiende a todas las agrupaciones. Por ejemplo, se puede identificar la relación de eliminación de datos duplicados como se indica a continuación:

```
# zpool list tank
NAME  SIZE  ALLOC  FREE  CAP  DEDUP  HEALTH  ALTROOT
tank  556G  4.19G  552G   0%  1.00x  ONLINE  -
```

La salida `zpool list` se ha actualizado para que sea compatible con la propiedad de eliminación de datos duplicados.

Para obtener más información sobre la configuración de la propiedad de eliminación de datos duplicados, consulte "Propiedad dedup" en la página 172.

No active la propiedad `dedup` de los sistemas de archivos que residen en sistemas de producción hasta que se revisen las siguientes consideraciones:

- Determinar si los datos se beneficiarían con ahorros de espacio de eliminación de datos duplicados.
- Determinar si el sistema tiene suficiente memoria física para admitir la eliminación de datos duplicados.
- Posibles impactos en el rendimiento del sistema.

Para obtener más información sobre estas consideraciones, consulte "Propiedad dedup" en la página 172.

¿Qué es Oracle Solaris ZFS?

El sistema de archivos ZFS de Oracle Solaris es un sistema de archivos que aporta una forma totalmente distinta de administrar sistemas de archivos, con funciones y ventajas que no hay en ningún otro sistema de archivos actual. ZFS es sólido, escalable y fácil de administrar.

Almacenamiento en grupos de ZFS

ZFS se basa en el concepto de *grupos de almacenamiento* para administrar el almacenamiento físico. Desde siempre, los sistemas de archivos se estructuran a partir de un solo único dispositivo físico. Para poder ocuparse de varios dispositivos y ofrecer redundancia de datos, se incorporó el concepto del *administrador de volúmenes*, con el fin de ofrecer una representación de un único dispositivo y evitar que los sistemas de archivos tuvieran que modificarse para aprovechar las ventajas de varios dispositivos. Este diseño significaba otro nivel de complejidad y obstaculizaba determinados avances en los sistemas de archivos, al carecer de control sobre la ubicación física de los datos en los volúmenes virtualizados

ZFS elimina del todo la administración de volúmenes. En vez de tener que crear volúmenes virtualizados, ZFS agrega dispositivos a una agrupación de almacenamiento. La agrupación de almacenamiento describe las características físicas del almacenamiento (organización del dispositivo, redundancia de datos, etc.) y actúa como almacén de datos arbitrario en el que se pueden crear sistemas de archivos. Los sistemas de archivos ya se limitan a dispositivos individuales y les permite compartir espacio en el disco con todos los sistemas de archivos de la agrupación. Ya no es necesario predeterminar el tamaño de un sistema de archivos, ya que el tamaño de los sistemas de archivos crece automáticamente en el espacio asignado a la agrupación de almacenamiento. Al incorporar un nuevo almacenamiento, todos los sistemas de archivos de la agrupación pueden usar de inmediato el espacio en el disco adicional sin procesos complementarios. En muchos sentidos, la agrupación de almacenamiento funciona del mismo modo que un sistema de memoria virtual: si se agrega al sistema un módulo de memoria DIMM, el sistema operativo no obliga a ejecutar comandos para configurar la memoria y asignarla a los procesos individuales. Todos los procesos del sistema utilizan automáticamente la memoria adicional.

Semántica transaccional

ZFS es un sistema de archivos transaccional. Ello significa que el estado del sistema de archivos siempre es coherente en el disco. Los sistemas de archivos tradicionales sobrescriben datos in situ. Esto significa que, si el equipo se queda sin alimentación (por ejemplo, entre el momento en que un bloque de datos se asigna y cuando se vincula a un directorio), el sistema de archivos se queda en un estado incoherente. En el pasado, este problema se solucionaba mediante el comando `fsck`. Este comando verificaba el estado del sistema de archivos e intentaba reparar cualquier incoherencia durante el proceso. Este problema de sistemas de archivos incoherentes daba muchos quebraderos de cabeza a los administradores y el comando `fsck` nunca garantizaba la solución a todos los problemas. Posteriormente, los sistemas de archivos han incorporado el concepto de *registro de diario*. *El registro de diario guarda las acciones en un diario aparte, el cual se puede volver a reproducir con seguridad si el sistema se bloquea.* Este proceso supone cargas innecesarias, porque los datos se deben escribir dos veces y a menudo provoca una nueva fuente de problemas (como no poder volver a reproducir correctamente el registro de diario).

Con un sistema de archivos transaccional, los datos se administran mediante la semántica *copia por escritura.* Los datos nunca se sobrescriben y ninguna secuencia de operaciones se compromete ni se ignora por completo. Este mecanismo hace que el sistema de archivos nunca pueda dañarse por una interrupción imprevista de la alimentación o un bloqueo del sistema. Aunque pueden perderse fragmentos de datos escritos más recientemente, el propio sistema de archivos siempre será coherente. Asimismo, siempre se garantiza que los datos sincrónicos (escritos mediante el indicador `O_DSYNC`) se escriban antes de la devolución, por lo que nunca se pierden.

Datos de reparación automática y sumas de comprobación

En ZFS se verifican todos los datos y metadatos mediante un algoritmo de suma de comprobación seleccionable por el usuario. Los sistemas de archivos tradicionales con suma de comprobación la efectúan por bloques obligatoriamente debido a la capa de administración de volúmenes y la disposición del sistema de archivos tradicional. El diseño tradicional significa que algunos errores, como la escritura de un bloque completo en una ubicación incorrecta, pueden hacer que los datos no sean correctos, pero no producen errores de suma de comprobación. Las sumas de comprobación de ZFS se almacenan de forma que estos errores se detecten y haya una recuperación eficaz. La suma de comprobación y la recuperación de datos se efectúan en la capa del sistema de archivos y son transparentes para las aplicaciones.

Asimismo, ZFS ofrece soluciones para la reparación automática de datos. ZFS admite agrupaciones de almacenamiento con diversos niveles de redundancia de datos. Si se detecta un bloque de datos incorrectos, ZFS recupera los datos correctos de otra copia redundante y repara los datos incorrectos al sustituirlos por una copia correcta.

Escalabilidad incomparable

Un elemento de diseño clave en el sistema de archivos ZFS es la escalabilidad. El sistema de archivos es de 128 bits y permite 256 trillones de zettabytes de almacenamiento. Todos los metadatos se asignan de forma dinámica, con lo que no hace falta asignar previamente inodes ni limitar la escalabilidad del sistema de archivos cuando se crea. Todos los algoritmos se han escrito teniendo en cuenta la escalabilidad. Los directorios pueden tener hasta 2^{48} (256 billones) de entradas; no existe un límite para el número de sistemas de archivos o de archivos que puede haber en un sistema de archivos.

Instantáneas de ZFS

Una *instantánea* es una copia de sólo lectura de un sistema de archivos o volumen. Las instantáneas se crean rápida y fácilmente. Inicialmente, las instantáneas no consumen espacio adicional en el disco dentro de la agrupación.

Como los datos de un conjunto de datos activo cambian, la instantánea consume espacio en el disco al seguir haciendo referencia a los datos antiguos. Como resultado, la instantánea impide que los datos pasen al grupo.

Administración simplificada

Uno de los aspectos más destacados de ZFS es su modelo de administración muy simplificado. Mediante un sistema de archivos con distribución jerárquica, herencia de propiedades y administración automática de puntos de montaje y semántica share de NFS, el ZFS facilita la creación y gestión de sistemas de archivos sin tener que usar varios comandos ni editar archivos de configuración. Con un solo comando puede establecer fácilmente cuotas o reservas, activar o desactivar la compresión, o administrar puntos de montaje para diversos sistemas de archivos. Puede examinar o sustituir dispositivos sin aprender un conjunto independiente de comandos de administrador de volúmenes. Puede enviar y recibir flujos de instantáneas del sistema de archivos .

ZFS administra los sistemas de archivos a través de una jerarquía que permite la administración simplificada de propiedades como cuotas, reservas, compresión y puntos de montaje. En este modelo, los sistemas de archivos se convierten en el punto central de control. Los sistemas de archivos son muy sencillos (equivalen a un nuevo directorio), por lo que se recomienda crear un sistema de archivos para cada usuario, proyecto, espacio de trabajo, etc. Este diseño permite definir los puntos de administración de forma detallada.

Terminología de ZFS

Esta sección describe la terminología básica utilizada en este manual:

entorno de inicio	Un entorno de inicio es un entorno de Oracle Solaris que se puede iniciar y está formado por un sistema de archivos raíz ZFS y, opcionalmente, por otros sistemas de archivos montados debajo de éste. No puede haber más de un entorno de inicio activo al mismo tiempo.
suma de comprobación	Cifrado de 256 bits de los datos en un bloque del sistema de archivos. La suma de comprobación puede ir de la rápida y sencilla fletcher4 (valor predeterminado) a cifrados criptográficamente complejos como SHA256.
clónico	Sistema de archivos cuyo contenido inicial es idéntico al de una instantánea. Para obtener más información sobre clones, consulte "Información general sobre clones de ZFS" en la página 226.

conjunto de datos

Nombre genérico de las entidades ZFS siguientes: clones, sistemas de archivos, instantáneas y volúmenes.

Cada conjunto de datos se identifica mediante un nombre exclusivo en el espacio de nombres de ZFS. Los conjuntos de datos se identifican mediante el formato siguiente:

agrupación/*ruta*[@*instantánea*]

agrupación	Identifica el nombre de la agrupación de almacenamiento que contiene el conjunto de datos
ruta	Nombre de ruta delimitado por barras para el componente del conjunto de datos
instantánea	Componente opcional que identifica una instantánea de un conjunto de datos

Para obtener más información sobre conjuntos de datos, consulte el Capítulo 5, "Administración de sistemas de archivos ZFS de Oracle Solaris".

sistema de archivos

Conjunto de datos de ZFS del tipo `filesystem` que se monta en el espacio de nombre del sistema estándar y se comporta igual que otros sistemas de archivos.

Para obtener más información sobre sistemas de archivos, consulte el Capítulo 5, "Administración de sistemas de archivos ZFS de Oracle Solaris".

reflejo

Dispositivo virtual que almacena copias idénticas de datos en dos discos o más. Si falla cualquier disco de un reflejo, cualquier otro disco de ese reflejo puede proporcionar los mismos datos.

agrupación

Conjunto lógico de dispositivos que describe la disposición y las características físicas del almacenamiento disponible. El espacio en el disco para conjuntos de datos que se asigna a partir de una agrupación.

Para obtener más información sobre agrupaciones de almacenamiento, consulte el Capítulo 3, "Administración de agrupaciones de almacenamiento de Oracle Solaris ZFS".

RAID-Z	Dispositivo virtual que almacena datos y la paridad en varios discos. Para obtener más información sobre RAID-Z, consulte "Configuración de grupos de almacenamiento RAID-Z" en la página 53.
actualización de duplicación	El proceso de transferir datos de un dispositivo a otro se denomina *actualización de duplicación*. Por ejemplo, si un dispositivo de reflejo se sustituye o se desconecta, los datos actualizados del dispositivo de reflejo se copian en el dispositivo de reflejo recién restaurado. Este proceso se denomina *resincronización de reflejo* en productos tradicionales de administración de volúmenes. Si desea más información sobre la actualización de de duplicación ZFS (resilver), consulte "Visualización del estado de la actualización de duplicación de datos" en la página 321.
instantánea	Imagen de sólo lectura de un sistema de archivos o volumen de un momento determinado. Para obtener más información sobre instantáneas, consulte "Información general de instantáneas de ZFS" en la página 219.
dispositivo virtual	Dispositivo lógico de un grupo que puede ser un dispositivo físico, un archivo o un conjunto de dispositivos. Si desea más información sobre dispositivos virtuales, consulte "Visualización de información de dispositivos virtuales de agrupaciones de almacenamiento" en la página 62.
volumen	Un conjunto de datos que representa un dispositivo de bloques. Por ejemplo, puede crear un volumen de ZFS como dispositivo de intercambio. Para obtener más información sobre volúmenes de ZFS, consulte "Volúmenes de ZFS" en la página 285.

Requisitos de asignación de nombres de componentes de ZFS

Cada componente de ZFS (por ejemplo, conjunto de datos y agrupación) debe recibir un nombre según las reglas siguientes:

- Cada componente sólo puede contener caracteres alfanuméricos, además de los cuatro caracteres especiales siguientes:

  - Guión bajo (_)
  - Guión (-)
  - Dos puntos (:)
  - Punto (.)
- Los nombres de las agrupaciones deben comenzar con una letra y únicamente pueden contener caracteres alfanuméricos, además de subrayado (_), guión (-) y punto (.). Tenga en cuenta las siguientes restricciones de nombres de agrupaciones:
  - No se permite la secuencia de inicio `c[0-9]`.
  - El nombre `log` está reservado.
  - No se permiten los nombres que comiencen por `mirror`, `raidz` , `raidz1`, `raidz2`, `raidz3` o `spare` porque dichos nombres están reservados.
  - Los nombres de las agrupaciones de datos no pueden contener un signo porcentual (`%`).
- Los nombres de los conjuntos de datos deben comenzar por un carácter alfanumérico.
- Los nombres de los conjuntos de datos no pueden contener un signo porcentual (`%`).

Además, no se permiten los componentes vacíos.

Oracle Solaris ZFS y sistemas de archivos tradicionales

Granularidad de sistemas de archivos ZFS

Desde siempre, los sistemas de archivos se han limitado a un dispositivo y, por lo tanto, al tamaño de dicho dispositivo. Crear y volver a crear sistemas de archivos tradicionales debido a las limitaciones de tamaño requiere mucho tiempo y llega a ser complicado. Los productos tradicionales de administración de volúmenes ayudan a llevar a cabo este proceso.

Como los sistemas de archivos ZFS no se limitan a determinados dispositivos, se pueden crear con rapidez y facilidad, de forma parecida a la creación de directorios. Los sistemas de archivos ZFS aumentan automáticamente en el espacio asignado a la agrupación de almacenamiento en la que residen.

En vez de crear un sistema de archivos, por ejemplo `/export/home`, para administrar numerosos subdirectorios de usuarios, puede crear un sistema de archivos por usuario. Puede

configurar y administrar fácilmente un gran número de sistemas de archivos aplicando propiedades que pueden heredar los sistemas de archivos descendientes dentro de la jerarquía.

Consulte "Creación de una jerarquía para el sistema de archivos ZFS" en la página 43 para ver un ejemplo de creación de una jerarquía de sistema de archivos.

Cálculo del espacio de ZFS

ZFS se basa en el concepto de almacenamiento en agrupaciones. A diferencia de los sistemas de archivos habituales, asignados al almacenamiento físico, todos los sistemas de archivos ZFS de una agrupación comparten el espacio de almacenamiento de la agrupación. Por lo tanto, el espacio disponible en el disco notificado por utilidades como `df` puede llegar a cambiar aunque el sistema de archivos no esté activo, debido a que otros sistemas de archivos de la agrupación consumen o liberan espacio.

El tamaño máximo de los sistemas de archivos se puede restringir mediante cuotas. Para obtener información sobre las cuotas, consulte "Establecimiento de cuotas en sistemas de archivos ZFS" en la página 200. Se puede garantizar una cantidad determinada de espacio en el disco para un sistema de archivos mediante reserva. Para obtener información acerca de las reservas, consulte "Establecimiento de reservas en sistemas de archivos ZFS" en la página 204. Este modelo es muy similar al de NFS, en el que varios directorios se montan desde el mismo sistema de archivos (`/home`).

Todos los metadatos de ZFS se asignan de forma dinámica. Casi todos los demás sistemas de archivos preasignan gran parte de sus metadatos. Al crearse el sistema de archivos, el resultado es un coste inmediato de asignación de espacio para estos metadatos. También significa que está predefinida la cantidad de archivos que admiten los sistemas de archivos. Como ZFS asigna sus metadatos conforme los necesita, no precisa asignación inicial de espacio y la cantidad de archivos que puede admitir está sólo en función del espacio disponible en el disco. La salida del comando `df -g` no significa lo mismo en ZFS que en otros sistemas. El valor de `total files` (total de archivos) que aparece es sólo un cálculo basado en la cantidad de almacenamiento disponible en la agrupación.

ZFS es un sistema de archivos transaccional. Casi todas las modificaciones de sistemas de archivos se incluyen en grupos de transacciones y se envían al disco de manera asíncrona. Hasta que no se envían al disco, se denominan *cambios pendientes*. La cantidad de espacio en el disco utilizado, disponible y que hace referencia a un archivo o sistema de archivos no tiene en cuenta los cambios pendientes. Los cambios pendientes suelen calcularse en pocos segundos. El hecho de enviar un cambio al disco mediante `fsync(3c)` o `O_SYNC` no garantiza necesariamente la actualización inmediata del espacio que se utiliza en el disco.

En un sistema de archivos UFS, el comando `du` informa el tamaño de los bloques de datos en el archivo. En un sistema de archivos ZFS, `du` informa el tamaño real del archivo mientras se encuentra almacenado en el disco. Este tamaño incluye metadatos y compresión. Estos

informes realmente ayudan a responder la pregunta "¿cuánto espacio obtengo si elimino este archivo?". Por lo tanto, incluso cuando la compresión esté desactivada, seguirá viendo resultados diferentes entre ZFS y UFS.

Cuando el consumo de espacio informado por el comando `df` se compara con el comando `zfs list`, tenga en cuenta que `df` está informando el tamaño de la agrupación y no sólo los tamaños del sistema de archivos. Además, `df` no registra los sistemas de archivos descendientes o si existen instantáneas. Si en los sistemas de archivos se establece cualquiera de las propiedades de ZFS, como la compresión y las cuotas, puede resultar difícil reconciliar el consumo de espacio informado por `df`.

Tenga en cuenta las siguientes situaciones que también podrían impactar el consumo de espacio informado:

- Para los archivos que son más mayores que `recordsize`, en general, el último bloque del archivo tendría 1/2 del espacio completo. Con el valor predeterminado `recordsize` establecido en 128 KB, se desaprovechan, aproximadamente, 64 KB por archivo, lo que podría producir un gran impacto. La integración de RFE 6812608 resolvería esta situación. Si activa la compresión, puede solucionar este problema. Incluso, si los datos ya están comprimidos, la porción no utilizada del último bloque estaría vacía y se podría comprimir de forma correcta.
- En una agrupación RAIDZ-2, cada bloque consume, por lo menos, 2 sectores (fragmentos de 512 bytes) de información de paridad. El espacio consumido por la información de paridad no se ha informado, pero dado que puede variar y que puede convertirse en un porcentaje mucho mayor para bloques pequeños, se puede ver un impacto en el informe del espacio. El impacto es más extremo para un `recordsize` establecido en 512 bytes, donde cada bloque lógico de 512 bytes consume 1,5 KB (3 veces más espacio). Independientemente de los datos que se almacenan, si su principal preocupación es la eficacia del espacio, debe dejar el valor predeterminado de `recordsize` (128 KB) y activar la compresión (en el valor predeterminado de `lzjb`).
- El comando `df` no registra los datos duplicados del archivo que fueron eliminados.

Comportamiento de falta de espacio

En ZFS, las instantáneas se crean sin dificultad ni coste alguno. Las instantáneas son comunes en casi todos los entornos de ZFS. Para obtener información sobre instantáneas de ZFS, consulte el Capítulo 6, “Uso de clones e instantáneas de Oracle Solaris ZFS”.

La presencia de instantáneas puede producir comportamientos imprevistos al intentar liberar espacio en el disco. En general, con los permisos pertinentes, es posible eliminar archivos de un sistema de archivos lleno y disponer así de más espacio en el disco en el sistema de archivos. No obstante, si el archivo que se va a eliminar existe en una instantánea del sistema de archivos, suprimirlo no proporcionará más espacio libre. Se sigue haciendo referencia a los bloques utilizados por el archivo desde la instantánea.

Como consecuencia, suprimir un archivo puede suponer más consumo del espacio en el disco, ya que para reflejar el nuevo estado del espacio de nombre se debe crear una versión nueva del directorio. Este comportamiento significa que al intentar eliminar un archivo se puede generar un error `ENOSPC` o `EDQUOT` imprevisto.

Montaje de sistemas de archivos ZFS

ZFS reduce la complejidad y facilita la administración. Por ejemplo, en los sistemas de archivos tradicionales debe editar el archivo `/etc/vfstab` cada vez que agregue un sistema de archivos nuevo. ZFS ha suprimido este requisito al montar y desmontar automáticamente los sistemas de archivos en función de las propiedades del sistema de archivos. No necesita gestionar entradas ZFS en el archivo `/etc/vfstab`.

Para obtener más información sobre cómo montar y compartir sistemas de archivos ZFS, consulte "Montaje de sistemas de archivos ZFS" en la página 183.

Administración tradicional de volúmenes

Como se explica en "Almacenamiento en grupos de ZFS" en la página 28, con ZFS no se necesita un administrador de volúmenes aparte. ZFS funciona en dispositivos básicos, lo que permite crear una agrupación de almacenamiento a base de volúmenes lógicos, ya sea de software o hardware. No se recomienda esta configuración, puesto que el funcionamiento óptimo de ZFS se da con dispositivos físicos básicos. El uso de volúmenes lógicos puede perjudicar el rendimiento, la fiabilidad o ambas cosas, y se debe evitar.

Modelo de ACL de Solaris basado en NFSv4

Las versiones anteriores del sistema operativo Solaris admitían una implementación de ACL que se basaba sobre todo en la especificación de ACL de borrador POSIX. Las ACL basadas en el borrador POSIX se utilizan para proteger los archivos UFS. Se emplea un nuevo modelo Solaris ACL basado en la especificación NFSv4 para proteger archivos ZFS.

A continuación se exponen las diferencias principales del nuevo modelo Solaris ACL:

- El modelo se basa en la especificación de NFSv4 y se parece a las ACL del tipo NT.
- Este modelo ofrece un conjunto mucho más granular de privilegios de acceso.
- Las listas ACL se definen y visualizan con los comandos `chmod` e `ls`, en lugar de los comandos `setfacl` y `getfacl`.
- Semántica heredada mucho más rica para establecer la forma en que se aplican privilegios de acceso del directorio a los subdirectorios, y así sucesivamente.

Para obtener más información sobre el uso de las ACL con archivos ZFS, consulte Capítulo 7, "Uso de listas de control de acceso y atributos para proteger archivos Oracle Solaris ZFS".

◆ ◆ ◆ CAPÍTULO 2

Procedimientos iniciales con Oracle Solaris ZFS

Este capítulo proporciona instrucciones paso a paso para definir una configuración básica de Oracle Solaris ZFS. Al terminar este capítulo, habrá adquirido nociones básicas sobre el funcionamiento de los comandos de ZFS, y debería ser capaz de crear sistemas de archivos y una agrupación sencilla. Este capítulo no profundiza en el contenido. Para obtener información más detallada, consulte los capítulos siguientes.

Este capítulo se divide en las secciones siguientes:

Perfiles de derechos de ZFS

Si desea efectuar tareas de administración de ZFS sin utilizar la cuenta de superusuario (root), puede asumir una función con cualquiera de los perfiles siguientes para llevar a cabo dichas tareas de administración:

- Administración de almacenamiento de ZFS: proporciona privilegios para crear, destruir y manipular dispositivos en una agrupación de almacenamiento de ZFS
- Administración de sistemas de archivos ZFS: proporciona privilegios para crear, destruir y modificar sistemas de archivos ZFS

Para obtener más información sobre la creación o la asignación de roles, consulte *Administración de Oracle Solaris 11.1: servicios de seguridad*.

Además de utilizar funciones RBAC para administrar sistemas de archivos ZFS, también puede considerar la posibilidad de utilizar la administración delegada de ZFS para tareas de administración ZFS distribuidas. Para más información, consulte el Capítulo 8, "Administración delegada de ZFS Oracle Solaris".

Recomendaciones y requisitos de software y hardware para ZFS

Antes de utilizar el software de ZFS, revise los requisitos y las recomendaciones de software y hardware siguientes:

- Utilice un sistema basado en SPARC o x86 que ejecute una versión compatible con Oracle de Solaris.
- Una agrupación de almacenamiento necesita como mínimo 64 MB de espacio en el disco. El tamaño de disco mínimo es 128 MB.
- Para obtener un buen rendimiento de ZFS, determine los requisitos de memoria en virtud de la carga de trabajo.
- Si crea una configuración de agrupación reflejada, utilice varios controladores.

Creación de un sistema de archivos ZFS básico

Se ha intentado diseñar la administración de ZFS con la máxima sencillez posible. Entre los objetivos del diseño está la reducción del número de comandos necesarios para crear un sistema de archivos utilizable. Por ejemplo, al crear una agrupación, se crea un sistema de archivos ZFS y se monta automáticamente.

El ejemplo siguiente ilustra la manera de crear una agrupación de almacenamiento reflejado denominado `tank` y un sistema de archivos ZFS denominado `tank` en un comando. Suponga que se pueden utilizar todos los discos `/dev/dsk/c1t0d0` y `/dev/dsk/c2t0d0`.

```
# zpool create tank mirror c1t0d0 c2t0d0
```

Para obtener más información sobre configuraciones de grupos ZFS redundantes, consulte "Funciones de repetición de una agrupación de almacenamiento de ZFS" en la página 52.

El nuevo sistema de archivos ZFS, `tank`, puede usar tanto espacio como necesite y se monta automáticamente en `/tank`.

```
# mkfile 100m /tank/foo
# df -h /tank
Filesystem             size   used  avail capacity  Mounted on
tank                   80G   100M    80G     1%    /tank
```

Quizá desee crear sistemas de archivos adicionales en una agrupación. Los sistemas de archivos ofrecen puntos que permiten administrar distintos conjuntos de datos en la misma agrupación.

El ejemplo siguiente ilustra la manera de crear un sistema de archivos denominado `fs` en la agrupación de almacenamiento `tank`.

```
# zfs create tank/fs
```

El nuevo sistema de archivos ZFS, `tank/fs`, puede utilizar la cantidad de espacio en el disco que necesite y se monta automáticamente en `/tank/fs`.

```
# mkfile 100m /tank/fs/foo
# df -h /tank/fs
Filesystem             size   used  avail capacity  Mounted on
tank/fs                 80G   100M    80G     1%    /tank/fs
```

Normalmente, el objetivo es crear y organizar una jerarquía de sistemas de archivos que se ajuste a los requisitos de su organización. Para obtener más información sobre cómo crear jerarquías de sistemas de archivos ZFS, consulte "Creación de una jerarquía para el sistema de archivos ZFS" en la página 43.

Creación de un grupo de almacenamiento de ZFS básico

El ejemplo anterior es una muestra de la sencillez de ZFS. El resto de este capítulo expone un ejemplo más completo y similar a la situación de su entorno. Las primeras tareas son establecer los requisitos de almacenamiento y crear un grupo de almacenamiento. La agrupación describe las características físicas del almacenamiento y se deben crear antes que un sistema de archivos.

▼ Identificación de los requisitos del grupo de almacenamiento de ZFS

1 Averigüe qué dispositivos están disponibles para la agrupación de almacenamiento.

Antes de crear una agrupación de almacenamiento, debe establecer los dispositivos que almacenarán los datos. Deben ser discos de al menos 128 MB y no los deben utilizar otros componentes del sistema operativo. Los dispositivos pueden ser segmentos de disco al que se ha dado formato previamente, o discos completos a los que ZFS da formato como un único segmento grande.

En el ejemplo de almacenamiento de "Cómo crear una agrupación de almacenamiento de ZFS" en la página 42, suponga que se pueden utilizar los discos `/dev/dsk/c2t0d0` y `/dev/dsk/c0t1d0` completos.

Para obtener más información sobre los discos y cómo se utilizan y etiquetan, consulte "Utilización de discos en un grupo de almacenamiento de ZFS" en la página 48.

2 Seleccione la replicación de datos.

ZFS admite diversos tipos de repetición de datos; esto determina los tipos de errores de hardware que puede soportar la agrupación. ZFS admite configuraciones no redundantes (repartidas en bandas), así como reflejo y RAID-Z (una variación de RAID-5).

En el ejemplo de almacenamiento de "Cómo crear una agrupación de almacenamiento de ZFS" en la página 42, se utiliza el reflejo básico de dos discos disponibles.

Si desea más información sobre las características de replicación de ZFS, consulte "Funciones de repetición de una agrupación de almacenamiento de ZFS" en la página 52.

▼ Cómo crear una agrupación de almacenamiento de ZFS

1 Adquiera el perfil de usuario root o asuma una función equivalente con el perfil adecuado de derechos de ZFS.

Para obtener más información sobre los perfiles de derechos de ZFS, consulte "Perfiles de derechos de ZFS" en la página 39.

2 Elija un nombre para la agrupación de almacenamiento.

El nombre de agrupación sirve para identificar la agrupación de almacenamiento cuando se utilizan los comandos `zpool` y `zfs`. Escoja el nombre de agrupación que prefiera, siempre y cuando cumpla los requisitos de asignación de nombres especificados en "Requisitos de asignación de nombres de componentes de ZFS" en la página 32.

3 Cree la agrupación.

Por ejemplo, el siguiente comando crea una agrupación reflejada denominada `tank`:

```
# zpool create tank mirror c1t0d0 c2t0d0
```

Si uno o más dispositivos contienen otro sistema de archivos o se están utilizando, el comando no puede crear la agrupación.

Para obtener más información sobre cómo crear agrupaciones de almacenamiento, consulte "Creación de grupos de almacenamiento de ZFS" en la página 55. Para obtener más información sobre cómo establecer el uso de dispositivos, consulte "Detección de dispositivos en uso" en la página 63.

4 Examine los resultados.

Puede determinar si la agrupación se ha creado correctamente mediante el comando `zpool list`.

```
# zpool list
NAME                    SIZE    ALLOC   FREE    CAP  HEALTH     ALTROOT
tank                     80G    137K     80G     0%  ONLINE     -
```

Para obtener más información sobre cómo ver el estado de las agrupaciones, consulte "Consulta del estado de una agrupación de almacenamiento de ZFS" en la página 93.

Creación de una jerarquía para el sistema de archivos ZFS

Después de crear una agrupación de almacenamiento para almacenar los datos, puede crear la jerarquía del sistema de archivos. Las jerarquías son mecanismos sencillos pero potentes para organizar la información. También resultan muy familiares a quienes hayan utilizado un sistema de archivos.

ZFS permite que los sistemas de archivos se organicen en jerarquías, donde cada sistema de archivos tiene un solo superior. El root de la jerarquía siempre es el nombre del grupo. ZFS integra esta jerarquía mediante la admisión de herencia de propiedades, de manera que las propiedades habituales se puedan configurar rápida y fácilmente en todos los árboles de los sistemas de archivos.

▼ Cómo establecer la jerarquía del sistema de archivos ZFS

1 Elija la granularidad del sistema de archivos.

Los sistemas de archivos ZFS son el punto central de administración. Son ligeros y se pueden crear fácilmente. Un modelo perfectamente válido es un sistema de archivos por usuario o proyecto, ya que posibilita propiedades, instantáneas y copias de seguridad que se controlan por usuario o por proyecto.

Se crean dos sistemas de archivos ZFS, `jeff` y `bill`, en "Creación de sistemas de archivos ZFS" en la página 44.

Para obtener más información sobre la administración de sistemas de archivos, consulte el Capítulo 5, "Administración de sistemas de archivos ZFS de Oracle Solaris".

2 Agrupe sistemas de archivos similares.

ZFS permite que los sistemas de archivos se organicen en jerarquías, de modo que se puedan agrupar los sistemas de archivos similares. Este modelo ofrece un punto central de administración para controlar propiedades y administrar sistemas de archivos. Los sistemas de archivos similares se deben crear con un nombre común.

En el ejemplo de "Creación de sistemas de archivos ZFS" en la página 44, los dos sistemas de archivos se ubican en un sistema de archivos denominado `home`.

3 Seleccione las propiedades del sistema de archivos.

La mayoría de las características del sistema de archivos se controlan mediante propiedades. Dichas propiedades controlan diversos comportamientos, por ejemplo la ubicación donde se montan los sistemas de archivos, su manera de compartirse, si utilizan compresión y si se ejecuta alguna cuota.

En el ejemplo de "Creación de sistemas de archivos ZFS" en la página 44, todos los directorios de inicio se montan en `/export/zfs/` *usuario*, se comparten mediante NFS y se activa la compresión. Además, se aplica una cuota de 10 GB en el usuario `jeff`.

Para obtener más información sobre propiedades, consulte "Introducción a las propiedades de ZFS" en la página 151.

▼ Creación de sistemas de archivos ZFS

1 Adquiera el perfil de usuario root o asuma una función equivalente con el perfil adecuado de derechos de ZFS.

Para obtener más información sobre los perfiles de derechos de ZFS, consulte "Perfiles de derechos de ZFS" en la página 39.

2 Cree la jerarquía que necesite.

En este ejemplo, se crea un sistema de archivos que actúa como contenedor de determinados sistemas de archivos.

```
# zfs create tank/home
```

3 Configure las propiedades heredadas.

Después de establecer la jerarquía del sistema de archivos, configure las propiedades que deben compartir todos los usuarios:

```
# zfs set mountpoint=/export/zfs tank/home
# zfs set share.nfs=on tank/home
# zfs set compression=on tank/home
# zfs get compression tank/home
NAME             PROPERTY       VALUE                      SOURCE
tank/home        compression    on                         local
```

Las propiedades del sistema de archivos pueden establecerse al crear dicho sistema de archivos. Por ejemplo:

```
# zfs create -o mountpoint=/export/zfs -o share.nfs=on -o compression=on tank/home
```

Para obtener más información sobre la propiedades y la herencia de propiedades, consulte "Introducción a las propiedades de ZFS" en la página 151.

A continuación, los sistemas de archivos se agrupan en el sistema de archivos `home` en la agrupación `tank`.

4 Cree los sistemas de archivos.

Puede que los sistemas de archivos se hayan creado y que las propiedades se hayan cambiado en el nivel de `home`. Todas las propiedades se pueden cambiar dinámicamente mientras se utilizan los sistemas de archivos.

```
# zfs create tank/home/jeff
# zfs create tank/home/bill
```

Estos sistemas de archivos heredan los valores de propiedades de sus superiores, de modo que se montan automáticamente en `/export/zfs/` *usuario* y se comparten con NFS. No hace falta editar el archivo `/etc/vfstab` ni `/etc/dfs/dfstab`.

Para obtener más información sobre cómo crear sistemas de archivos, consulte "Creación de un sistema de archivos ZFS" en la página 148.

Para obtener más información sobre cómo montar y compartir sistemas de archivos , consulte "Montaje de sistemas de archivos ZFS" en la página 183.

5 Configure las propiedades específicas del sistema de archivos.

En este ejemplo, se asigna una cuota de 10 GB al usuario `jeff`. Esta propiedad establece un límite en la cantidad de espacio que puede consumir, sea cual sea el espacio disponible en la agrupación.

```
# zfs set quota=10G tank/home/jeff
```

6 Examine los resultados.

Consulte la información disponible sobre el sistema de archivos mediante el comando `zfs list`:

```
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
tank                  92.0K  67.0G   9.5K  /tank
tank/home             24.0K  67.0G     8K  /export/zfs
tank/home/bill           8K  67.0G     8K  /export/zfs/bill
tank/home/jeff           8K  10.0G     8K  /export/zfs/jeff
```

Tenga en cuenta que el usuario `jeff` sólo tiene disponible un espacio de 10 GB, mientras que el usuario `bill` puede utilizar toda la agrupación (67 GB).

Para obtener más información sobre cómo ver el estado del sistema de archivos, consulte "Consulta de información del sistema de archivos ZFS" en la página 176.

Para obtener más información sobre cómo se utiliza y calcula el espacio en el disco, consulte "Cálculo del espacio de ZFS" en la página 34.

◆ ◆ ◆ CAPÍTULO 3

Administración de agrupaciones de almacenamiento de Oracle Solaris ZFS

Este capítulo describe cómo crear y administrar agrupaciones de almacenamiento en Oracle Solaris ZFS.

Este capítulo se divide en las secciones siguientes:

Componentes de una agrupación de almacenamiento de ZFS

Las secciones siguientes ofrecen información detallada sobre estos componentes de agrupación de almacenamiento:

Utilización de discos en un grupo de almacenamiento de ZFS

El elemento más básico de una agrupación de almacenamiento es el almacenamiento físico. El almacenamiento físico puede ser cualquier dispositivo de bloque de al menos 128 MB. Este dispositivo suele ser una unidad de disco duro visible en el sistema en el directorio `/dev/dsk`.

Un dispositivo de almacenamiento puede ser todo un disco (`c1t0d0`) o un determinado segmento (`c0t0d0s7`). Se recomienda utilizar un disco entero, para lo cual no hace falta dar ningún formato especial al disco. ZFS da formato al disco mediante la etiqueta EFI para que contenga un solo segmento grande. Si se utiliza de este modo, la tabla de partición que aparece junto al comando `format` tiene un aspecto similar al siguiente:

```
Current partition table (original):
Total disk sectors available: 143358287 + 16384 (reserved sectors)

Part      Tag    Flag     First Sector         Size         Last Sector
  0        usr    wm               256       68.36GB          143358320
  1 unassigned    wm                 0           0               0
  2 unassigned    wm                 0           0               0
  3 unassigned    wm                 0           0               0
  4 unassigned    wm                 0           0               0
  5 unassigned    wm                 0           0               0
  6 unassigned    wm                 0           0               0
  8   reserved    wm         143358321        8.00MB          143374704
```

Cuando Oracle Solaris 11.1 está instalado, se aplica una etiqueta EFI (GPT) a discos de agrupación raíz en un sistema basado en x86, en la mayoría de los casos, que es similar a la siguiente:

```
Current partition table (original):
Total disk sectors available: 27246525 + 16384 (reserved sectors)

Part      Tag    Flag     First Sector         Size         Last Sector
  0  BIOS_boot    wm               256     256.00MB          524543
  1        usr    wm            524544      12.74GB          27246558
  2 unassigned    wm                 0           0               0
  3 unassigned    wm                 0           0               0
  4 unassigned    wm                 0           0               0
  5 unassigned    wm                 0           0               0
  6 unassigned    wm                 0           0               0
  8   reserved    wm          27246559       8.00MB          27262942
```

En la salida anterior, la partición 0 (`BIOS boot`) contiene información de inicio GPT necesaria. De manera similar a partición 8, no requiere administración y no se debe modificar. El sistema de archivos raíz se incluye en la partición 1.

En un sistema SPARC con firmware actualizado que se instaló con Oracle Solaris 11.1, se aplica una etiqueta de disco EFI (GPT). Por ejemplo:

```
Current partition table (original):
Total disk sectors available: 143358320 + 16384 (reserved sectors)

Part      Tag    Flag     First Sector         Size         Last Sector
  0        usr    wm               256       68.36GB          143358320
  1 unassigned    wm                 0           0               0
  2 unassigned    wm                 0           0               0
  3 unassigned    wm                 0           0               0
  4 unassigned    wm                 0           0               0
  5 unassigned    wm                 0           0               0
  6 unassigned    wm                 0           0               0
  8   reserved    wm         143358321        8.00MB          143374704
```

Revise las siguientes consideraciones al utilizar discos enteros en las agrupaciones de almacenamiento de ZFS:

- Para utilizar un disco entero, se le debe asignar un nombre de acuerdo con la convención de nombres `/dev/dsk/cNtNdN`. Algunos controladores de terceros utilizan otra convención de asignación de nombres o sitúan discos en una ubicación diferente de la del directorio `/dev/dsk`. Para utilizar estos discos, debe etiquetarlos manualmente y proporcionar un segmento a ZFS.
- En un sistema basado en x86, el disco debe tener una partición `fdisk` válida de Solaris. Para obtener más información sobre la creación o el cambio de una partición `fdisk` de Solaris, consulte "configuración de discos para sistemas de archivos ZFS (mapa de tareas)" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.
- ZFS aplica una etiqueta EFI cuando crea una agrupación de almacenamiento con discos completos. Para obtener más información sobre etiquetas EFI, consulte "Etiqueta de disco EFI (GPT)" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.
- El programa de instalación de Oracle Solaris 11.1 aplica una etiqueta EFI (GPT) para los discos de agrupación raíz en un sistema basado en SPARC con firmware compatible con GPT y en un sistema basado en x86, en la mayoría de los casos. Para obtener más información, consulte "Requisitos de la agrupación raíz ZFS" en la página 120.

Los discos se pueden especificar mediante una ruta completa, como `/dev/dsk/c1t0d0`, o un nombre abreviado que se componga del nombre de dispositivo en el directorio `/dev/dsk`, por ejemplo `c1t0d0`. A continuación puede ver algunos nombres de disco válidos:

- `c1t0d0`
- `/dev/dsk/c1t0d0`
- `/dev/foo/disk`

Uso de segmentos en una agrupación de almacenamiento de ZFS

Los discos se pueden etiquetar con una etiqueta VTOC (SMI) antigua de Solaris al crear una agrupación de almacenamiento con un segmento de disco, pero no se recomienda el uso de segmentos de disco para una agrupación, ya que la gestión de segmentos de discos es más difícil.

En un sistema basado en SPARC, un disco de 72 GB tiene 68 GB de espacio utilizable ubicados en el segmento 0, tal y como se muestra en la siguiente salida de `format`.

```
# format
.
.
.
Specify disk (enter its number): 4
selecting c1t1d0
partition> p
Current partition table (original):
Total disk cylinders available: 14087 + 2 (reserved cylinders)

Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       0 - 14086       68.35GB    (14087/0/0) 143349312
  1 unassigned    wm       0                0         (0/0/0)             0
  2     backup    wm       0 - 14086       68.35GB    (14087/0/0) 143349312
  3 unassigned    wm       0                0         (0/0/0)             0
  4 unassigned    wm       0                0         (0/0/0)             0
  5 unassigned    wm       0                0         (0/0/0)             0
  6 unassigned    wm       0                0         (0/0/0)             0
  7 unassigned    wm       0                0         (0/0/0)             0
```

En un sistema basado en x86, un disco de 72 GB tiene 68 GB de espacio utilizable ubicados en el segmento 0, tal y como se muestra en la siguiente salida de `format`. En el segmento 8 se incluye una pequeña cantidad de información de inicio. El segmento 8 no requiere administración y no se puede cambiar.

```
# format
.
.
.
selecting c1t0d0
partition> p
Current partition table (original):
Total disk cylinders available: 49779 + 2 (reserved cylinders)

Part      Tag    Flag     Cylinders         Size            Blocks
  0       root    wm       1 - 49778       68.36GB    (49778/0/0) 143360640
  1 unassigned    wu       0                0         (0/0/0)             0
  2     backup    wm       0 - 49778       68.36GB    (49779/0/0) 143363520
  3 unassigned    wu       0                0         (0/0/0)             0
  4 unassigned    wu       0                0         (0/0/0)             0
  5 unassigned    wu       0                0         (0/0/0)             0
  6 unassigned    wu       0                0         (0/0/0)             0
  7 unassigned    wu       0                0         (0/0/0)             0
  8       boot    wu       0 -     0        1.41MB    (1/0/0)          2880
  9 unassigned    wu       0                0         (0/0/0)             0
```

En un sistema basado en x86, también existe una partición `fdisk`. Una partición `fdisk` es representada por un nombre de dispositivo `/dev/dsk/cN[tN]dNpN` y actúa como un contenedor de los segmentos disponibles del disco. No utilice un dispositivo `cN[tN]dNpN` para un componente de agrupación de almacenamiento ZFS porque esta configuración no está probada ni admitida.

Utilización de archivos en un grupo de almacenamiento de ZFS

El ZFS también permite utilizar los archivos como dispositivos virtuales en la agrupación de almacenamiento. Esta función se aplica sobre todo a verificaciones y pruebas sencillas, no es apta la producción.

- Si crea un grupo de ZFS a partir de archivos en un sistema de archivos UFS, de forma implícita se basa en UFS para garantizar la corrección y la semántica sincrónica.
- Si crea un grupo de ZFS a partir de archivos o volúmenes creados en otra agrupación de ZFS, entonces puede que el sistema se bloquee o produzca un aviso grave del sistema.

Sin embargo, los archivos pueden ser bastante útiles al probar ZFS por primera vez o experimentar con configuraciones más complejas cuando no hay suficientes dispositivos físicos. Se deben especificar todos los archivos como rutas completas y deben tener al menos 64 MB de tamaño.

Consideraciones para grupos de almacenamiento de ZFS

Tenga en cuenta lo siguiente al crear y gestionar grupos de almacenamiento de ZFS.

- La forma más sencilla de crear agrupaciones de almacenamiento de ZFS es usar todo el disco físico. Las configuraciones de ZFS se vuelven más complejas de forma progresiva respecto a administración, fiabilidad y rendimiento, cuando se crean grupos de segmentos de discos, LUN (unidades lógicas) en matrices RAID de hardware o volúmenes presentados por administradores de volúmenes basados en software. Las consideraciones siguientes pueden ayudar a determinar la configuración de ZFS con otras soluciones de almacenamiento de hardware o software:
  - Si crea una configuración de ZFS sobre unidades LUN a partir de matrices RAID de hardware, debe comprender la relación entre las características de redundancia de ZFS y las de redundancia ofrecidas por la matriz. Determinadas configuraciones pueden dar una redundancia y un rendimiento adecuados, pero otras quizá no lo hagan.
  - Puede crear dispositivos lógicos para ZFS mediante volúmenes presentados por administradores de volúmenes basados en software. Sin embargo, estas configuraciones no se recomiendan. Aunque ZFS funcione correctamente en estos dispositivos, podría presentar un rendimiento no del todo satisfactorio.

    Para obtener información adicional sobre las recomendaciones de agrupaciones de almacenamiento, consulte Capítulo 12, “Prácticas de ZFS recomendadas por Oracle Solaris”.

- Los discos se identifican por la ruta e ID de dispositivo, si lo hay. En sistemas donde hay información de ID de dispositivo disponible, este método de identificación permite volver a configurar los dispositivos sin tener que actualizar ZFS. Debido a que los procedimientos de generación y administración de ID de dispositivos pueden variar de un sistema a otro, se recomienda exportar la agrupación antes de mover dispositivos (por ejemplo, trasladar un disco de un controlador a otro). Un evento del sistema como, por ejemplo, una actualización de firmware u otro cambio de hardware, podría cambiar el ID de dispositivo en la agrupación de almacenamiento de ZFS y hacer que los dispositivos no estén disponibles.

Funciones de repetición de una agrupación de almacenamiento de ZFS

ZFS proporciona redundancia de datos y propiedades de autocorrección en configuraciones reflejadas y RAID-Z.

- "Configuración reflejada de agrupaciones de almacenamiento" en la página 52
- "Configuración de grupos de almacenamiento RAID-Z" en la página 53
- "Datos de recuperación automática en una configuración redundante" en la página 54
- "Reparto dinámico de discos en bandas en un grupo de almacenamiento" en la página 54
- "Agrupación de almacenamiento híbrido de ZFS " en la página 54

Configuración reflejada de agrupaciones de almacenamiento

Una configuración reflejada de agrupación de almacenamiento necesita al menos dos discos, preferiblemente en controladores independientes. En una configuración reflejada se pueden utilizar muchos discos. Asimismo, puede crear más de un reflejo en cada grupo. Conceptualmente hablando, una configuración reflejada sencilla tendría un aspecto similar al siguiente:

```
mirror c1t0d0 c2t0d0
```

Desde un punto de vista conceptual, una configuración reflejada más compleja tendría un aspecto similar al siguiente:

```
mirror c1t0d0 c2t0d0 c3t0d0 mirror c4t0d0 c5t0d0 c6t0d0
```

Para obtener información sobre cómo crear agrupaciones de almacenamiento reflejadas, consulte "Creación de una agrupación de almacenamiento reflejado" en la página 56.

Configuración de grupos de almacenamiento RAID-Z

Además de una configuración reflejada de agrupación de almacenamiento, ZFS ofrece una configuración de RAID-Z con tolerancia a fallos de paridad sencilla, doble o triple. RAID-Z de paridad sencilla (`raidz` o `raidz1`) es similar a RAID-5. RAID-Z de paridad doble (`raidz2`) es similar a RAID-6.

Para obtener más información sobre RAIDZ-3 (`raidz3`), consulte el blog siguiente:

`http://blogs.oracle.com/ahl/entry/triple_parity_raid_z`

Todos los algoritmos tradicionales similares a RAID-5 (por ejemplo, RAID-4, RAID-6, RDP y EVEN-ODD) tienen un problema conocido como *error de escritura por caída del sistema de RAID-5*. Si sólo se escribe parte de una distribución de discos en bandas de RAID-5 y la alimentación se interrumpe antes de que todos los bloques se hayan escrito en el disco, la paridad permanece sin sincronizarse con los datos, y por eso deja de ser útil (a menos que se sobrescriba con una escritura posterior de todas las bandas). En RAID-Z, ZFS utiliza repartos de discos en bandas de RAID de ancho variable, de manera que todas las escrituras son de reparto total de discos en bandas. Este diseño sólo es posible porque ZFS integra el sistema de archivos y la administración de dispositivos de manera que los metadatos del sistema de archivos tengan suficiente información sobre el modelo de redundancia de los datos subyacentes para controlar los repartos de discos en bandas de RAID de anchura variable. RAID-Z es la primera solución exclusiva de software en el mundo para el error de escritura por caída del sistema de RAID-5.

Una configuración de RAID-Z con N discos de tamaño X con discos de paridad P puede contener aproximadamente (N-P)*X bytes, así como admitir uno o más dispositivos P con errores antes de que se comprometa la integridad de los datos. Para la configuración de RAID-Z de paridad sencilla se necesita un mínimo de dos discos; se necesitan al menos tres para la configuración de RAID-Z de paridad doble, y así sucesivamente. Por ejemplo, si tiene tres discos en una configuración de RAID-Z de paridad sencilla, los datos de la paridad ocupan un espacio equivalente a uno de los tres discos. Por otro lado, para crear una configuración de RAID-Z no se necesita hardware especial.

Conceptualmente hablando, una configuración de RAID-Z con tres discos tendría un aspecto similar al siguiente:

```
raidz c1t0d0 c2t0d0 c3t0d0
```

Mientras que una configuración reflejada más compleja tendría un aspecto similar al siguiente:

```
raidz c1t0d0 c2t0d0 c3t0d0 c4t0d0 c5t0d0 c6t0d0 c7t0d0
raidz c8t0d0 c9t0d0 c10t0d0 c11t0d0c12t0d0 c13t0d0 c14t0d0
```

Si desea crear una configuración de RAID-Z con muchos discos, puede ser conveniente dividir los discos en varios grupos. Por ejemplo, una configuración de RAID-Z con 14 discos se puede dividir en dos grupos de 7 discos. En principio, las configuraciones de RAID-Z con agrupaciones de un solo dígito de discos funcionan mejor.

Para obtener información sobre cómo crear una agrupación de almacenamiento de RAID-Z, consulte "Creación de una agrupación de almacenamiento de RAID-Z" en la página 57.

Para obtener más información sobre cómo elegir entre una configuración reflejada o una de RAID-Z en función del espacio y el rendimiento, consulte el blog siguiente:

`http://blogs.oracle.com/roch/entry/when_to_and_not_to`

Para obtener información adicional sobre las recomendaciones de agrupaciones de almacenamiento RAID-Z, consulte Capítulo 12, "Prácticas de ZFS recomendadas por Oracle Solaris".

Agrupación de almacenamiento híbrido de ZFS

La agrupación de almacenamiento híbrido de ZFS, disponible en la serie de productos de Oracle Sun Storage 7000, es una agrupación de almacenamiento especial que combina DRAM, SSD y HDD con el fin de mejorar el rendimiento y aumentar la capacidad, al tiempo que se reduce el consumo de energía. Con la interfaz de administración de este producto, puede seleccionar la configuración de redundancia de ZFS de la agrupación de almacenamiento y administrar fácilmente otras opciones de configuración.

Para obtener más información acerca de este producto, consulte la *Guía de administración del sistema de almacenamiento unificado Sun Storage*.

Datos de recuperación automática en una configuración redundante

ZFS ofrece soluciones para datos de recuperación automática en una configuración de RAID-Z o reflejada.

Si se detecta un bloque de datos incorrectos, ZFS no sólo recupera los datos correctos de otra copia redundante, sino que también repara los datos incorrectos al sustituirlos por la copia correcta.

Reparto dinámico de discos en bandas en un grupo de almacenamiento

ZFS reparte dinámicamente los datos de los discos en bandas entre todos los dispositivos virtuales de nivel superior. La elección de ubicación de los datos se efectúa en el momento de la escritura, por lo que en el momento de la asignación no se crean bandas de ancho fijo.

Cuando se agregan a una agrupación dispositivos virtuales nuevos, ZFS asigna datos gradualmente al nuevo dispositivo con el fin de mantener el rendimiento y las normas de asignación de espacio. Cada dispositivo virtual puede ser también un reflejo o un dispositivo de

RAID-Z que contenga otros archivos o dispositivos de discos. Esta configuración ofrece flexibilidad a la hora de controlar las características predeterminadas de la agrupación. Por ejemplo, puede crear las configuraciones siguientes a partir de cuatro discos:

- Cuatro discos que utilicen reparto dinámico de discos en bandas
- Una configuración de RAID-Z de cuatro vías
- Dos reflejos de dos vías que utilicen reparto dinámico de discos en bandas

Aunque ZFS admite la combinación de diversos tipos de dispositivos virtuales en la misma agrupación, debe evitar hacerlo. Por ejemplo, puede crear un grupo con un reflejo de dos vías y una configuración de RAID-Z de tres vías. Sin embargo, la tolerancia a errores es tan buena como el peor de los dispositivos virtuales de que disponga, en este caso RAID-Z. La práctica recomendada es utilizar dispositivos virtuales de nivel superior del mismo tipo con idéntico nivel de redundancia en cada dispositivo.

Creación y destrucción de agrupaciones de almacenamiento de ZFS

Las secciones siguientes describen distintas situaciones de creación y destrucción de agrupaciones de almacenamiento de ZFS:

La creación y la destrucción de agrupaciones son procesos fáciles y rápidos. Sin embargo, estas operaciones se deben efectuar con cuidado. Aunque las comprobaciones se efectúan para impedir el uso de dispositivos se están usando en una nueva agrupación, ZFS no puede saber siempre si un dispositivo ya se está utilizando. La destrucción de una agrupación es más fácil que crear uno. Utilice `zpool destroy` con precaución. Este comando sencillo tiene importantes consecuencias.

Creación de grupos de almacenamiento de ZFS

Para crear una agrupación de almacenamiento, utilice el comando `zpool create`. Este comando toma un nombre de grupo y cualquier cantidad de dispositivos virtuales como argumentos. El nombre de la agrupación debe atenerse a los requisitos de denominación indicados en "Requisitos de asignación de nombres de componentes de ZFS" en la página 32.

Creación de un grupo de almacenamiento básico

El comando siguiente crea un recurso con el nombre `tank` que se compone de los discos `c1t0d0` y `c1t1d0`:

```
# zpool create tank c1t0d0 c1t1d0
```

Los nombres de dispositivo que representan los discos completos se encuentran en el directorio `/dev/dsk`; ZFS los etiqueta correspondientemente para que contengan un segmento único y de gran tamaño. Los datos se reparten dinámicamente en ambos discos.

Creación de una agrupación de almacenamiento reflejado

Para crear una agrupación reflejada, utilice la palabra clave `mirror`, seguida de varios dispositivos de almacenamiento que incluirán el reflejo. Se pueden especificar varios reflejos si se repite la palabra clave `mirror` en la línea de comandos. El comando siguiente crea una agrupación con dos reflejos de dos vías:

```
# zpool create tank mirror c1d0 c2d0 mirror c3d0 c4d0
```

La segunda palabra clave `mirror` indica que se especifica un nuevo dispositivo virtual de nivel superior. Los datos se colocan dinámicamente en bandas en los dos reflejos, con la correspondiente redundancia de datos en cada disco.

Para obtener más información sobre las configuraciones reflejadas recomendadas, consulte Capítulo 12, "Prácticas de ZFS recomendadas por Oracle Solaris".

En la actualidad, en una configuración reflejada de ZFS son posibles las operaciones siguientes:

- Agregar otro conjunto de discos de nivel superior adicional (vdev) a una configuración reflejada existente. Para obtener más información, consulte "Agregación de dispositivos a un grupo de almacenamiento" en la página 68.
- Conectar discos adicionales a una configuración reflejada. Conectar discos adicionales a una configuración no repetida para crear una configuración reflejada. Para obtener más información, consulte "Conexión y desconexión de dispositivos en una agrupación de almacenamiento" en la página 73.
- Reemplazar uno o varios discos de una configuración reflejada existente si los discos de sustitución son mayores o iguales que el dispositivo que se va a reemplazar. Para obtener más información, consulte "Sustitución de dispositivos en un grupo de almacenamiento" en la página 81.
- Desconectar un disco de una configuración reflejada si los demás dispositivos proporcionan a la configuración la redundancia necesaria. Para obtener más información, consulte "Conexión y desconexión de dispositivos en una agrupación de almacenamiento" en la página 73.

- Dividir una configuración reflejada mediante la desconexión de uno de los discos para crear una agrupación nueva idéntica. Para obtener más información, consulte "Creación de una nueva agrupación mediante la división de una agrupación de almacenamiento de ZFS reflejada" en la página 75.

No se puede eliminar directamente de una agrupación de almacenamiento reflejada un dispositivo que no sea una reserva, un dispositivo de registro o un dispositivo de caché.

Creación de una agrupación raíz ZFS

Tenga en cuenta las siguientes requisitos de configuración de la agrupación raíz:

- En Oracle Solaris 11.1, los discos utilizados para la agrupación raíz se instalan con una etiqueta EFI (GPT) en un sistema basado en x86 o un sistema SPARC admitido con firmware compatible con GPT. O bien, una etiqueta SMI (VTOC) se aplica en un sistema basado en SPARC sin firmware compatible con GPT. El instalador de Oracle Solaris 11.1 aplica una etiqueta EFI (GPT), si es posible, y si necesita volver a crear una agrupación raíz ZFS después de la instalación, puede utilizar el siguiente comando para aplicar la etiqueta de disco EFI (GPT) y la información de inicio correcta:

  ```
  # zpool create -B rpool2 c1t0d0
  ```
- La agrupación raíz debe crearse como configuración reflejada o una configuración de un solo disco. No se pueden agregar discos adicionales para crear varios dispositivos virtuales reflejados de nivel superior mediante el comando `zpool add`, pero se puede ampliar un dispositivo virtual reflejado mediante el comando `zpool attach`.
- No se admite una configuración RAID-Z o repartida.
- Una agrupación raíz no puede tener un dispositivo de registro independiente.
- Si intenta utilizar una configuración no admitida para una agrupación raíz, verá mensajes similares a los siguientes:

  ```
  ERROR: ZFS pool <pool-name> does not support boot environments
  ```

  ```
  # zpool add -f rpool log c0t6d0s0
  cannot add to 'rpool': root pool can not have multiple vdevs or separate logs
  ```

Para obtener más información sobre cómo instalar e iniciar un sistema de archivos raíz ZFS, consulte Capítulo 4, "Gestión de componentes de la agrupación raíz ZFS".

Creación de una agrupación de almacenamiento de RAID-Z

Una agrupación de RAID-Z de paridad sencilla se crea del mismo modo que una agrupación reflejada, excepto que se utiliza la palabra clave `raidz` o `raidz1` en lugar de `mirror`. El ejemplo siguiente muestra cómo crear un grupo con un único dispositivo de RAID-Z que se compone de cinco discos:

```
# zpool create tank raidz c1t0d0 c2t0d0 c3t0d0 c4t0d0 /dev/dsk/c5t0d0
```

Este ejemplo muestra que los discos se pueden especificar con sus nombres de dispositivo abreviados o completos. `/dev/dsk/c5t0d0` y `c5t0d0` hacen referencia al mismo disco.

Puede crear una configuración de RAID-Z de paridad doble mediante la palabra clave `raidz2` o `raidz3` al crear la agrupación. Por ejemplo:

```
# zpool create tank raidz2 c1t0d0 c2t0d0 c3t0d0 c4t0d0 c5t0d0
# zpool status -v tank
  pool: tank
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          raidz2-0  ONLINE       0     0     0
            c1t0d0  ONLINE       0     0     0
            c2t0d0  ONLINE       0     0     0
            c3t0d0  ONLINE       0     0     0
            c4t0d0  ONLINE       0     0     0
            c5t0d0  ONLINE       0     0     0

errors: No known data errors

# zpool create tank raidz3 c0t0d0 c1t0d0 c2t0d0 c3t0d0 c4t0d0
c5t0d0 c6t0d0 c7t0d0 c8t0d0
# zpool status -v tank
  pool: tank
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          raidz3-0  ONLINE       0     0     0
            c0t0d0  ONLINE       0     0     0
            c1t0d0  ONLINE       0     0     0
            c2t0d0  ONLINE       0     0     0
            c3t0d0  ONLINE       0     0     0
            c4t0d0  ONLINE       0     0     0
            c5t0d0  ONLINE       0     0     0
            c6t0d0  ONLINE       0     0     0
            c7t0d0  ONLINE       0     0     0
            c8t0d0  ONLINE       0     0     0
errors: No known data errors
```

En la actualidad, en una configuración RAID-Z de ZFS son posibles las operaciones siguientes:

- Agregar a una configuración RAID-Z existente otro conjunto de discos para un dispositivo virtual de nivel superior. Para obtener más información, consulte "Agregación de dispositivos a un grupo de almacenamiento" en la página 68.
- Reemplazar uno o varios discos de una configuración RAID-Z existente si los discos de sustitución son mayores o iguales que el dispositivo que se va a reemplazar. Para obtener más información, consulte "Sustitución de dispositivos en un grupo de almacenamiento" en la página 81.

Actualmente *no* se permiten las siguientes operaciones en una configuración RAID-Z:

- Conectar un disco adicional a una configuración de RAID-Z.
- Desconectar un disco de una configuración de RAID-Z, excepto cuando se desconecta un disco que se sustituye por un disco de repuesto o cuando se necesita desconectar un disco de repuesto.
- No se puede eliminar directamente de una configuración de RAID-Z un dispositivo que no sea de registro o caché. Para esta función se presenta un RFE.

Para obtener más información sobre una configuración de RAID-Z, consulte "Configuración de grupos de almacenamiento RAID-Z" en la página 53.

Creación de una agrupación de almacenamiento de ZFS con dispositivos de registro

El registro de intención de ZFS (ZIL) se proporciona para satisfacer los requisitos de POSIX para transacciones síncronas. Por ejemplo, las bases de datos precisan con frecuencia que sus transacciones se encuentren en dispositivos de almacenamiento estables al volver de una llamada del sistema. NFS y otras aplicaciones también pueden utilizar `fsync()` para garantizar la estabilidad de los datos.

De forma predeterminada, ZIL se asigna a partir de bloques de la agrupación principal. Sin embargo, el rendimiento puede mejorar si se usan dispositivos de registro independientes, por ejemplo NVRAM o un disco dedicado.

Para saber si es apropiado configurar un dispositivo de registro de ZFS se deben tener en cuenta los puntos siguientes:

- Los dispositivos de registros para ZIL no están relacionados con los archivos del registro de la base de datos.
- Cualquier mejora en el rendimiento que haya al implementar un dispositivo de registro independiente está sujeta al tipo de dispositivo, la configuración de hardware de la aplicación y la carga de trabajo de la aplicación. Para obtener información preliminar sobre el rendimiento, consulte este blog:

  `http://blogs.oracle.com/perrin/entry/slog_blog_or_blogging_on`
- Los dispositivos de registro pueden ser reflejados o sin reflejar, pero RAID-Z no es válido para dispositivos de registro.
- Si no se refleja un dispositivo de registro independiente y falla el dispositivo que contiene el registro, el registro que se almacena vuelve a la agrupación de almacenamiento.
- Los dispositivos de registro se pueden agregar, reemplazar, eliminar, vincular, desvincular, importar y exportar como parte de la agrupación de almacenamiento de mayor tamaño.
- Puede vincular un dispositivo de registro a uno ya creado para crear un dispositivo de registro reflejado. Esta operación es idéntica a la de vincular un dispositivo en una agrupación de almacenamiento sin reflejar.

- El tamaño mínimo de un dispositivo de registro es el mismo que el de cada dispositivo en una agrupación, es decir, 64 MB. La cantidad de datos en reproducción que se puede almacenar en un dispositivo de registro es relativamente pequeña. Los bloques de registros se liberan si se ejecuta la transacción de registros (llamada del sistema).
- El tamaño máximo de un dispositivo de registro debe ser aproximadamente la mitad de la memoria física, ya que es la cantidad máxima de datos de reproducción potenciales que se pueden almacenar. Por ejemplo, si un dispositivo tiene una memoria física de 16 GB, el dispositivo de registro debería tener como máximo 8 GB.

Puede crear un dispositivo de registro ZFS durante la creación de la agrupación o una vez creada.

El ejemplo siguiente muestra cómo crear una agrupación de almacenamiento reflejada con dispositivos de registro reflejados:

```
# zpool create datap mirror c0t5000C500335F95E3d0 c0t5000C500335F907Fd0 mirror
c0t5000C500335BD117d0 c0t5000C500335DC60Fd0 log mirror c0t5000C500335E106Bd0 c0t5000C500335FC3E7d0
# zpool status datap
  pool: datap
 state: ONLINE
 scrub: none requested
config:

        NAME                      STATE     READ WRITE CKSUM
        datap                     ONLINE       0     0     0
          mirror-0                ONLINE       0     0     0
            c0t5000C500335F95E3d0  ONLINE       0     0     0
            c0t5000C500335F907Fd0  ONLINE       0     0     0
          mirror-1                ONLINE       0     0     0
            c0t5000C500335BD117d0  ONLINE       0     0     0
            c0t5000C500335DC60Fd0  ONLINE       0     0     0
        logs
          mirror-2                ONLINE       0     0     0
            c0t5000C500335E106Bd0  ONLINE       0     0     0
            c0t5000C500335FC3E7d0  ONLINE       0     0     0

errors: No known data errors
```

Para obtener información sobre la recuperación de un error en un dispositivo de registro, consulte el Ejemplo 10–2.

Creación de una agrupación de almacenamiento de ZFS con dispositivos caché

Los dispositivos de caché ofrecen un nivel adicional de grabación de datos en caché entre la memoria principal y el disco. El uso de dispositivos caché optimiza el rendimiento con cargas de trabajo de lectura aleatorias de contenido principalmente estático.

Puede crear una agrupación de almacenamiento con dispositivos para guardar en caché datos de la agrupación de almacenamiento. Por ejemplo:

```
# zpool create tank mirror c2t0d0 c2t1d0 c2t3d0 cache c2t5d0 c2t8d0
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c2t0d0  ONLINE       0     0     0
            c2t1d0  ONLINE       0     0     0
            c2t3d0  ONLINE       0     0     0
        cache
          c2t5d0    ONLINE       0     0     0
          c2t8d0    ONLINE       0     0     0

errors: No known data errors
```

Tras agregar los dispositivos de la caché, gradualmente se llenan con contenido de la memoria principal. Según el tamaño del dispositivo de la caché, puede llevar más de una hora en llenarse. La capacidad y las lecturas se pueden supervisar con el comando `zpool iostat` del modo siguiente:

```
# zpool iostat -v pool 5
```

Los dispositivos caché se pueden agregar o quitar de una agrupación después de crearse dicha agrupación.

Tenga en cuenta los siguientes puntos antes de decidir si se debe crear una agrupación de almacenamiento de ZFS con dispositivos caché:

- El uso de dispositivos caché optimiza el rendimiento con cargas de trabajo de lectura aleatorias de contenido principalmente estático.
- La capacidad y las lecturas se pueden supervisar mediante el comando `zpool iostat`.
- Se pueden agregar varios dispositivos caché cuando se crea la agrupación. Asimismo se pueden agregar y eliminar después de crearse la agrupación. Para obtener más información, consulte el Ejemplo 3–4.
- Los dispositivos caché no se pueden reflejar ni pueden formar parte de una configuración de RAID-Z.
- Si se encuentra un error de lectura en un dispositivo caché, la E/S de lectura se vuelve a enviar al dispositivo de agrupación de almacenamiento original, que puede formar parte de una configuración de RAID-Z o reflejada. El contenido de los dispositivos caché se considera volátil, de forma similar a otras memorias caché del sistema.

Precauciones para la creación de grupos de almacenamiento

Revise las precauciones siguientes al crear y gestionar grupos de almacenamiento de ZFS.

- No vuelva a crear particiones o etiquetar discos que forman parte de una agrupación de almacenamiento existente. Si intenta volver a crear la partición o etiquetar un disco de la agrupación raíz, es posible que tenga que volver a instalar el sistema operativo.
- No cree un grupo de almacenamiento que contenga componentes de otra agrupación de almacenamiento, como archivos o volúmenes. Se pueden producir bloqueos en esta configuración no admitida.
- Una agrupación creada con un único segmento o disco no tiene redundancia y tiene riesgo de pérdida de datos. Una agrupación creada con varios segmentos pero sin redundancia también tiene riesgo de pérdida de datos. Una agrupación creada con varios segmentos entre discos resulta más difícil de administrar que una agrupación creada con discos completos.
- Una agrupación que no se haya creado con redundancia de ZFS (RAIDZ o reflejo) sólo puede generar informes de las incoherencias de datos. No puede reparar incoherencias de datos.
- Aunque una agrupación creada con redundancia de ZFS puede ayudar a reducir el tiempo de inactividad debido a fallos de hardware, no es inmune a fallos de hardware, fallos de energía o cables desconectados. Asegúrese de que se realicen copias de seguridad de los datos de forma regular. Es importante realizar copias de seguridad de rutina de los datos de la agrupación en hardware de grado no empresarial.
- Una agrupación no se puede compartir entre sistemas. El ZFS no es un sistema de archivos de clúster.

Visualización de información de dispositivos virtuales de agrupaciones de almacenamiento

Cada agrupación de almacenamiento contiene uno o más dispositivos virtuales. Un *dispositivo virtual* es una representación interna de la agrupación de almacenamiento que describe la disposición del almacenamiento físico y sus características predeterminadas. Un dispositivo virtual representa los archivos o dispositivos de disco que se utilizan para crear la agrupación de almacenamiento. Una agrupación puede tener en la parte superior de la configuración cualquier cantidad de dispositivos virtuales, denominados *vdev de nivel superior*.

Si el dispositivo virtual de nivel superior contiene dos o más dispositivos físicos, la configuración ofrece redundancia de datos como reflejo o dispositivo virtual RAID-Z. Estos dispositivos virtuales se componen de discos, segmentos de discos o archivos. Un repuesto es un caso especial de dispositivo virtual que hace un seguimiento de repuestos disponibles para una agrupación.

El ejemplo siguiente muestra cómo crear una agrupación formada por dos dispositivos virtuales de nivel superior, cada uno de los cuales es un reflejo de dos discos:

```
# zpool create tank mirror c1d0 c2d0 mirror c3d0 c4d0
```

El ejemplo siguiente muestra cómo crear una agrupación formada por un dispositivo virtual de nivel superior de cuatro discos:

```
# zpool create mypool raidz2 c1d0 c2d0 c3d0 c4d0
```

Se puede agregar otro dispositivo virtual de nivel superior a esta agrupación mediante el comando `zpool add`. Por ejemplo:

```
# zpool add mypool raidz2 c2d1 c3d1 c4d1 c5d1
```

Los discos, segmentos de discos o archivos que se utilizan en agrupaciones no redundantes funcionan como dispositivos virtuales de nivel superior. Los grupos de almacenamiento suelen contener diversos dispositivos virtuales de nivel superior. ZFS reparte dinámicamente los discos en bandas entre todos los dispositivos virtuales de nivel superior en una agrupación.

Los dispositivos virtuales y físicos que se incluyen en una agrupación de almacenamiento de ZFS se muestran con el comando `zpool status`. Por ejemplo:

```
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c0t1d0  ONLINE       0     0     0
            c1t1d0  ONLINE       0     0     0
          mirror-1  ONLINE       0     0     0
            c0t2d0  ONLINE       0     0     0
            c1t2d0  ONLINE       0     0     0
          mirror-2  ONLINE       0     0     0
            c0t3d0  ONLINE       0     0     0
            c1t3d0  ONLINE       0     0     0

errors: No known data errors
```

Administración de errores de creación de agrupaciones de almacenamiento de ZFS

Los errores de creación de grupos pueden deberse a diversos motivos. Algunos de ellos son obvios (por ejemplo, un dispositivo especificado que no existe), mientras que otros no lo son tanto.

Detección de dispositivos en uso

Antes de dar formato a un dispositivo, ZFS determina si el disco lo está utilizando ZFS o cualquier otro componente del sistema operativo. Si el disco está en uso, puede haber errores como el siguiente:

```
# zpool create tank c1t0d0 c1t1d0
invalid vdev specification
use '-f' to override the following errors:
/dev/dsk/c1t0d0s0 is currently mounted on /. Please see umount(1M).
/dev/dsk/c1t0d0s1 is currently mounted on swap. Please see swap(1M).
/dev/dsk/c1t1d0s0 is part of active ZFS pool zeepool. Please see zpool(1M).
```

Algunos errores pueden omitirse mediante la opción `-f`, pero no es algo aplicable a la mayoría. Las condiciones siguientes no pueden omitirse mediante la opción `-f`; se deben corregir manualmente:

Sistema de archivos montado — El disco o uno de sus segmentos contiene un sistema de archivos que está montado. Para corregir este error, utilice el comando `umount`.

Sistema de archivos en /etc/vfstab — El disco contiene un sistema de archivos que se muestra en el archivo `/etc/vfstab`, pero el sistema de archivos no está montado. Para corregir este error, suprima la línea del archivo `/etc/vfstab` o conviértala en comentario.

Dispositivo de volcado dedicado — El disco se utiliza como dispositivo de volcado dedicado para el sistema. Para corregir este error, utilice el comando `dumpadm`.

Parte de una agrupación de ZFS — El disco o archivo es parte de una agrupación de almacenamiento de ZFS activa. Para corregir este error, utilice el comando `zpool destroy` para destruir la otra agrupación, si ya no se necesita. También puede utilizar el comando `zpool detach` para desvincular el disco de la otra agrupación. Sólo se puede desvincular un disco de una agrupación de almacenamiento reflejada.

Las siguientes comprobaciones en uso son advertencias útiles; se pueden anular mediante la opción `-f` para crear la agrupación:

Contiene un sistema de archivos — El disco contiene un sistema de archivos conocido, aunque no está montado y no parece que se utilice.

Parte de volumen — El disco es parte de un volumen de Solaris Volume Manager.

Parte de grupo de ZFS exportado — El disco es parte de una agrupación de almacenamiento que se ha exportado o suprimido manualmente de un sistema. En el último caso, se informa de que la agrupación es `potentially active`, ya que el disco quizá sea o no una unidad conectada a la red que otro sistema utiliza. Actúe con precaución al anular una agrupación potencialmente activa.

El ejemplo siguiente muestra la forma de utilizar la opción `-f`:

```
# zpool create tank c1t0d0
invalid vdev specification
use '-f' to override the following errors:
/dev/dsk/c1t0d0s0 contains a ufs filesystem.
# zpool create -f tank c1t0d0
```

En lugar de utilizar la opción `-f` es preferible corregir los errores.

Niveles de replicación no coincidentes

No se recomienda crear agrupaciones con dispositivos virtuales de niveles de repetición diferentes. El comando `zpool` impide la creación involuntaria de una agrupación con niveles de redundancia que no coinciden. Si intenta crear un grupo con una configuración de ese tipo, aparecen errores similares al siguiente:

```
# zpool create tank c1t0d0 mirror c2t0d0 c3t0d0
invalid vdev specification
use '-f' to override the following errors:
mismatched replication level: both disk and mirror vdevs are present
# zpool create tank mirror c1t0d0 c2t0d0 mirror c3t0d0 c4t0d0 c5t0d0
invalid vdev specification
use '-f' to override the following errors:
mismatched replication level: 2-way mirror and 3-way mirror vdevs are present
```

Puede anular estos errores con la opción `-f`, pero debería evitar esta práctica. El comando también advierte sobre la creación de una agrupación de RAID-Z o reflejada mediante dispositivos de diversos tamaños. Aunque esta configuración se permite, los niveles sin correspondencia de redundancia generan espacio sin usar en disco en el dispositivo de mayor tamaño. Se necesita la opción `-f` para anular la advertencia.

Ensayo de creación de una agrupación de almacenamiento

Los intentos de creación de agrupación pueden fallar de modo imprevisto y formas diferentes; la aplicación de formato a discos es una acción potencialmente perjudicial. Por ello, el comando `zpool create` tiene la opción adicional `-n` que simula la creación de la agrupación sin escribir realmente en el dispositivo. Esta opción de *ensayo* realiza la comprobación del dispositivo en uso y la validación de nivel de repetición, y notifica si se producen errores en el proceso. Si no se encuentran errores, se genera una salida similar a la siguiente:

```
# zpool create -n tank mirror c1t0d0 c1t1d0
would create 'tank' with the following layout:

        tank
          mirror
            c1t0d0
            c1t1d0
```

Algunos errores no se pueden detectar sin crear el grupo. El ejemplo más habitual es especificar el mismo dispositivo dos veces en la misma configuración. Este error puede pasar desapercibido si no se escriben los datos, por lo que el comando `zpool create -n` podría notificar que la operación es correcta y aun así no conseguir crear la agrupación cuando el comando se ejecuta sin esta opción.

Punto de montaje predeterminado para agrupaciones de almacenamiento

Cuando se crea una agrupación, el punto de montaje predeterminado del sistema de archivos de nivel superior es */pool-name*. Este directorio no debe existir o debe estar vacío. Si el directorio no existe, se crea automáticamente. Si está vacío, el sistema de archivos raíz se monta en la parte superior del directorio ya creado. Para crear una agrupación con un punto de montaje predeterminado diferente, utilice la opción `-m` del comando `zpool create`. Por ejemplo:

```
# zpool create home c1t0d0
default mountpoint '/home' exists and is not empty
use '-m' option to provide a different default
# zpool create -m /export/zfs home c1t0d0
```

Este comando crea la nueva agrupación `home` y el sistema de archivos `home` con un punto de montaje de `/export/zfs`.

Para obtener más información sobre los puntos de montaje, consulte "Administración de puntos de montaje de ZFS" en la página 183.

Destrucción de agrupaciones de almacenamiento de ZFS

Para destruir agrupaciones se utiliza el comando `zpool destroy`. Este comando destruye la agrupación aunque contenga conjuntos de datos montados.

```
# zpool destroy tank
```

Precaución – Ponga el máximo cuidado al destruir una agrupación. Asegúrese de que se va a destruir la agrupación correcta y guarde siempre copias de los datos. Si destruye involuntariamente el grupo incorrecto, puede intentar su recuperación. Para obtener más información, consulte "Recuperación de agrupaciones de almacenamiento de ZFS destruidas" en la página 114.

Cuando se destruye una agrupación con el comando `zpool destroy`, la agrupación sigue estando disponible para importar según se describe en "Recuperación de agrupaciones de

almacenamiento de ZFS destruidas" en la página 114. Esto significa que los datos confidenciales pueden todavía estar disponibles en los discos que formaban parte de la agrupación. Si desea destruir los datos en los discos de la agrupación destruida, debe utilizar una función, como la opción `analyze->purge` de la utilidad `format` en todos los discos de la agrupación destruida.

Otra opción para mantener confidenciales los datos del sistema de archivos es crear sistemas de archivos de ZFS cifrados. Cuando una agrupación con un sistema de archivos cifrado se destruye, no se podrá acceder a los datos sin las claves de cifrado, incluso si se recuperara la agrupación. Para obtener más información, consulte "Cifrado de sistemas de archivos ZFS" en la página 205.

Destrucción de una agrupación con dispositivos no disponibles

La destrucción de una agrupación requiere que los datos se escriban en el disco para indicar que la agrupación ya no es válida. Esta información del estado impide que los dispositivos aparezcan como un grupo potencial cuando efectúa una importación. Si uno o más dispositivos dejan de estar disponibles, el grupo todavía puede destruirse. Pero la información de estado necesaria no se escribirá en estos dispositivos no disponibles.

Cuando se reparan adecuadamente, estos dispositivos se notifican como *potencialmente activos* cuando se crea una agrupación. Se incluyen como dispositivos válidos si se buscan agrupaciones para importar. Si una agrupación tiene tantos dispositivos `UNAVAIL` que la propia agrupación tiene el estado `UNAVAIL` (lo que significa que el dispositivo virtual de nivel superior tiene el estado `UNAVAIL`), el comando imprime una advertencia y no se puede completar sin la opción `-f`. Esta opción es necesaria porque la agrupación no se puede abrir, de manera que no se sabe si los datos están o no almacenados allí. Por ejemplo:

```
# zpool destroy tank
cannot destroy 'tank': pool is faulted
use '-f' to force destruction anyway
# zpool destroy -f tank
```

Para obtener más información sobre la situación de dispositivos y agrupaciones, consulte "Cómo determinar el estado de las agrupaciones de almacenamiento de ZFS" en la página 101.

Para obtener más información sobre importación de agrupaciones, consulte "Importación de grupos de almacenamiento de ZFS" en la página 110.

Administración de dispositivos en agrupaciones de almacenamiento de ZFS

La mayor parte de la información básica relacionada con los dispositivos se puede consultar en "Componentes de una agrupación de almacenamiento de ZFS" en la página 47. Después de crear una agrupación, puede efectuar diversas tareas para administrar los dispositivos físicos en ella.

Agregación de dispositivos a un grupo de almacenamiento

Puede agregar espacio en el disco a una agrupación de forma dinámica, incorporando un nuevo dispositivo virtual de nivel superior. Este espacio está inmediatamente disponible para todos los conjuntos de datos de la agrupación. Para agregar un dispositivo virtual a una agrupación, utilice el comando `zpool add`. Por ejemplo:

```
# zpool add zeepool mirror c2t1d0 c2t2d0
```

El formato para especificar dispositivos virtuales es el mismo que para el comando `zpool create`. Los dispositivos se comprueban para determinar si se utilizan y el comando no puede cambiar el nivel de redundancia sin la opción `-f`. El comando también es compatible con la opción `-n` de manera que puede ejecutar un ensayo. Por ejemplo:

```
# zpool add -n zeepool mirror c3t1d0 c3t2d0
would update 'zeepool' to the following configuration:
      zeepool
        mirror
            c1t0d0
            c1t1d0
        mirror
            c2t1d0
            c2t2d0
        mirror
            c3t1d0
            c3t2d0
```

La sintaxis de este comando agregaría dispositivos reflejados `c3t1d0` y `c3t2d0` a la configuración existente de la agrupación `zeepool`.

Para obtener más información sobre cómo validar dispositivos virtuales, consulte "Detección de dispositivos en uso" en la página 63.

EJEMPLO 3–1 Agregación de discos a una configuración de ZFS reflejada

En el siguiente ejemplo, se agrega otro reflejo a una configuración del ZFS reflejado existente.

```
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c0t1d0  ONLINE       0     0     0
            c1t1d0  ONLINE       0     0     0
          mirror-1  ONLINE       0     0     0
            c0t2d0  ONLINE       0     0     0
            c1t2d0  ONLINE       0     0     0

errors: No known data errors
# zpool add tank mirror c0t3d0 c1t3d0
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c0t1d0  ONLINE       0     0     0
            c1t1d0  ONLINE       0     0     0
          mirror-1  ONLINE       0     0     0
            c0t2d0  ONLINE       0     0     0
            c1t2d0  ONLINE       0     0     0
          mirror-2  ONLINE       0     0     0
            c0t3d0  ONLINE       0     0     0
            c1t3d0  ONLINE       0     0     0

errors: No known data errors
```

EJEMPLO 3–2 Agregación de discos a una configuración de RAID-Z

Se pueden agregar discos adicionales de modo similar a una configuración de RAID-Z. El ejemplo siguiente muestra cómo convertir una agrupación de almacenamiento con un dispositivo RAID–Z que contiene tres discos en una agrupación de almacenamiento con dos dispositivos RAID-Z con tres discos cada uno.

EJEMPLO 3–2 Agregación de discos a una configuración de RAID-Z *(Continuación)*

```
# zpool status rzpool
  pool: rzpool
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        rzpool      ONLINE       0     0     0
          raidz1-0  ONLINE       0     0     0
            c1t2d0  ONLINE       0     0     0
            c1t3d0  ONLINE       0     0     0
            c1t4d0  ONLINE       0     0     0

errors: No known data errors
# zpool add rzpool raidz c2t2d0 c2t3d0 c2t4d0
# zpool status rzpool
  pool: rzpool
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        rzpool      ONLINE       0     0     0
          raidz1-0  ONLINE       0     0     0
            c1t0d0  ONLINE       0     0     0
            c1t2d0  ONLINE       0     0     0
            c1t3d0  ONLINE       0     0     0
          raidz1-1  ONLINE       0     0     0
            c2t2d0  ONLINE       0     0     0
            c2t3d0  ONLINE       0     0     0
            c2t4d0  ONLINE       0     0     0

errors: No known data errors
```

EJEMPLO 3–3 Agregación y eliminación de un dispositivo de registro reflejado

El siguiente ejemplo muestra cómo agregar un dispositivo de registro reflejado a una agrupación de almacenamiento reflejada.

```
# zpool status newpool
  pool: newpool
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        newpool     ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c0t4d0  ONLINE       0     0     0
            c0t5d0  ONLINE       0     0     0

errors: No known data errors
# zpool add newpool log mirror c0t6d0 c0t7d0
# zpool status newpool
```

EJEMPLO 3–3 Agregación y eliminación de un dispositivo de registro reflejado *(Continuación)*

```
  pool: newpool
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        newpool     ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c0t4d0  ONLINE       0     0     0
            c0t5d0  ONLINE       0     0     0
        logs
          mirror-1  ONLINE       0     0     0
            c0t6d0  ONLINE       0     0     0
            c0t7d0  ONLINE       0     0     0

errors: No known data errors
```

Puede vincular un dispositivo de registro a uno ya creado para crear un dispositivo de registro reflejado. Esta operación es idéntica a la de conectar un dispositivo en una agrupación de almacenamiento sin reflejar.

Puede eliminar los dispositivos de registro mediante el comando `zpool remove`. El dispositivo de registro reflejado en el ejemplo anterior se puede eliminar mediante la especificación del argumento `mirror-1`. Por ejemplo:

```
# zpool remove newpool mirror-1
# zpool status newpool
  pool: newpool
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        newpool     ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c0t4d0  ONLINE       0     0     0
            c0t5d0  ONLINE       0     0     0

errors: No known data errors
```

Si su configuración de agrupación sólo contiene un dispositivo de registro, para eliminarlo tendrá que especificar el nombre del dispositivo. Por ejemplo:

```
# zpool status pool
  pool: pool
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        pool        ONLINE       0     0     0
          raidz1-0  ONLINE       0     0     0
```

EJEMPLO 3–3 Agregación y eliminación de un dispositivo de registro reflejado *(Continuación)*

```
            c0t8d0  ONLINE       0     0     0
            c0t9d0  ONLINE       0     0     0
        logs
          c0t10d0   ONLINE       0     0     0

errors: No known data errors
# zpool remove pool c0t10d0
```

EJEMPLO 3–4 Cómo agregar y eliminar dispositivos caché

Puede agregar dispositivos de caché a la agrupación de almacenamiento de ZFS y eliminarlos si dejan de ser necesarios.

Utilice el comando `zpool add` para agregar dispositivos caché. Por ejemplo:

```
# zpool add tank cache c2t5d0 c2t8d0
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c2t0d0  ONLINE       0     0     0
            c2t1d0  ONLINE       0     0     0
            c2t3d0  ONLINE       0     0     0
        cache
          c2t5d0    ONLINE       0     0     0
          c2t8d0    ONLINE       0     0     0

errors: No known data errors
```

Los dispositivos caché no se pueden reflejar ni pueden formar parte de una configuración de RAID-Z.

Utilice el comando `zpool remove` para eliminar dispositivos caché. Por ejemplo:

```
# zpool remove tank c2t5d0 c2t8d0
# zpool status tank
 pool: tank
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c2t0d0  ONLINE       0     0     0
            c2t1d0  ONLINE       0     0     0
            c2t3d0  ONLINE       0     0     0
```

EJEMPLO 3–4 Cómo agregar y eliminar dispositivos caché *(Continuación)*

```
errors: No known data errors
```

Actualmente, el comando `zpool remove` sólo admite la eliminación de dispositivos caché, dispositivos de registro y repuestos en marcha. Los dispositivos que forman parte de la configuración de la agrupación reflejada principal se pueden eliminar mediante el comando `zpool detach`. Los dispositivos no redundantes y de RAID-Z no se pueden eliminar de una agrupación.

Para obtener más información sobre cómo utilizar dispositivos caché en una agrupación de almacenamiento de ZFS, consulte "Creación de una agrupación de almacenamiento de ZFS con dispositivos caché" en la página 60.

Conexión y desconexión de dispositivos en una agrupación de almacenamiento

Además del comando `zpool add`, puede utilizar el comando `zpool attach` para agregar un nuevo dispositivo a un dispositivo reflejado o no reflejado existente.

Si va a conectar un disco para crear una agrupación raíz reflejada, consulte "Cómo configurar una agrupación raíz reflejada (SPARC o x86/VTOC)" en la página 125.

Si va a reemplazar un disco en una agrupación raíz ZFS, consulte "Cómo reemplazar un disco en una agrupación raíz ZFS (SPARC o x86/VTOC)" en la página 128.

EJEMPLO 3–5 Conversión de una agrupación de almacenamiento reflejada de dos vías a una reflejada de tres vías

En este ejemplo, `zeepool` es un reflejo de dos vías que se transforma en uno de tres vías mediante la conexión del nuevo dispositivo `c2t1d0` a `c1t1d0`, el que ya existía.

```
# zpool status zeepool
  pool: zeepool
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        zeepool     ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c0t1d0  ONLINE       0     0     0
            c1t1d0  ONLINE       0     0     0

errors: No known data errors
# zpool attach zeepool c1t1d0 c2t1d0
# zpool status zeepool
```

EJEMPLO 3–5 Conversión de una agrupación de almacenamiento reflejada de dos vías a una reflejada de tres vías *(Continuación)*

```
  pool: zeepool
 state: ONLINE
 scrub: resilver completed after 0h0m with 0 errors on Fri Jan  8 12:59:20 2010
config:

        NAME        STATE     READ WRITE CKSUM
        zeepool     ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c0t1d0  ONLINE       0     0     0
            c1t1d0  ONLINE       0     0     0
            c2t1d0  ONLINE       0     0     0  592K resilvered

errors: No known data errors
```

Si el dispositivo existente forma parte de un reflejo de tres vías, al conectar el nuevo dispositivo se crea un reflejo de cuatro vías, y así sucesivamente. En cualquier caso, el nuevo dispositivo comienza inmediatamente la actualización de la duplicación.

EJEMPLO 3–6 Conversión de una agrupación de almacenamiento de ZFS no redundante a una de ZFS reflejada

También se puede convertir una agrupación de almacenamiento no redundante en una redundante mediante el comando `zpool attach`. Por ejemplo:

```
# zpool create tank c0t1d0
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: none requested
config:
        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          c0t1d0    ONLINE       0     0     0

errors: No known data errors
# zpool attach tank c0t1d0 c1t1d0
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: resilver completed after 0h0m with 0 errors on Fri Jan  8 14:28:23 2010
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c0t1d0  ONLINE       0     0     0
            c1t1d0  ONLINE       0     0     0  73.5K resilvered

errors: No known data errors
```

Puede utilizar el comando `zpool detach` para desconectar un dispositivo de una agrupación de almacenamiento reflejada. Por ejemplo:

```
# zpool detach zeepool c2t1d0
```

Pero esta operación fallará si no hay ninguna otra réplica válida de los datos. Por ejemplo:

```
# zpool detach newpool c1t2d0
cannot detach c1t2d0: only applicable to mirror and replacing vdevs
```

Creación de una nueva agrupación mediante la división de una agrupación de almacenamiento de ZFS reflejada

Una agrupación de almacenamiento ZFS reflejada puede ser rápidamente clonada como una agrupación de copia de seguridad mediante el comando `zpool split`. Puede utilizar esta función para dividir una agrupación raíz reflejada, pero la agrupación dividida no se puede iniciar hasta realizar pasos adicionales.

Puede utilizar el comando `zpool split` para desconectar uno o varios discos de una agrupación de almacenamiento ZFS reflejada para crear una nueva agrupación con los discos desconectados. La nueva agrupación tendrá el mismo contenido que la agrupación original de almacenamiento de ZFS reflejada.

De manera predeterminada, una operación `zpool split` en una agrupación reflejada desvincula el último disco de la agrupación recién creada. Después de la operación de división, importe la nueva agrupación. Por ejemplo:

```
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c1t0d0  ONLINE       0     0     0
            c1t2d0  ONLINE       0     0     0

errors: No known data errors
# zpool split tank tank2
# zpool import tank2
# zpool status tank tank2
  pool: tank
 state: ONLINE
 scrub: none requested
config:
```

```
        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          c1t0d0    ONLINE       0     0     0

errors: No known data errors

  pool: tank2
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        tank2       ONLINE       0     0     0
          c1t2d0    ONLINE       0     0     0

errors: No known data errors
```

Puede identificar qué disco utilizar para la nueva agrupación especificando ésta con el comando `zpool split`. Por ejemplo:

```
# zpool split tank tank2 c1t0d0
```

Antes de que se produzca la división, los datos en memoria se vaciarán en los discos reflejados. Después de vaciarse los datos, el disco se desconecta de la agrupación y se le asigna un nuevo GUID de agrupación. Se genera un nuevo GUID para permitir la importación de la agrupación en el mismo sistema en que se ha dividido.

Si la agrupación que se va a dividir tiene puntos de montaje de sistema de archivos no predeterminados y la nueva agrupación se crea en el mismo sistema, deberá usar la opción `zpool split -R` para identificar un directorio raíz alternativo para la nueva agrupación, a fin de evitar conflictos entre los puntos de montaje existentes. Por ejemplo:

```
# zpool split -R /tank2 tank tank2
```

Si no utiliza la opción de `zpool split -R` y observa que hay conflictos entre puntos de montaje al intentar importar la nueva agrupación, impórtela utilizando la opción `-R`. Si la nueva agrupación se crea en un sistema distinto, no debería ser preciso especificar un directorio raíz alternativo a menos que haya conflictos de puntos de montaje.

Tenga en cuenta lo siguiente antes de utilizar la función `zpool split`:

- Esta función no está disponible para una configuración RAID-Z o una agrupación no redundante de varios discos.
- Antes de intentar una operación `zpool split`, no debería haber activas operaciones de aplicación ni datos.
- Una agrupación no se puede dividir si la actualización de duplicación está en curso.

- La división de una agrupación reflejada es óptima cuando la agrupación está compuesta por dos o tres discos y el último disco de la agrupación original se utiliza para crear la nueva agrupación. Luego, puede usar el comando `zpool attach` para volver a crear la agrupación de almacenamiento reflejada original o para convertir la agrupación recién creada en una agrupación de almacenamiento reflejada. Actualmente no existe ningún método para crear una *nueva* agrupación reflejada a partir de una agrupación reflejada *existente* en una operación `zpool split`, ya que la nueva agrupación (dividida) no es redundante.
- Si la agrupación ya existente es un reflejo de tres vías, la nueva agrupación contendrá un disco después de la operación de división. Si la agrupación ya existente es un reflejo de dos vías de dos discos, el resultado son dos agrupaciones no redundantes de dos discos. Tendrá que conectar dos discos adicionales para convertir las agrupaciones no redundantes en agrupaciones reflejadas.
- Una buena forma de mantener los datos redundantes durante una operación de división consiste en dividir una agrupación de almacenamiento reflejada compuesta de tres discos de manera que la agrupación original se componga de dos discos reflejados después de la operación de división.
- Confirme que el hardware esté configurado correctamente antes de dividir una agrupación reflejada. Para obtener información sobre la confirmación de la configuración de vaciado de caché del hardware, consulte "Prácticas generales del sistema" en la página 339.

EJEMPLO 3–7 División de una agrupación de ZFS reflejada

En el siguiente ejemplo, se divide una agrupación reflejada denominada `mothership`, con tres discos. Las dos agrupaciones resultantes son la agrupación reflejada `mothership`, con dos discos, y la nueva agrupación, `luna`, con un disco. Cada agrupación tiene el mismo contenido.

La agrupación `luna` se puede importar en otro sistema para realizar copias de seguridad. Una vez finalizada la copia de seguridad, se puede destruir la agrupación `luna` y el disco se vuelve a conectar a `mothership`. A continuación, se puede repetir el proceso.

```
# zpool status mothership
  pool: mothership
 state: ONLINE
  scan: none requested
config:

        NAME                       STATE     READ WRITE CKSUM
        mothership                 ONLINE       0     0     0
          mirror-0                 ONLINE       0     0     0
            c0t5000C500335F95E3d0  ONLINE       0     0     0
            c0t5000C500335BD117d0  ONLINE       0     0     0
            c0t5000C500335F907Fd0  ONLINE       0     0     0

errors: No known data errors
# zpool split mothership luna
# zpool import luna
# zpool status mothership luna
  pool: luna
```

EJEMPLO 3–7 División de una agrupación de ZFS reflejada *(Continuación)*

```
 state: ONLINE
  scan: none requested
config:

        NAME                      STATE     READ WRITE CKSUM
        luna                      ONLINE       0     0     0
          c0t5000C500335F907Fd0   ONLINE       0     0     0

errors: No known data errors

  pool: mothership
 state: ONLINE
  scan: none requested
config:

        NAME                        STATE     READ WRITE CKSUM
        mothership                  ONLINE       0     0     0
          mirror-0                  ONLINE       0     0     0
            c0t5000C500335F95E3d0   ONLINE       0     0     0
            c0t5000C500335BD117d0   ONLINE       0     0     0

errors: No known data errors
```

Dispositivos con conexión y sin conexión en un grupo de almacenamiento

ZFS permite que los dispositivos individuales queden sin conexión o con conexión. Cuando el hardware no es fiable o no funciona adecuadamente, ZFS continúa con la lectura o la escritura de datos en el dispositivo, suponiendo que la condición es sólo temporal. Si no es temporal, es posible indicar a ZFS que termine la conexión del dispositivo para que éste se pase por alto. ZFS no envía solicitudes a un dispositivo sin conexión.

Nota – Para sustituir dispositivos no es necesario desconectarlos.

Cómo terminar la conexión de un dispositivo

Puede terminar la conexión de un dispositivo mediante el comando `zpool offline`. El dispositivo se puede especificar mediante la ruta o un nombre abreviado, si el dispositivo es un disco. Por ejemplo:

```
# zpool offline tank c0t5000C500335F95E3d0
```

Tenga en cuenta los puntos siguientes al desconectar un dispositivo:

- No puede desconectar una agrupación si, como resultado, tendrá el estado `UNAVAIL`. Por ejemplo, no puede desconectar dos dispositivos en una configuración `raidz1`, ni tampoco puede desconectar un dispositivo virtual de nivel superior.

  ```
  # zpool offline tank c0t5000C500335F95E3d0
  cannot offline c0t5000C500335F95E3d0: no valid replicas
  ```

- De modo predeterminado, el estado `OFFLINE` es persistente. El dispositivo permanece sin conexión cuando el sistema se reinicia.

  Para desconectar temporalmente un dispositivo, utilice la opción `zpool offline -t`. Por ejemplo:

  ```
  # zpool offline -t tank c1t0d0
   bringing device 'c1t0d0' offline
  ```

  Cuando el sistema se reinicia, este dispositivo vuelve automáticamente al estado `ONLINE`.

- Si un dispositivo se queda sin conexión, no se desconecta del grupo de almacenamiento. Si intenta utilizar el dispositivo sin conexión en otra agrupación, incluso después de que la agrupación original se haya destruido, aparece en pantalla un mensaje similar al siguiente:

  device `is part of exported or potentially active ZFS` *pool*`. Please see zpool(1M)`

  Si desea utilizar el dispositivo sin conexión en otra agrupación de almacenamiento después de destruir la agrupación de almacenamiento original, conecte el dispositivo y destruya la agrupación de almacenamiento original.

  Otra forma de utilizar un dispositivo de otra agrupación de almacenamiento a la vez que se mantiene la agrupación de almacenamiento original consiste en sustituir el dispositivo de la agrupación de almacenamiento original por otro equivalente. Para obtener información sobre la sustitución de dispositivos, consulte "Sustitución de dispositivos en un grupo de almacenamiento" en la página 81.

Los dispositivos sin conexión aparecen con el estado `OFFLINE` al consultar el estado de la agrupación. Para obtener información sobre cómo saber el estado del grupo, consulte "Consulta del estado de una agrupación de almacenamiento de ZFS" en la página 93.

Para obtener más información sobre la situación del dispositivo, consulte "Cómo determinar el estado de las agrupaciones de almacenamiento de ZFS" en la página 101.

Cómo conectar un dispositivo

Si se anula la conexión de un dispositivo, se puede restablecer mediante el comando `zpool online`. Por ejemplo:

```
# zpool online tank c0t5000C500335F95E3d0
```

Si se conecta un dispositivo, los datos escritos en la agrupación se vuelven a sincronizar con el dispositivo que acaba de quedar disponible. Para sustituir un disco, no se puede conectar un dispositivo. Si desconecta un dispositivo, lo reemplaza por otro e intenta conectar ese otro, permanece en el estado `UNAVAIL`.

Si intenta conectar un dispositivo `UNAVAIL`, aparece un mensaje similar al siguiente:

También puede que vea el mensaje de disco defectuoso en la consola o escrito en el archivo `/var/adm/messages`. Por ejemplo:

```
SUNW-MSG-ID: ZFS-8000-LR, TYPE: Fault, VER: 1, SEVERITY: Major
EVENT-TIME: Wed Jun 20 11:35:26 MDT 2012
PLATFORM: ORCL,SPARC-T3-4, CSN: 1120BDRCCD, HOSTNAME: tardis
SOURCE: zfs-diagnosis, REV: 1.0
EVENT-ID: fb6699c8-6bfb-eefa-88bb-81479182e3b7
DESC: ZFS device 'id1,sd@n5000c500335dc60f/a' in pool 'pond' failed to open.
AUTO-RESPONSE: An attempt will be made to activate a hot spare if available.
IMPACT: Fault tolerance of the pool may be compromised.
REC-ACTION: Use 'fmadm faulty' to provide a more detailed view of this event.
Run 'zpool status -lx' for more information. Please refer to the associated
reference document at http://support.oracle.com/msg/ZFS-8000-LR for the latest
service procedures and policies regarding this diagnosis.
```

Para obtener más información sobre cómo reemplazar un dispositivo defectuoso, consulte "Resolución de un dispositivo que no se encuentra" en la página 309.

Puede utilizar el comando `zpool online -e` para expandir una LUN. De manera predeterminada, una LUN que se agrega a una agrupación no se expande a su tamaño máximo a menos que esté activada la propiedad de agrupación `autoexpand`. Puede ampliar la LUN automáticamente por medio del comando `zpool online -e` con la LUN tanto en línea como sin conexión. Por ejemplo:

```
# zpool online -e tank c0t5000C500335F95E3d0
```

Borrado de errores de dispositivo de agrupación de almacenamiento

Si un dispositivo pierde la conexión por un error que hace que los errores aparezcan en la salida de `zpool status`, los recuentos de errores se pueden borrar con el comando `zpool clear`.

Si se especifica sin argumentos, este comando borra todos los errores de dispositivo de la agrupación. Por ejemplo:

```
# zpool clear tank
```

Si se especifican uno o más dispositivos, este comando sólo borra los errores asociados con los dispositivos especificados. Por ejemplo:

```
# zpool clear tank c0t5000C500335F95E3d0
```

Para obtener más información sobre cómo borrar errores de `zpool`, consulte "Supresión de errores transitorios" en la página 315.

Sustitución de dispositivos en un grupo de almacenamiento

Puede reemplazar un dispositivo en una agrupación de almacenamiento mediante el comando `zpool replace`.

Si se reemplaza físicamente un dispositivo por otro en la misma ubicación de una agrupación redundante, puede que sólo haga falta identificar el dispositivo sustituido. En algunos dispositivos de hardware, ZFS reconoce que el dispositivo es un disco distinto de la misma ubicación. Por ejemplo, para reemplazar un disco defectuoso (`c1t1d0`) quitándolo y colocándolo en la misma ubicación, emplee la siguiente sintaxis:

```
# zpool replace tank c1t1d0
```

Si va a reemplazar un dispositivo de una agrupación de almacenamiento con un disco de otra ubicación física, tendrá que especificar ambos dispositivos. Por ejemplo:

```
# zpool replace tank c1t1d0 c1t2d0
```

Si va a reemplazar un disco en la agrupación raíz ZFS, consulte "Cómo reemplazar un disco en una agrupación raíz ZFS (SPARC o x86/VTOC)" en la página 128.

A continuación se detalla el procedimiento básico para sustituir un disco:

1. Si es preciso, desconecte el dispositivo con el comando `zpool offline`.
2. Retire el disco que se debe reemplazar.
3. Inserte el disco nuevo.
4. Revise la salida de `format` para determinar si el disco de reemplazo es visible.

   Además, compruebe si el identificador de dispositivo ha cambiado. Si el disco de reemplazo tiene WWN, el identificador de dispositivo para el disco fallido ha cambiado.
5. Informe a ZFS que se reemplazó el disco. Por ejemplo:

   ```
   # zpool replace tank c1t1d0
   ```

   Si el disco de reemplazo tiene un identificador de dispositivo diferente, como se identificó antes, incluya el nuevo identificador de dispositivo.

   ```
   # zpool replace tank c0t5000C500335FC3E7d0 c0t5000C500335BA8C3d0
   ```
6. Conecte el disco mediante el comando `zpool online`, si es necesario.

7. Notifique a FMA que el dispositivo se ha sustituido.

   En la salida de `fmadm faulty`, identifique la cadena `zfs://pool=name/vdev=guid` en la sección `Affects:` y proporcione esa cadena como argumento para el comando `fmadm repaired`.

   ```
   # fmadm faulty
   # fmadm repaired zfs://pool=name/vdev=guid
   ```

En algunos sistemas con discos SATA, los discos se deben desconfigurar antes de desconectarlos. Si va a reemplazar un disco en la misma posición de ranura en este sistema, puede ejecutar el comando `zpool replace` del modo descrito en el primer ejemplo de esta sección.

Si desea obtener un ejemplo de cómo reemplazar un disco SATA, consulte Ejemplo 10–1.

Tenga en cuenta lo siguiente al sustituir dispositivos en una agrupación de almacenamiento de ZFS:

- Si la propiedad de agrupación `autoreplace` se configura como activada `(on)`, se aplicará formato y sustitución a cualquier dispositivo que se encuentre en la misma ubicación física que un dispositivo previamente perteneciente a la ubicación. No es necesario que utilice el comando `zpool replace` cuando esta propiedad está activada. Es posible que no todos los tipos de hardware dispongan de esta función.
- El estado `REMOVED` de la agrupación de almacenamiento se proporciona cuando se ha extraído físicamente un dispositivo o repuesto en marcha con el sistema en funcionamiento. Un dispositivo de repuesto en marcha se sustituye por el dispositivo extraído, si lo hay.
- Si un dispositivo se extrae y después se vuelve a insertar, queda conectado. Si el repuesto en marcha se activó al volverse a insertar el dispositivo, el repuesto se extrae cuando termina la operación con conexión.
- La detección automática cuando los dispositivos se extraen o insertan depende del hardware, y quizá no sea compatible en todas las plataformas. Por ejemplo, los dispositivos USB se configuran automáticamente al insertarse. Ahora bien, quizá deba utilizar el comando `cfgadm -c configure` para configurar una unidad SATA.
- Los repuestos en marcha se comprueban periódicamente para asegurarse de que tengan conexión y estén disponibles.
- El tamaño del dispositivo de sustitución debe ser igual o mayor que el disco más pequeño en una configuración de RAID-Z o reflejada.
- Cuando un dispositivo de sustitución de un tamaño mayor que el del dispositivo que va a sustituir se agrega a una agrupación, ésta no se amplía automáticamente a su tamaño máximo. El valor de la propiedad `autoexpand` determina si una LUN de sustitución se amplía a su tamaño máximo cuando el disco se agrega a la agrupación. De manera

predeterminada, la propiedad `autoexpand` está activada. Se puede activar esta propiedad para ampliar el tamaño del LUN antes o después de que se agregue el mayor LUN a la agrupación.

En el ejemplo siguiente, se sustituyen dos discos de 16 GB de una agrupación reflejada por dos discos de 72 GB. Asegúrese de que el primer dispositivo esté completamente reconstruido antes de intentar realizar la segunda sustitución del dispositivo. La propiedad `autoexpand` se activa tras las sustituciones de disco para ampliar el tamaño del disco al máximo.

```
# zpool create pool mirror c1t16d0 c1t17d0
# zpool status
  pool: pool
 state: ONLINE
 scrub: none requested
config:

        NAME         STATE     READ WRITE CKSUM
        pool         ONLINE       0     0     0
          mirror     ONLINE       0     0     0
            c1t16d0  ONLINE       0     0     0
            c1t17d0  ONLINE       0     0     0

zpool list pool
NAME   SIZE   ALLOC  FREE    CAP  HEALTH  ALTROOT
pool  16.8G  76.5K  16.7G     0%  ONLINE  -
# zpool replace pool c1t16d0 c1t1d0
# zpool replace pool c1t17d0 c1t2d0
# zpool list pool
NAME   SIZE   ALLOC  FREE    CAP  HEALTH  ALTROOT
pool  16.8G  88.5K  16.7G     0%  ONLINE  -
# zpool set autoexpand=on pool
# zpool list pool
NAME   SIZE   ALLOC  FREE    CAP  HEALTH  ALTROOT
pool  68.2G   117K  68.2G     0%  ONLINE  -
```

- La sustitución de muchos discos en una agrupación de gran tamaño tarda mucho en realizarse debido al proceso de actualizar la duplicación de los datos en los discos nuevos. Además, es recomendable ejecutar el comando `zpool scrub` entre sustituciones de discos, para asegurarse de que los dispositivos de sustitución estén operativos y que los datos se escriban correctamente.
- Si se ha sustituido automáticamente un disco fallido con un repuesto en marcha, es posible que sea necesario desconectar el repuesto después de sustituir el disco fallido. Puede utilizar el comando `zpool detach` para desconectar un repuesto en una agrupación RAID-Z o reflejada. Para obtener información sobre cómo desconectar un repuesto en marcha, consulte "Activación y desactivación de repuestos en marcha en el grupo de almacenamiento" en la página 86.

Para obtener más información sobre cómo reemplazar dispositivos, consulte "Resolución de un dispositivo que no se encuentra" en la página 309 y "Sustitución o reparación de un dispositivo dañado" en la página 313.

Designación de repuestos en marcha en la agrupación de almacenamiento

La función de repuesto permite identificar discos que se podrían utilizar para sustituir un dispositivo defectuoso en una agrupación de almacenamiento. Si un dispositivo se designa como *repuesto en marcha*, significa que no es un dispositivo activo en una agrupación. Ahora bien, si un dispositivo activo falla, el repuesto en marcha sustituye automáticamente al defectuoso.

Los dispositivos se pueden designar como repuestos en marcha de los modos siguientes:

- Cuando se crea la agrupación con el comando `zpool create`.
- Después de crear la agrupación con el comando `zpool add`.

El ejemplo siguiente muestra cómo designar dispositivos como repuestos en marcha cuando se crea la agrupación:

```
# zpool create zeepool mirror c0t5000C500335F95E3d0 c0t5000C500335F907Fd0
mirror c0t5000C500335BD117d0 c0t5000C500335DC60Fd0 spare c0t5000C500335E106Bd0 c0t5000C500335FC3E7d0
# zpool status zeepool
  pool: zeepool
 state: ONLINE
  scan: none requested
config:

        NAME                      STATE     READ WRITE CKSUM
        zeepool                   ONLINE       0     0     0
          mirror-0                ONLINE       0     0     0
            c0t5000C500335F95E3d0  ONLINE       0     0     0
            c0t5000C500335F907Fd0  ONLINE       0     0     0
          mirror-1                ONLINE       0     0     0
            c0t5000C500335BD117d0  ONLINE       0     0     0
            c0t5000C500335DC60Fd0  ONLINE       0     0     0
        spares
          c0t5000C500335E106Bd0    AVAIL
          c0t5000C500335FC3E7d0    AVAIL

errors: No known data errors
```

El ejemplo siguiente muestra cómo designar repuestos en marcha agregándolos a una agrupación después de crearla:

```
# zpool add zeepool spare c0t5000C500335E106Bd0 c0t5000C500335FC3E7d0
# zpool status zeepool
  pool: zeepool
 state: ONLINE
  scan: none requested
config:

        NAME                      STATE     READ WRITE CKSUM
        zeepool                   ONLINE       0     0     0
          mirror-0                ONLINE       0     0     0
```

```
            c0t5000C500335F95E3d0  ONLINE       0     0     0
            c0t5000C500335F907Fd0  ONLINE       0     0     0
          mirror-1                 ONLINE       0     0     0
            c0t5000C500335BD117d0  ONLINE       0     0     0
            c0t5000C500335DC60Fd0  ONLINE       0     0     0
        spares
          c0t5000C500335E106Bd0    AVAIL
          c0t5000C500335FC3E7d0    AVAIL

errors: No known data errors
```

Los repuestos en marcha se pueden suprimir de un grupo de almacenamiento mediante el comando `zpool remove`. Por ejemplo:

```
# zpool remove zeepool c0t5000C500335FC3E7d0
# zpool status zeepool
  pool: zeepool
 state: ONLINE
  scan: none requested
config:

        NAME                       STATE     READ WRITE CKSUM
        zeepool                    ONLINE       0     0     0
          mirror-0                 ONLINE       0     0     0
            c0t5000C500335F95E3d0  ONLINE       0     0     0
            c0t5000C500335F907Fd0  ONLINE       0     0     0
          mirror-1                 ONLINE       0     0     0
            c0t5000C500335BD117d0  ONLINE       0     0     0
            c0t5000C500335DC60Fd0  ONLINE       0     0     0
        spares
          c0t5000C500335E106Bd0    AVAIL

errors: No known data errors
```

No se puede suprimir un repuesto en marcha si se está utilizando en una agrupación de almacenamiento.

Tenga en cuenta lo siguiente al utilizar repuestos en marcha de ZFS:

- Actualmente, el comando `zpool remove` sólo es apto para eliminar repuestos en marcha, dispositivos caché y dispositivos de registro.
- Para agregar un disco como repuesto en marcha, el repuesto en marcha debe ser igual o mayor que el disco más grande de la agrupación. Se puede agregar un disco de repuesto de tamaño inferior. Ahora bien, al activar ese disco de repuesto de tamaño inferior, de forma automática o con el comando `zpool replace`, la operación falla y genera un mensaje de error parecido al siguiente:

  `cannot replace` *disk3* `with` *disk4*`: device is too small`
- No puede compartir un disco de repuesto entre varios sistemas.
- Tenga en cuenta que, si comparte un disco de repuesto entre dos agrupaciones de datos en el mismo sistema, debe coordinar el uso del disco de repuesto entre las dos agrupaciones. Por ejemplo, la agrupación B tiene el repuesto en uso y la agrupación A se exporta. Es posible

que la agrupación B use inadvertidamente el repuesto mientras la agrupación A se exporta. Cuando se importa la agrupación A, pueden ocasionarse daños en los datos porque ambas agrupaciones están usando el mismo disco.

- No comparta un disco de repuesto entre una agrupación raíz y una agrupación de datos.

Activación y desactivación de repuestos en marcha en el grupo de almacenamiento

Los repuestos en marcha se activan de los modos siguientes:

- Sustitución manual: sustituya un dispositivo incorrecto en una agrupación de almacenamiento con un repuesto en marcha mediante el comando `zpool replace`.
- Sustitución automática: cuando se detecta un error, un agente FMA examina la agrupación para ver si dispone de repuestos en marcha. Si es así, sustituye el dispositivo con errores por un repuesto en marcha.

  Si falla un repuesto en marcha que está en uso, el agente FMA quita el repuesto y cancela la sustitución. El agente intenta sustituir el dispositivo por otro repuesto en marcha, si lo hay. Esta función está limitada por el hecho de que el motor de diagnóstico ZFS sólo emite errores cuando un dispositivo desaparece del sistema.

  Si sustituye físicamente un dispositivo defectuoso con un repuesto activo, puede reactivar el original, pero debe desactivar el dispositivo reemplazado mediante el comando `zpool detach` para desconectar el repuesto. Si configura la propiedad de agrupación `autoreplace` como activada (`on`), el repuesto se desconecta automáticamente y vuelve a la agrupación de repuestos cuando se inserta el dispositivo nuevo y se completa la operación de conexión.

Un dispositivo `UNAVAIL` se reemplaza automáticamente si hay disponible un repuesto en marcha. Por ejemplo:

```
# zpool status -x
  pool: zeepool
 state: DEGRADED
status: One or more devices are unavailable in response to persistent errors.
        Sufficient replicas exist for the pool to continue functioning in a
        degraded state.
action: Determine if the device needs to be replaced, and clear the errors
        using 'zpool clear' or 'fmadm repaired', or replace the device
        with 'zpool replace'.
        Run 'zpool status -v' to see device specific details.
  scan: resilvered 3.15G in 0h0m with 0 errors on Thu Jun 21 16:46:19 2012
config:

        NAME                        STATE     READ WRITE CKSUM
        zeepool                     DEGRADED     0     0     0
          mirror-0                  ONLINE       0     0     0
            c0t5000C500335F95E3d0   ONLINE       0     0     0
            c0t5000C500335F907Fd0   ONLINE       0     0     0
          mirror-1                  DEGRADED     0     0     0
            c0t5000C500335BD117d0   ONLINE       0     0     0
```

```
            spare-1                 DEGRADED   449     0     0
              c0t5000C500335DC60Fd0  UNAVAIL      0     0     0
              c0t5000C500335E106Bd0  ONLINE       0     0     0
        spares
          c0t5000C500335E106Bd0      INUSE

errors: No known data errors
```

Actualmente se puede desactivar un repuesto en marcha de las siguientes maneras:

- Eliminando el repuesto de la agrupación de almacenamiento.
- Desconectando el repuesto después de la sustitución de un disco fallido. Consulte el Ejemplo 3–8.
- Intercambiando el repuesto de forma temporal o permanente con otro repuesto. Consulte el Ejemplo 3–9.

EJEMPLO 3–8 Desconexión de un repuesto en marcha después de sustituir el disco fallido

En este ejemplo, el disco fallido (`c0t5000C500335DC60Fd0`) se reemplaza físicamente y se informa a ZFS con el comando `zpool replace`.

```
# zpool replace zeepool c0t5000C500335DC60Fd0
# zpool status zeepool
  pool: zeepool
 state: ONLINE
  scan: resilvered 3.15G in 0h0m with 0 errors on Thu Jun 21 16:53:43 2012
config:

        NAME                      STATE     READ WRITE CKSUM
        zeepool                   ONLINE       0     0     0
          mirror-0                ONLINE       0     0     0
            c0t5000C500335F95E3d0  ONLINE       0     0     0
            c0t5000C500335F907Fd0  ONLINE       0     0     0
          mirror-1                ONLINE       0     0     0
            c0t5000C500335BD117d0  ONLINE       0     0     0
            c0t5000C500335DC60Fd0  ONLINE       0     0     0
        spares
          c0t5000C500335E106Bd0    AVAIL
```

Si es necesario, puede utilizar el comando `zpool detach` para devolver el repuesto en marcha a la agrupación de repuesto. Por ejemplo:

```
# zpool detach zeepool c0t5000C500335E106Bd0
```

EJEMPLO 3–9 Desconexión de un disco averiado y uso del repuesto en marcha

Si desea sustituir un disco fallido mediante un intercambio temporal o permanente del repuesto que lo está sustituyendo, desconecte el disco original (fallido). Si se sustituye el disco fallido en algún momento, se podrá agregar a la agrupación de almacenamiento como repuesto. Por ejemplo:

EJEMPLO 3–9 Desconexión de un disco averiado y uso del repuesto en marcha *(Continuación)*

```
# zpool status zeepool
  pool: zeepool
 state: DEGRADED
status: One or more devices are unavailable in response to persistent errors.
        Sufficient replicas exist for the pool to continue functioning in a
        degraded state.
action: Determine if the device needs to be replaced, and clear the errors
        using 'zpool clear' or 'fmadm repaired', or replace the device
        with 'zpool replace'.
        Run 'zpool status -v' to see device specific details.
  scan: scrub in progress since Thu Jun 21 17:01:49 2012
    1.07G scanned out of 6.29G at 220M/s, 0h0m to go
    0 repaired, 17.05% done
config:

        NAME                      STATE     READ WRITE CKSUM
        zeepool                   DEGRADED     0     0     0
          mirror-0                ONLINE       0     0     0
            c0t5000C500335F95E3d0 ONLINE       0     0     0
            c0t5000C500335F907Fd0 ONLINE       0     0     0
          mirror-1                DEGRADED     0     0     0
            c0t5000C500335BD117d0 ONLINE       0     0     0
            c0t5000C500335DC60Fd0 UNAVAIL      0     0     0
        spares
          c0t5000C500335E106Bd0   AVAIL

errors: No known data errors
# zpool detach zeepool c0t5000C500335DC60Fd0
# zpool status zeepool
  pool: zeepool
 state: ONLINE
  scan: resilvered 3.15G in 0h0m with 0 errors on Thu Jun 21 17:02:35 2012
config:

        NAME                      STATE     READ WRITE CKSUM
        zeepool                   ONLINE       0     0     0
          mirror-0                ONLINE       0     0     0
            c0t5000C500335F95E3d0 ONLINE       0     0     0
            c0t5000C500335F907Fd0 ONLINE       0     0     0
          mirror-1                ONLINE       0     0     0
            c0t5000C500335BD117d0 ONLINE       0     0     0
            c0t5000C500335E106Bd0 ONLINE       0     0     0

errors: No known data errors
(Original failed disk c0t5000C500335DC60Fd0 is physically replaced)
# zpool add zeepool spare c0t5000C500335DC60Fd0
# zpool status zeepool
  pool: zeepool
 state: ONLINE
  scan: resilvered 3.15G in 0h0m with 0 errors on Thu Jun 21 17:02:35 2012
config:

        NAME                      STATE     READ WRITE CKSUM
        zeepool                   ONLINE       0     0     0
          mirror-0                ONLINE       0     0     0
```

EJEMPLO 3–9 Desconexión de un disco averiado y uso del repuesto en marcha *(Continuación)*

```
            c0t5000C500335F95E3d0  ONLINE       0     0     0
            c0t5000C500335F907Fd0  ONLINE       0     0     0
          mirror-1                 ONLINE       0     0     0
            c0t5000C500335BD117d0  ONLINE       0     0     0
            c0t5000C500335E106Bd0  ONLINE       0     0     0
        spares
          c0t5000C500335DC60Fd0    AVAIL

errors: No known data errors
```

Después de reemplazar un disco y de desconectar el repuesto, informe a FMA que el disco está reparado.

```
# fmadm faulty
# fmadm repaired zfs://pool=name/vdev=guid
```

Administración de propiedades de agrupaciones de almacenamiento de ZFS

Utilice el comando `zpool get` para visualizar información de propiedades de almacenamiento. Por ejemplo:

```
# zpool get all zeepool
NAME     PROPERTY       VALUE                SOURCE
zeepool  allocated      6.29G                -
zeepool  altroot        -                    default
zeepool  autoexpand     off                  default
zeepool  autoreplace    off                  default
zeepool  bootfs         -                    default
zeepool  cachefile      -                    default
zeepool  capacity       1%                   -
zeepool  dedupditto     0                    default
zeepool  dedupratio     1.00x                -
zeepool  delegation     on                   default
zeepool  failmode       wait                 default
zeepool  free           550G                 -
zeepool  guid           7543986419840620672  -
zeepool  health         ONLINE               -
zeepool  listshares     off                  default
zeepool  listsnapshots  off                  default
zeepool  readonly       off                  -
zeepool  size           556G                 -
zeepool  version        34                   default
```

Con el comando `zpool set` se pueden establecer las propiedades de agrupaciones de almacenamiento. Por ejemplo:

```
# zpool set autoreplace=on zeepool
# zpool get autoreplace zeepool
NAME     PROPERTY     VALUE    SOURCE
zeepool  autoreplace  on       local
```

Si trata de establecer una propiedad de agrupación en una agrupación que esté completamente llena, aparece en pantalla un mensaje similar al siguiente:

```
# zpool set autoreplace=on tank
cannot set property for 'tank': out of space
```

Para obtener información sobre cómo prevenir problemas de capacidad de espacio de la agrupación, consulte el Capítulo 12, "Prácticas de ZFS recomendadas por Oracle Solaris".

TABLA 3–1 Descripciones de propiedades de agrupaciones ZFS

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`allocated`	Cadena	N/A	Valor de sólo lectura que identifica la cantidad de espacio de almacenamiento dentro de la agrupación que se ha asignado físicamente.
`altroot`	Cadena	`off`	Identifica un directorio raíz alternativo. Si se establece, el directorio se antepone a cualquier punto de montaje de la agrupación. Esta propiedad es apta para examinar una agrupación desconocida, si los puntos de montaje no son de confianza o en un entorno de inicio alternativo en que las rutas habituales no sean válidas.
`autoreplace`	Booleano	`off`	Controla el reemplazo automático de dispositivos. Si la propiedad se establece en `off`, la sustitución del dispositivo debe iniciarla el administrador mediante el comando `zpool replace`. Si se establece en `on`, automáticamente se da formato y se sustituye cualquier dispositivo nuevo que se detecte en la misma ubicación física que un dispositivo previamente perteneciente a la agrupación. La abreviatura de la propiedad es `replace`.
`bootfs`	Booleano	N/A	Identifica el sistema de archivos predeterminado que se puede iniciar para la agrupación raíz. En general, esta propiedad la establecen los programas de instalación.

TABLA 3–1 Descripciones de propiedades de agrupaciones ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`cachefile`	Cadena	N/A	Los controles donde se almacena en la memoria caché la configuración de agrupación. Todas las agrupaciones de la caché se importan automáticamente cuando se inicia el sistema. Sin embargo, los entornos de instalación y administración de clústeres podrían requerir el almacenamiento en caché de esta información en otra ubicación, para impedir la importación automática de las agrupaciones. Puede configurar esta propiedad para almacenar en caché información de configuración de agrupación en una ubicación distinta. Esta información se puede importar más adelante mediante el comando `zpool import -c`. La mayoría de las configuraciones ZFS no usan esta propiedad.
`capacity`	Número	N/A	Valor de sólo lectura que identifica el porcentaje del espacio de agrupación utilizado. La abreviatura de la propiedad es `cap`.
`dedupditto`	Cadena	N/A	Establece un umbral, y si el recuento de referencia para un bloque con datos duplicados eliminados supera el umbral, se almacena automáticamente otra copia ditto del bloque.
`dedupratio`	Cadena	N/A	Relación de eliminación de datos duplicados de sólo lectura alcanzada para una agrupación, expresada como multiplicador.
`delegation`	Booleano	`on`	Controla si a un usuario sin privilegios se le pueden conceder permisos de acceso que se definen para un sistema de archivos. Para más información, consulte el Capítulo 8, "Administración delegada de ZFS Oracle Solaris".

TABLA 3–1 Descripciones de propiedades de agrupaciones ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`failmode`	Cadena	`wait`	Controla el comportamiento del sistema en caso de producirse un error grave de agrupación. Esta situación suele deberse a la pérdida de conexión con el dispositivo o los dispositivos de almacenamiento subyacentes, o a un error de todos los dispositivos de la agrupación. El comportamiento de dicho evento depende de uno de estos valores: ■ `wait`: bloquea todas las solicitudes de acceso a la agrupación hasta que se restablece la conexión del dispositivo y los errores se borran mediante el comando `zpool clear`. En este estado, las operaciones de E/S de la agrupación están bloqueadas, pero las operaciones de lectura podrían ser viables. Una agrupación se mantiene en estado `wait` hasta que se resuelve el problema del dispositivo. ■ `continue`: devuelve un error EIO a cualquier solicitud de E/S nueva, pero permite lecturas en cualquier otro dispositivo que esté en buen estado. Se bloqueará cualquier solicitud pendiente de ejecución en el disco. Después de volver a colocar o conectar el dispositivo, los errores se deben eliminar con el comando `zpool clear`. ■ `panic`: imprime un mensaje en la consola y genera un volcado de bloqueo del sistema.
`free`	Cadena	N/A	Valor de sólo lectura que identifica el número de bloques aún sin asignar dentro de la agrupación.
`guid`	Cadena	N/A	Propiedad de sólo lectura que detecta el identificador exclusivo de la agrupación.
`health`	Cadena	N/A	Propiedad de sólo lectura que identifica el estado actual de la agrupación como ONLINE, DEGRADED, SUSPENDED, REMOVED.
`listshares`	Cadena	off	Controla si la información de uso compartido de esta agrupación se muestra con el comando `zfs list`. El valor predeterminado es `off`.
`listsnapshots`	Cadena	off	Controla si la información de instantánea que está asociada con esta agrupación se muestra con el comando `zfs list`. Si se desactiva esta propiedad, la información de la instantánea se puede mostrar con el comando `zfs list - t snapshot`.

TABLA 3–1 Descripciones de propiedades de agrupaciones ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`readonly`	Booleano	off	Identifica si una agrupación se puede modificar. Esta propiedad sólo se activa cuando una agrupación se ha importado en modo de sólo lectura. Si está activada, no se podrá acceder a ningún dato síncrono que exista únicamente en el registro de intento hasta que la agrupación se vuelva a importar en modo de lectura-escritura.
`size`	Número	N/A	Propiedad de sólo lectura que identifica el tamaño total de la agrupación de almacenamiento.
`version`	Número	N/A	Identifica la versión actual en disco de la agrupación. El método preferido de actualizar agrupaciones se realiza con el comando `zpool upgrade`, aunque esta propiedad se puede utilizar si se necesita una versión predeterminada por cuestiones de compatibilidad retroactiva. Esta propiedad se puede establecer en cualquier número que esté entre 1 y la versión actual indicada por el comando `zpool upgrade -v`.

Consulta del estado de una agrupación de almacenamiento de ZFS

El comando `zpool list` ofrece diversos modos de solicitar información sobre el estado de la agrupación. La información disponible suele pertenecer a una de estas tres categorías: información básica de utilización, estadística de E/S y situación. En esta sección se abordan los tres tipos de información de agrupaciones de almacenamiento.

- "Visualización de información de agrupaciones de almacenamiento de ZFS" en la página 93
- "Visualización de estadísticas de E/S de agrupaciones de almacenamiento de ZFS " en la página 98
- "Cómo determinar el estado de las agrupaciones de almacenamiento de ZFS" en la página 101

Visualización de información de agrupaciones de almacenamiento de ZFS

El comando `zpool list` es apto para mostrar información básica sobre agrupaciones.

Visualización de información relativa a todas las agrupaciones de almacenamiento o a una agrupación específica

Sin argumentos, el comando `zpool list` sólo muestra los siguientes datos para todas las agrupaciones del sistema:

```
# zpool list
NAME                    SIZE    ALLOC   FREE    CAP  HEALTH     ALTROOT
tank                   80.0G   22.3G   47.7G    28%  ONLINE     -
dozer                   1.2T    384G    816G    32%  ONLINE     -
```

La salida de este comando muestra los siguientes datos:

NAME — El nombre de la agrupación.

SIZE — El tamaño total de la agrupación, igual a la suma del tamaño de todos los dispositivos virtuales de nivel superior.

ALLOC — La cantidad de espacio físico asignada a todos los conjuntos de datos y los metadatos internos. Esta cantidad es diferente de la cantidad de espacio en el disco según se indica en el nivel del sistema de archivos.

Para obtener más información sobre la especificación del espacio disponible en el sistema de archivos, consulte "Cálculo del espacio de ZFS" en la página 34.

FREE — Cantidad de espacio sin asignar en la agrupación.

CAP (CAPACITY) — Cantidad de espacio utilizado, expresada como porcentaje del espacio total en el disco.

HEALTH — Estado actual de la agrupación.

Para obtener más información sobre la situación de la agrupación, consulte "Cómo determinar el estado de las agrupaciones de almacenamiento de ZFS" en la página 101.

ALTROOT — Raíz alternativa de la agrupación, de haberla.

Para obtener más información sobre las agrupaciones raíz alternativas, consulte "Uso de agrupaciones raíz de ZFS alternativas" en la página 294.

También puede reunir estadísticas para una agrupación determinada especificando el nombre de la agrupación. Por ejemplo:

```
# zpool list tank
NAME                    SIZE    ALLOC   FREE    CAP   HEALTH     ALTROOT
tank                   80.0G    22.3G   47.7G    28%  ONLINE     -
```

Puede utilizar las opciones de recuento e intervalo `zpool list` para recopilar estadísticas durante un período. Además, puede mostrar una indicación de hora mediante la opción `-T`. Por ejemplo:

```
# zpool list -T d 3 2
Tue Nov  2 10:36:11 MDT 2010
NAME    SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
pool   33.8G  83.5K  33.7G     0%  1.00x  ONLINE  -
```

```
rpool  33.8G  12.2G  21.5G    36%  1.00x  ONLINE  -
Tue Nov  2 10:36:14 MDT 2010
pool   33.8G  83.5K  33.7G     0%  1.00x  ONLINE  -
rpool  33.8G  12.2G  21.5G    36%  1.00x  ONLINE  -
```

Visualización de dispositivos de agrupaciones por ubicaciones físicas

Puede utilizar la opción `zpool status -l` para mostrar información sobre la ubicación física de dispositivos de agrupaciones. Es útil revisar la información de ubicación física cuando se necesita eliminar o sustituir físicamente un disco.

Además, puede utilizar el comando `fmadm add-alias` para incluir un nombre de alias de disco que lo ayude a identificar la ubicación física de los discos en su entorno. Por ejemplo:

```
# fmadm add-alias SUN-Storage-J4400.1002QCQ015 Lab10Rack5...

# zpool status -l tank
  pool: tank
 state: ONLINE
  scan: scrub repaired 0 in 0h0m with 0 errors on Fri Aug  3 16:00:35 2012
config:

        NAME                                        STATE     READ WRITE CKSUM
        tank                                        ONLINE       0     0     0
          mirror-0                                  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_02/disk  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_20/disk  ONLINE       0     0     0
          mirror-1                                  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_22/disk  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_14/disk  ONLINE       0     0     0
          mirror-2                                  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_10/disk  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_16/disk  ONLINE       0     0     0
          mirror-3                                  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_01/disk  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_21/disk  ONLINE       0     0     0
          mirror-4                                  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_23/disk  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_15/disk  ONLINE       0     0     0
          mirror-5                                  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_09/disk  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_04/disk  ONLINE       0     0     0
          mirror-6                                  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_08/disk  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_05/disk  ONLINE       0     0     0
          mirror-7                                  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_07/disk  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_11/disk  ONLINE       0     0     0
          mirror-8                                  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_06/disk  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_19/disk  ONLINE       0     0     0
          mirror-9                                  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_00/disk  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_13/disk  ONLINE       0     0     0
          mirror-10                                 ONLINE       0     0     0
```

```
            /dev/chassis/Lab10Rack5.../DISK_03/disk  ONLINE       0     0     0
            /dev/chassis/Lab10Rack5.../DISK_18/disk  ONLINE       0     0     0
        spares
          /dev/chassis/Lab10Rack5.../DISK_17/disk    AVAIL
          /dev/chassis/Lab10Rack5.../DISK_12/disk    AVAIL

errors: No known data errors
```

Visualización de estadísticas específicas de una agrupación de almacenamiento

Las estadísticas específicas se pueden solicitar mediante la opción `-o`. Esta opción ofrece informes personalizados o un modo rápido de visualizar la información pertinente. Por ejemplo, para ver sólo el nombre y el tamaño de cada agrupación, utilice la sintaxis siguiente:

```
# zpool list -o name,size
NAME                    SIZE
tank                   80.0G
dozer                   1.2T
```

Los nombres de columna corresponden a las propiedades que se enumeran en "Visualización de información relativa a todas las agrupaciones de almacenamiento o a una agrupación específica" en la página 93.

Salida de la secuencia de comandos de la agrupación de almacenamiento de ZFS

La salida predeterminada del comando `zpool list` está diseñada para mejorar la legibilidad; no es fácil de utilizar como parte de una secuencia de comandos shell. Para facilitar los usos de programación del comando, la opción `-H` es válida para suprimir encabezados de columna y separar los campos con tabuladores, en lugar de espacios. Por ejemplo, para solicitar una lista con todos los nombres de agrupaciones en el sistema debería usar esta sintaxis:

```
# zpool list -Ho name
tank
dozer
```

Aquí puede ver otro ejemplo:

```
# zpool list -H -o name,size
tank   80.0G
dozer  1.2T
```

Cómo mostrar el historial de comandos de la agrupación de almacenamiento de ZFS

ZFS registra automáticamente los comandos `zfs` y `zpool` que se ejecutan satisfactoriamente para modificar la información de estado de la agrupación. Esta información se puede mostrar mediante el comando `zpool history`.

Por ejemplo, la sintaxis siguiente muestra la salida del comando para la agrupación raíz:

```
# zpool history
History for 'rpool':
2012-04-06.14:02:55 zpool create -f rpool c3t0d0s0
2012-04-06.14:02:56 zfs create -p -o mountpoint=/export rpool/export
2012-04-06.14:02:58 zfs set mountpoint=/export rpool/export
2012-04-06.14:02:58 zfs create -p rpool/export/home
2012-04-06.14:03:03 zfs create -p -V 2048m rpool/swap
2012-04-06.14:03:08 zfs set primarycache=metadata rpool/swap
2012-04-06.14:03:09 zfs create -p -V 4094m rpool/dump
2012-04-06.14:26:47 zpool set bootfs=rpool/ROOT/s11u1 rpool
2012-04-06.14:31:15 zfs set primarycache=metadata rpool/swap
2012-04-06.14:31:46 zfs create -o canmount=noauto -o mountpoint=/var/share rpool/VARSHARE
2012-04-06.15:22:33 zfs set primarycache=metadata rpool/swap
2012-04-06.16:42:48 zfs set primarycache=metadata rpool/swap
2012-04-09.16:17:24 zfs snapshot -r rpool/ROOT@yesterday
2012-04-09.16:17:54 zfs snapshot -r rpool/ROOT@now
```

Puede utilizar una salida similar en el sistema para identificar el conjunto *exacto* de comandos de ZFS que se han ejecutado para resolver una situación de error.

Este registro de historial presenta las características siguientes:

- El registro no se puede desactivar.
- El registro se mantiene de forma persistente en el disco, lo que significa que se guarda en los reinicios del sistema.
- El registro se implementa como búfer de anillo. El tamaño mínimo es de 128 KB. El tamaño máximo es de 32 MB.
- En agrupaciones pequeñas, el tamaño máximo se restringe al 1% del tamaño de la agrupación, donde el *tamaño* se determina al crear agrupaciones.
- El registro no requiere administración; eso significa que no es necesario ajustar el tamaño del registro ni cambiar la ubicación del registro.

Para identificar el historial de comandos de una agrupación de almacenamiento específica, utilice una sintaxis similar a la siguiente:

```
# zpool history tank
2012-01-25.16:35:32 zpool create -f tank mirror c3t1d0 c3t2d0 spare c3t3d0
2012-02-17.13:04:10 zfs create tank/test
2012-02-17.13:05:01 zfs snapshot -r tank/test@snap1
```

Utilice la opción `-l` para ver el formato completo que incluye el nombre de usuario, el nombre de host y la zona en que se ha efectuado la operación. Por ejemplo:

```
# zpool history -l tank
History for 'tank':
2012-01-25.16:35:32 zpool create -f tank mirror c3t1d0 c3t2d0 spare c3t3d0
[user root on tardis:global]
2012-02-17.13:04:10 zfs create tank/test [user root on tardis:global]
2012-02-17.13:05:01 zfs snapshot -r tank/test@snap1 [user root on tardis:global]
```

Utilice la opción `-i` para ver información de eventos internos válida para tareas de diagnóstico. Por ejemplo:

```
# zpool history -i tank
History for 'tank':
2012-01-25.16:35:32 zpool create -f tank mirror c3t1d0 c3t2d0 spare c3t3d0
2012-01-25.16:35:32 [internal pool create txg:5] pool spa 33; zfs spa 33; zpl 5;
uts tardis 5.11 11.1 sun4v
2012-02-17.13:04:10 zfs create tank/test
2012-02-17.13:04:10 [internal property set txg:66094] $share2=2 dataset = 34
2012-02-17.13:04:31 [internal snapshot txg:66095] dataset = 56
2012-02-17.13:05:01 zfs snapshot -r tank/test@snap1
2012-02-17.13:08:00 [internal user hold txg:66102] <.send-4736-1> temp = 1 ...
```

Visualización de estadísticas de E/S de agrupaciones de almacenamiento de ZFS

Para solicitar estadísticas de E/S relativas a agrupaciones o dispositivos virtuales específicos, utilice el comando `zpool iostat`. Similar al comando `iostat`, este comando puede mostrar una instantánea estática de toda la actividad de E/S, así como las estadísticas actualizadas para cada intervalo especificado. Se informa de las estadísticas siguientes:

`alloc capacity`
: Cantidad de datos almacenados en la agrupación o el dispositivo. Esta cifra difiere de la cantidad de espacio disponible en los sistemas de archivos reales en una pequeña cantidad debido a detalles de implementación internos.

  Para obtener más información sobre la diferencia entre el espacio de la agrupación y el del conjunto de datos, consulte "Cálculo del espacio de ZFS" en la página 34.

`free capacity`
: Cantidad de espacio en el disco disponible en la agrupación o dispositivo. Al igual que con la estadística `used`, esta cantidad difiere por un pequeño margen de la cantidad de espacio en el disco disponible para conjuntos de datos.

`read operations`
: Número de operaciones de E/S de lectura enviadas a la agrupación o al dispositivo, incluidas las solicitudes de metadatos.

`write operations`	Número de operaciones de E/S de escritura enviadas a la agrupación o al dispositivo.
`read bandwidth`	Ancho de banda de todas las operaciones de lectura (incluidos los metadatos), expresado en unidades por segundo.
`write bandwidth`	Ancho de banda de todas las operaciones de escritura, expresadas en unidades por segundo.

Lista de estadísticas de E/S de todas las agrupaciones

Sin opciones, el comando `zpool iostat` muestra las estadísticas acumuladas desde el inicio de todos los grupos del sistema. Por ejemplo:

```
# zpool iostat
               capacity     operations    bandwidth
pool        alloc   free   read  write   read  write
----------  -----  -----  -----  -----  -----  -----
rpool       6.05G  61.9G      0      0    786    107
tank        31.3G  36.7G      4      1   296K  86.1K
----------  -----  -----  -----  -----  -----  -----
```

Como estas estadísticas se acumulan desde el inicio, el ancho de banda puede parecer bajo si la agrupación está relativamente inactiva. Para solicitar una vista más exacta del uso actual del ancho de banda, especifique un intervalo. Por ejemplo:

```
# zpool iostat tank 2
               capacity     operations    bandwidth
pool        alloc   free   read  write   read  write
----------  -----  -----  -----  -----  -----  -----
tank        18.5G  49.5G      0    187      0  23.3M
tank        18.5G  49.5G      0    464      0  57.7M
tank        18.5G  49.5G      0    457      0  56.6M
tank        18.8G  49.2G      0    435      0  51.3M
```

En el siguiente ejemplo, el comando muestra las estadísticas de uso de la agrupación `tank` cada dos segundos hasta que se pulsa Ctrl-C. Otra opción consiste en especificar un argumento `count` adicional, que hace que el comando se termine tras el número especificado iteraciones.

Por ejemplo, `zpool iostat 2 3` imprimiría un resumen cada dos segundos para tres iteraciones, durante un total de seis segundos. Si sólo hay una agrupación, las estadísticas se muestran en líneas consecutivas. Si hay más de una agrupación, la línea de guiones adicional marca cada iteración para ofrecer una separación visual.

Lista de estadísticas de E/S de dispositivos virtuales

Además de las estadísticas de E/S de todas las agrupaciones, el comando `zpool iostat` puede mostrar estadísticas de E/S para dispositivos virtuales. Este comando se puede usar para identificar dispositivos anormalmente lentos o para observar la distribución de E/S generada

por ZFS. Para solicitar toda la distribución de dispositivos virtuales, así como todas las estadísticas de E/S, utilice el comando `zpool iostat -v`. Por ejemplo:

```
# zpool iostat -v
               capacity     operations    bandwidth
pool        alloc   free   read  write   read  write
----------  -----  -----  -----  -----  -----  -----
rpool       6.05G  61.9G      0      0    785    107
  mirror    6.05G  61.9G      0      0    785    107
    c1t0d0s0    -      -      0      0    578    109
    c1t1d0s0    -      -      0      0    595    109
----------  -----  -----  -----  -----  -----  -----
tank        36.5G  31.5G      4      1   295K   146K
  mirror    36.5G  31.5G    126     45  8.13M  4.01M
    c1t2d0      -      -      0      3   100K   386K
    c1t3d0      -      -      0      3   104K   386K
----------  -----  -----  -----  -----  -----  -----
```

Tenga en cuenta dos puntos importantes al visualizar estadísticas de E/S de dispositivos virtuales:

- En primer lugar, las estadísticas de uso del espacio en el disco sólo están disponibles para dispositivos virtuales de nivel superior. El modo en que el espacio en el disco se asigna entre el reflejo y los dispositivos virtuales RAID-Z es específico de la implementación y es difícil de expresar en un solo número.
- Segundo, los números quizá no se agreguen exactamente como cabría esperar. En concreto, las operaciones en dispositivos reflejados y RAID-Z no serán exactamente iguales. Esta diferencia se aprecia sobre todo inmediatamente después de crear una agrupación, puesto que una cantidad significativa de E/S se efectúa directamente en los discos como parte de la creación de agrupaciones y no se tiene en cuenta en el nivel del reflejo. Con el tiempo se igualan estos números. Pero esta simetría se puede ver afectada si hay dispositivos defectuosos, averiados o desconectados.

Puede utilizar el mismo conjunto de opciones (interval y count) al examinar estadísticas de dispositivos virtuales.

También puede mostrar información de ubicación física sobre los dispositivos virtuales de la agrupación. Por ejemplo:

```
# zpool iostat -lv
               capacity     operations    bandwidth
pool        alloc   free   read  write   read  write
----------  -----  -----  -----  -----  -----  -----
export      2.39T  2.14T     13     27  42.7K   300K
  mirror     490G   438G      2      5  8.53K  60.3K
    /dev/chassis/lab10rack15/SCSI_Device__2/disk      -      -      1      0  4.47K  60.3K
    /dev/chassis/lab10rack15/SCSI_Device__3/disk      -      -      1      0  4.45K  60.3K
  mirror     490G   438G      2      5  8.62K  59.9K
    /dev/chassis/lab10rack15/SCSI_Device__4/disk      -      -      1      0  4.52K  59.9K
    /dev/chassis/lab10rack15/SCSI_Device__5/disk      -      -      1      0  4.48K  59.9K
  mirror     490G   438G      2      5  8.60K  60.2K
```

```
    /dev/chassis/lab10rack15/SCSI_Device__6/disk      -      -      1      0  4.50K  60.2K
    /dev/chassis/lab10rack15/SCSI_Device__7/disk      -      -      1      0  4.49K  60.2K
  mirror     490G   438G      2      5  8.47K  60.1K
    /dev/chassis/lab10rack15/SCSI_Device__8/disk      -      -      1      0  4.42K  60.1K
    /dev/chassis/lab10rack15/SCSI_Device__9/disk      -      -      1      0  4.43K  60.1K
.
.
.
```

Cómo determinar el estado de las agrupaciones de almacenamiento de ZFS

ZFS ofrece un método integrado para examinar el estado de dispositivos y agrupaciones. La situación de una agrupación la determina el estado de todos sus dispositivos. Esta información sobre el estado se obtiene con el comando `zpool status`. Además, `fmd` informa de posibles errores en dispositivos y agrupaciones, que se muestran en la consola del sistema y en el archivo `/var/adm/messages`.

Esta sección describe cómo determinar el estado de grupos y dispositivos. En este capítulo no se explica cómo reparar o recuperarse de grupos cuyo estado es defectuoso. Si desea más información sobre cómo resolver problemas y recuperar datos, consulte el Capítulo 10, “Recuperación de agrupaciones y solución de problemas de Oracle Solaris ZFS”.

El estado de una agrupación se describe mediante uno de cuatro estados:

`DEGRADED`
: Una agrupación con uno o varios dispositivos fallidos, pero cuyos datos permanecen disponibles debido a una configuración redundante.

`ONLINE`
: Una agrupación cuyos dispositivos funcionan con normalidad.

`SUSPENDED`
: Una agrupación en espera de restauración de la conectividad del dispositivo. Una agrupación `SUSPENDED` se mantiene en estado de espera hasta que se resuelve el problema del dispositivo.

`UNAVAIL`
: Una agrupación con metadatos dañados, o uno o varios dispositivos no disponibles, y réplicas insuficientes para seguir funcionando.

Cada dispositivo de agrupación puede tener uno de los estados siguientes:

`DEGRADED` El dispositivo virtual ha sufrido un fallo pero sigue funcionando. Es el estado más habitual si un dispositivo RAID-Z o un reflejo pierden uno o más dispositivos

constituyentes. La tolerancia a errores de la agrupación puede verse comprometida: un error posterior en otro dispositivo puede llegar a ser irrecuperable.

`OFFLINE` El administrador ha dejado expresamente sin conexión el dispositivo.

`ONLINE` El dispositivo o dispositivo virtual funciona normalmente. Quizá haya algunos errores transitorios, pero el dispositivo funciona.

`REMOVED` Se ha extraído físicamente el dispositivo mientras el sistema estaba ejecutándose. La detección de extracción de dispositivos depende del hardware y quizá no se admita en todas las plataformas.

`UNAVAIL` El dispositivo o dispositivo virtual no se puede abrir. En algunos casos, las agrupaciones con dispositivos en estado `UNAVAIL` se muestran en modo `DEGRADED`. Si un dispositivo virtual de nivel superior tiene estado `UNAVAIL`, la agrupación queda completamente inaccesible.

El estado de una agrupación lo determina el estado de todos sus dispositivos virtuales de nivel superior. Si todos los dispositivos virtuales están `ONLINE`, la agrupación también está `ONLINE`. Si uno de los dispositivos virtuales tiene el estado `DEGRADED` o `UNAVAIL`, la agrupación también tiene el estado `DEGRADED`. Si un dispositivo virtual de nivel superior tiene el estado `UNAVAIL` o `OFFLINE`, la agrupación también tiene el estado `UNAVAIL` o `SUSPENDED`. Una agrupación con el estado `UNAVAIL` o `SUSPENDED` es completamente inaccesible. La recuperación de datos no es factible hasta que los dispositivos necesarios se conectan o reparan. Una agrupación con estado `DEGRADED` sigue funcionando, pero quizá no obtenga el mismo nivel de redundancia o rendimiento de datos que si tuviera conexión.

El comando `zpool status` también proporciona detalles sobre operaciones de reconstrucción y limpieza de datos.

- Informe de reconstrucción en curso. Por ejemplo:

```
scan: resilver in progress since Wed Jun 20 14:19:38 2012
    7.43G scanned out of 71.8G at 36.4M/s, 0h30m to go
    7.43G resilvered, 10.35% done
```

- Informe de limpieza en curso. Por ejemplo:

```
scan: scrub in progress since Wed Jun 20 14:56:52 2012
    529M scanned out of 71.8G at 48.1M/s, 0h25m to go
    0 repaired, 0.72% done
```

- Mensaje de reconstrucción finalizada. Por ejemplo:

```
scan: resilvered 71.8G in 0h14m with 0 errors on Wed Jun 20 14:33:42 2012
```

- Mensaje de limpieza finalizada. Por ejemplo:

```
scan: scrub repaired 0 in 0h11m with 0 errors on Wed Jun 20 15:08:23 2012
```

- Mensaje de cancelación de limpieza en curso. Por ejemplo:

```
scan: scrub canceled on Wed Jun 20 16:04:40 2012
```

- Los mensajes de finalización de limpieza y reconstrucción se mantienen durante los reinicios del sistema.

Estado de la agrupación de almacenamiento básico

El modo más rápido de averiguar el estado de salud de agrupaciones consiste en usar el comando `zpool status` como se indica a continuación:

```
# zpool status -x
all pools are healthy
```

Si desea examinar una determinada agrupación, indique su nombre en la sintaxis de comando. Cualquier grupo que no esté en estado `ONLINE` debe comprobarse para descartar problemas potenciales, tal como se explica en la sección siguiente.

Estado detallado

Puede solicitar un resumen de estado más detallado mediante la opción `-v`. Por ejemplo:

```
# zpool status -v pond
  pool: pond
 state: DEGRADED
status: One or more devices are unavailable in response to persistent errors.
        Sufficient replicas exist for the pool to continue functioning in a
        degraded state.
action: Determine if the device needs to be replaced, and clear the errors
        using 'zpool clear' or 'fmadm repaired', or replace the device
        with 'zpool replace'.
  scan: scrub repaired 0 in 0h0m with 0 errors on Wed Jun 20 15:38:08 2012
config:

        NAME                      STATE     READ WRITE CKSUM
        pond                      DEGRADED     0     0     0
          mirror-0                DEGRADED     0     0     0
            c0t5000C500335F95E3d0 ONLINE       0     0     0
            c0t5000C500335F907Fd0 UNAVAIL      0     0     0
          mirror-1                ONLINE       0     0     0
            c0t5000C500335BD117d0 ONLINE       0     0     0
            c0t5000C500335DC60Fd0 ONLINE       0     0     0

device details:

        c0t5000C500335F907Fd0    UNAVAIL          cannot open
        status: ZFS detected errors on this device.
                The device was missing.
           see: http://support.oracle.com/msg/ZFS-8000-LR for recovery


errors: No known data errors
```

Esta salida muestra la descripción completa de por qué el grupo se encuentra en un estado determinado, incluida una descripción legible del problema y un vínculo a un artículo sobre la

materia para obtener más información. Cada artículo técnico ofrece información actualizada sobre el mejor método de resolución del problema actual. El uso de la información de configuración detallada permite determinar el dispositivo dañado y la forma de reparar la agrupación.

En el ejemplo anterior, debe reemplazarse el dispositivo `UNAVAIL`. Una vez reemplazado el dispositivo, utilice el comando `zpool online` para conectar el dispositivo, si es necesario. Por ejemplo:

```
# zpool online pond c0t5000C500335F907Fd0
warning: device 'c0t5000C500335DC60Fd0' onlined, but remains in degraded state
# zpool status -x
all pools are healthy
```

La salida anterior indica que el dispositivo permanece en un estado degradado hasta completar la reconstrucción.

Si la propiedad `autoreplace` está activada, es posible que no sea necesario conectar el dispositivo reemplazado.

Si una agrupación tiene un dispositivo sin conexión, la salida del comando identifica la agrupación problemática. Por ejemplo:

```
# zpool status -x
 pool: pond
 state: DEGRADED
status: One or more devices has been taken offline by the administrator.
        Sufficient replicas exist for the pool to continue functioning in a
        degraded state.
action: Online the device using 'zpool online' or replace the device with
        'zpool replace'.
config:

        NAME                      STATE     READ WRITE CKSUM
        pond                      DEGRADED     0     0     0
          mirror-0                DEGRADED     0     0     0
            c0t5000C500335F95E3d0 ONLINE       0     0     0
            c0t5000C500335F907Fd0 OFFLINE      0     0     0
          mirror-1                ONLINE       0     0     0
            c0t5000C500335BD117d0 ONLINE       0     0     0
            c0t5000C500335DC60Fd0 ONLINE       0     0     0

errors: No known data errors
```

Las columnas `READ` y `WRITE` ofrecen un recuento de errores de E/S producidos en el dispositivo; y la columna `CKSUM` ofrece un recuento de errores de suma de comprobación del dispositivo que no pueden corregirse. Ambos recuentos de errores indican un error potencial del dispositivo y las pertinentes acciones correctivas. Si se informa de que un dispositivo virtual de nivel superior tiene errores distintos de cero, quizá ya no se pueda acceder a algunas porciones de datos.

El campo `errors:` identifica cualquier error de datos conocido.

En la salida del ejemplo anterior, el dispositivo que no está conectado no provoca errores de datos.

Para obtener más información sobre el diagnóstico y la reparación de datos y agrupaciones `UNAVAIL`, consulte Capítulo 10, "Recuperación de agrupaciones y solución de problemas de Oracle Solaris ZFS".

Recopilación de información sobre el estado de la agrupación ZFS

Puede utilizar las opciones de recuento e intervalo `zpool status` para recopilar estadísticas durante un período. Además, puede mostrar una indicación de hora mediante la opción `-T`. Por ejemplo:

```
# zpool status -T d 3 2
Wed Jun 20 16:10:09 MDT 2012
  pool: pond
 state: ONLINE
  scan: resilvered 9.50K in 0h0m with 0 errors on Wed Jun 20 16:07:34 2012
config:

        NAME                       STATE     READ WRITE CKSUM
        pond                       ONLINE       0     0     0
          mirror-0                 ONLINE       0     0     0
            c0t5000C500335F95E3d0  ONLINE       0     0     0
            c0t5000C500335F907Fd0  ONLINE       0     0     0
          mirror-1                 ONLINE       0     0     0
            c0t5000C500335BD117d0  ONLINE       0     0     0
            c0t5000C500335DC60Fd0  ONLINE       0     0     0

errors: No known data errors

  pool: rpool
 state: ONLINE
  scan: scrub repaired 0 in 0h11m with 0 errors on Wed Jun 20 15:08:23 2012
config:

        NAME                         STATE     READ WRITE CKSUM
        rpool                        ONLINE       0     0     0
          mirror-0                   ONLINE       0     0     0
            c0t5000C500335BA8C3d0s0  ONLINE       0     0     0
            c0t5000C500335FC3E7d0s0  ONLINE       0     0     0

errors: No known data errors
Wed Jun 20 16:10:12 MDT 2012

  pool: pond
 state: ONLINE
  scan: resilvered 9.50K in 0h0m with 0 errors on Wed Jun 20 16:07:34 2012
config:

        NAME                       STATE     READ WRITE CKSUM
        pond                       ONLINE       0     0     0
          mirror-0                 ONLINE       0     0     0
            c0t5000C500335F95E3d0  ONLINE       0     0     0
```

```
            c0t5000C500335F907Fd0  ONLINE       0     0     0
          mirror-1                 ONLINE       0     0     0
            c0t5000C500335BD117d0  ONLINE       0     0     0
            c0t5000C500335DC60Fd0  ONLINE       0     0     0

errors: No known data errors

  pool: rpool
 state: ONLINE
  scan: scrub repaired 0 in 0h11m with 0 errors on Wed Jun 20 15:08:23 2012
config:

        NAME                          STATE     READ WRITE CKSUM
        rpool                         ONLINE       0     0     0
          mirror-0                    ONLINE       0     0     0
            c0t5000C500335BA8C3d0s0   ONLINE       0     0     0
            c0t5000C500335FC3E7d0s0   ONLINE       0     0     0

errors: No known data errors
```

Migración de agrupaciones de almacenamiento de ZFS

En ocasiones puede ser preciso mover una agrupación de almacenamiento de un sistema a otro. Para hacerlo, los dispositivos de almacenamiento se deben desconectar del sistema original y volver a conectar en el de destino. Esta tarea se debe efectuar mediante el recableado físico de los dispositivos o mediante dispositivos con varios puertos como los de una SAN. El ZFS le permite exportar la agrupación de un sistema e importarla en el de destino, incluso si los sistemas tienen un orden diferente de almacenamiento de una secuencia de datos en la memoria. Para obtener información sobre la réplica o la migración de sistemas de archivos entre diferentes agrupaciones de almacenamiento, que pueden residir en equipos distintos, consulte "Envío y recepción de datos ZFS" en la página 228.

Preparación para la migración de grupos de almacenamiento de ZFS

Los grupos de almacenamiento se deben exportar para indicar que están preparados para la migración. Esta operación purga cualquier dato no escrito en el disco, escribe datos en el disco para indicar que la exportación se ha realizado y elimina del sistema cualquier información de la agrupación.

Si no exporta la agrupación, sino que elimina manualmente los discos, aún es posible importar la agrupación resultante en otro sistema. Sin embargo, posiblemente pierda los últimos segundos de transacciones de datos y la agrupación tendrá el estado `UNAVAIL` en el sistema original debido a que los dispositivos ya no están presentes. De forma predeterminada, el sistema de destino es incapaz de importar una agrupación que no se ha exportado explícitamente. Esta condición es necesaria para impedir la importación accidental de una agrupación activa con almacenamiento conectado a la red que todavía se utilice en otro sistema.

Exportación a un grupo de almacenamiento de ZFS

Si desea exportar una agrupación, utilice el comando `zpool export`. Por ejemplo:

```
# zpool export tank
```

Antes de continuar, el comando intenta desmontar cualquier sistema de archivos montado en el grupo. Si alguno de los sistemas de archivos no consigue desmontarse, puede forzar el desmontaje mediante la opción `-f`. Por ejemplo:

```
# zpool export tank
cannot unmount '/export/home/eric': Device busy
# zpool export -f tank
```

Tras ejecutar este comando, la agrupación `tank` deja de estar visible en el sistema.

Si al exportar hay dispositivos no disponibles, no se pueden especificar como exportados correctamente. Si uno de estos dispositivos se conecta más adelante a un sistema sin uno de los dispositivos en funcionamiento, aparece como "potencialmente activo".

Si los volúmenes de ZFS se utilizan en la agrupación, ésta no se puede exportar, ni siquiera con la opción `-f`. Para exportar una agrupación con un volumen de ZFS, antes debe comprobar que no esté activo ninguno de los consumidores del volumen.

Para obtener más información sobre los volúmenes de ZFS, consulte "Volúmenes de ZFS" en la página 285.

Especificación de grupos de almacenamiento disponibles para importar

Cuando la agrupación se haya eliminado del sistema (ya sea al exportar explícitamente o eliminar dispositivos de manera forzada), conecte los dispositivos al sistema de destino. ZFS puede controlar determinadas situaciones en que sólo algunos de los dispositivos están disponibles, pero una migración de agrupaciones correcta depende de la salud global de los dispositivos. Además, no es esencial que los dispositivos estén vinculados bajo el mismo nombre de dispositivo. ZFS detecta cualquier dispositivo que se haya movido o al que se haya cambiado el nombre, y ajusta la configuración en consonancia. Para detectar las agrupaciones disponibles, ejecute el comando `zpool import` sin opciones. Por ejemplo:

```
# zpool import
 pool: tank
    id: 11809215114195894163
 state: ONLINE
action: The pool can be imported using its name or numeric identifier.
config:

        tank        ONLINE
          mirror-0  ONLINE
            c1t0d0  ONLINE
            c1t1d0  ONLINE
```

En este ejemplo, la agrupación `tank` está disponible para importarla al sistema de destino. Cada grupo está identificado mediante un nombre, así como un identificador numérico exclusivo. Si hay varias agrupaciones para importar con el mismo nombre, puede utilizar el identificador numérico para diferenciarlas.

De forma parecida a la salida del comando `zpool status`, la salida `zpool import` incluye un vínculo a un artículo divulgativo con la información más actualizada sobre procedimientos de resolución de un problema que impide la importación de una agrupación. En este caso, el usuario puede forzar la importación de un grupo. Sin embargo, importar un grupo que utiliza otro sistema en una red de almacenamiento puede dañar datos y generar avisos graves del sistema, puesto que ambos sistemas intentan escribir en el mismo almacenamiento. Si algunos dispositivos de la agrupación no están disponibles pero hay suficiente redundancia para tener una agrupación utilizable, la agrupación mostrará el estado `DEGRADED`. Por ejemplo:

```
# zpool import
 pool: tank
    id: 4715259469716913940
 state: DEGRADED
status: One or more devices are unavailable.
action: The pool can be imported despite missing or damaged devices.  The
        fault tolerance of the pool may be compromised if imported.
config:

    tank                            DEGRADED
       mirror-0                     DEGRADED
```

```
        c0t5000C500335E106Bd0      ONLINE
        c0t5000C500335FC3E7d0      UNAVAIL  cannot open

device details:

        c0t5000C500335FC3E7d0     UNAVAIL   cannot open
        status: ZFS detected errors on this device.
               The device was missing.
```

En este ejemplo, el primer disco está dañado o no se encuentra, aunque aún puede importar la agrupación porque todavía se puede acceder a los datos reflejados. Si hay demasiados dispositivos no disponibles, la agrupación no se puede importar.

En este ejemplo faltan dos discos de un dispositivo virtual RAID-Z. Eso significa que no hay suficientes datos redundantes disponibles para reconstruir la agrupación. En algunos casos no hay suficientes dispositivos para determinar la configuración completa. En este caso, ZFS desconoce los demás dispositivos que formaban parte de la agrupación, aunque ZFS proporciona todos los datos posibles relativos a la situación. Por ejemplo:

```
# zpool import
pool: mothership
    id: 3702878663042245922
 state: UNAVAIL
status: One or more devices are unavailable.
action: The pool cannot be imported due to unavailable devices or data.
config:

        mothership     UNAVAIL  insufficient replicas
          raidz1-0     UNAVAIL  insufficient replicas
            c8t0d0     UNAVAIL  cannot open
            c8t1d0     UNAVAIL  cannot open
            c8t2d0     ONLINE
            c8t3d0     ONLINE

device details:

        c8t0d0    UNAVAIL          cannot open
        status: ZFS detected errors on this device.
                The device was missing.

        c8t1d0    UNAVAIL          cannot open
        status: ZFS detected errors on this device.
                The device was missing.
```

Importación de agrupaciones de almacenamiento de ZFS de directorios alternativos

De modo predeterminado, el comando `zpool import` sólo busca dispositivos en el directorio `/dev/dsk`. Si los dispositivos existen en otro directorio, o si utiliza agrupaciones de las que se ha hecho copia de seguridad mediante archivos, utilice la opción `-d` para buscar en directorios alternativos. Por ejemplo:

```
# zpool create dozer mirror /file/a /file/b
# zpool export dozer
# zpool import -d /file
  pool: dozer
    id: 7318163511366751416
 state: ONLINE
action: The pool can be imported using its name or numeric identifier.
config:

        dozer        ONLINE
          mirror-0   ONLINE
            /file/a  ONLINE
            /file/b  ONLINE
# zpool import -d /file dozer
```

Si los dispositivos están en varios directorios, puede especificar múltiples opciones de `-d`.

Importación de grupos de almacenamiento de ZFS

Tras identificar una agrupación para importarla, debe especificar el nombre de la agrupación o su identificador numérico como argumento en el comando `zpool import`. Por ejemplo:

```
# zpool import tank
```

Si hay varias agrupaciones con el mismo nombre, indique la agrupación que desea importar mediante el identificador numérico. Por ejemplo:

```
# zpool import
  pool: dozer
    id: 2704475622193776801
 state: ONLINE
action: The pool can be imported using its name or numeric identifier.
config:

        dozer       ONLINE
          c1t9d0    ONLINE

  pool: dozer
    id: 6223921996155991199
 state: ONLINE
action: The pool can be imported using its name or numeric identifier.
config:
```

```
        dozer       ONLINE
          c1t8d0    ONLINE
# zpool import dozer
cannot import 'dozer': more than one matching pool
import by numeric ID instead
# zpool import 6223921996155991199
```

Si el nombre de la agrupación entra en conflicto con un nombre de agrupación que ya existe, puede importarlo con otro nombre. Por ejemplo:

```
# zpool import dozer zeepool
```

Este comando importa el grupo exportado `dozer` con el nombre nuevo `zeepool`. El nuevo nombre de la agrupación persiste.

Si el grupo no se ha exportado correctamente, ZFS solicita que el indicador `-f` impida la importación accidental de un grupo que otro sistema todavía está usando. Por ejemplo:

```
# zpool import dozer
cannot import 'dozer': pool may be in use on another system
use '-f' to import anyway
# zpool import -f dozer
```

Nota – No intente importar una agrupación que esté activa en un sistema a otro. ZFS no es un clúster nativo, ni un sistema de archivos paralelo o distribuido y no puede proporcionar acceso simultáneo de varios hosts diferentes.

Las agrupaciones también se pueden importar en una raíz alternativa mediante la opción `-R`. Si desea más información sobre otras agrupaciones raíz, consulte "Uso de agrupaciones raíz de ZFS alternativas" en la página 294.

Importación de una agrupación a la que le falta un dispositivo de registro

De manera predeterminada, una agrupación a la que le falta un dispositivo de registro no se puede importar. Puede utilizar el comando `zpool import -m` para forzar la importación de una agrupación a la que le falta un dispositivo de registro. Por ejemplo:

```
# zpool import dozer
  pool: dozer
    id: 16216589278751424645
 state: UNAVAIL
status: One or more devices are missing from the system.
action: The pool cannot be imported. Attach the missing
        devices and try again.
   see: http://support.oracle.com/msg/ZFS-8000-6X
config:
```

```
        dozer             UNAVAIL  missing device
          mirror-0        ONLINE
            c8t0d0  ONLINE
            c8t1d0  ONLINE

device details:

        missing-1       UNAVAIL          corrupted data
        status: ZFS detected errors on this device.
                The device has bad label or disk contents.


        Additional devices are known to be part of this pool, though their
        exact configuration cannot be determined.
```

Importe la agrupación a la que le falta el dispositivo de registro. Por ejemplo:

```
# zpool import -m dozer
# zpool status dozer
  pool: dozer
 state: DEGRADED
status: One or more devices are unavailable in response to persistent errors.
        Sufficient replicas exist for the pool to continue functioning in a
        degraded state.
action: Determine if the device needs to be replaced, and clear the errors
        using 'zpool clear' or 'fmadm repaired', or replace the device
        with 'zpool replace'.
        Run 'zpool status -v' to see device specific details.
  scan: none requested
config:

        NAME                    STATE     READ WRITE CKSUM
        dozer                   DEGRADED     0     0     0
          mirror-0              ONLINE       0     0     0
            c8t0d0              ONLINE       0     0     0
            c8t1d0              ONLINE       0     0     0
        logs
          2189413556875979854  UNAVAIL      0     0     0

errors: No known data errors
```

Después de conectar el dispositivo de registro que faltaba, ejecute el comando `zpool clear` para eliminar los errores de agrupación.

Se puede intentar una recuperación similar con los dispositivos de registro reflejados faltantes. Por ejemplo:

Después de conectar los dispositivos de registro que faltaban, ejecute el comando `zpool clear` para eliminar los errores de agrupación.

Importación de una agrupación en modo de sólo lectura

Puede importar una agrupación en el modo de sólo lectura. Si una agrupación se daña de tal manera que no se puede acceder a ella, es posible que esta función le permita recuperar los datos de la agrupación. Por ejemplo:

```
# zpool import -o readonly=on tank
# zpool scrub tank
cannot scrub tank: pool is read-only
```

Cuando una agrupación se importa en modo de sólo lectura, se aplican las siguientes condiciones:

- Todos los volúmenes y sistemas de archivos se montan en modo de sólo lectura.
- El procesamiento de transacciones de agrupación está desactivado. Esto también significa que cualquier escritura síncrona pendiente en el intento de registro no se aplica hasta que la agrupación se haya importado con permiso de lectura y escritura.
- Los intentos de establecer una propiedad de agrupación durante la importación de sólo lectura se ignoran.

Para volver a establecer una agrupación de sólo lectura en modo de lectura y escritura, se debe exportar e importar la agrupación. Por ejemplo:

```
# zpool export tank
# zpool import tank
# zpool scrub tank
```

Importación de una agrupación mediante una ruta de dispositivo específico

El siguiente comando permite importar la agrupación `dpool` mediante la identificación de uno de los dispositivos específicos de la agrupación, `/dev/dsk/c2t3d0`, en este ejemplo.

```
# zpool import -d /dev/dsk/c2t3d0s0 dpool
# zpool status dpool
  pool: dpool
 state: ONLINE
 scan: resilvered 952K in 0h0m with 0 errors on Fri Jun 29 16:22:06 2012
config:

        NAME        STATE     READ WRITE CKSUM
        dpool       ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c2t3d0  ONLINE       0     0     0
            c2t1d0  ONLINE       0     0     0
```

Si bien esta agrupación está compuesta por discos enteros, el comando debe incluir el identificador de segmento del dispositivo específico.

Recuperación de agrupaciones de almacenamiento de ZFS destruidas

El comando `zpool import -D` es apto para recuperar una agrupación de almacenamiento que se haya destruido. Por ejemplo:

```
# zpool destroy tank
# zpool import -D
  pool: tank
    id: 5154272182900538157
 state: ONLINE (DESTROYED)
action: The pool can be imported using its name or numeric identifier.
config:

        tank        ONLINE
          mirror-0  ONLINE
            c1t0d0  ONLINE
            c1t1d0  ONLINE
```

En esta salida `zpool import`, puede identificar la agrupación `tank` como la destruida debido a la siguiente información de estado:

```
state: ONLINE (DESTROYED)
```

Para recuperar la agrupación destruida, ejecute de nuevo el comando `zpool import -D` con la agrupación que se debe recuperar. Por ejemplo:

```
# zpool import -D tank
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE
          mirror-0  ONLINE
            c1t0d0  ONLINE
            c1t1d0  ONLINE

errors: No known data errors
```

Si uno de los dispositivos de la agrupación destruida no está disponible, es posible que pueda recuperar la agrupación destruida al incluir la opción `-f`. En esta situación, debería importar la agrupación degradada y después intentar solucionar el error de dispositivo. Por ejemplo:

```
# zpool destroy dozer
# zpool import -D
  pool: dozer
    id: 4107023015970708695
 state: DEGRADED (DESTROYED)
status: One or more devices are unavailable.
```

```
action: The pool can be imported despite missing or damaged devices.  The
        fault tolerance of the pool may be compromised if imported.
config:

        dozer             DEGRADED
          raidz2-0        DEGRADED
            c8t0d0        ONLINE
            c8t1d0        ONLINE
            c8t2d0        ONLINE
            c8t3d0        UNAVAIL  cannot open
            c8t4d0        ONLINE

device details:

        c8t3d0    UNAVAIL          cannot open
        status: ZFS detected errors on this device.
                The device was missing.
# zpool import -Df dozer
# zpool status -x
  pool: dozer
 state: DEGRADED
status: One or more devices are unavailable in response to persistent errors.
        Sufficient replicas exist for the pool to continue functioning in a
        degraded state.
action: Determine if the device needs to be replaced, and clear the errors
        using 'zpool clear' or 'fmadm repaired', or replace the device
        with 'zpool replace'.
        Run 'zpool status -v' to see device specific details.
  scan: none requested
config:

        NAME                     STATE     READ WRITE CKSUM
        dozer                    DEGRADED     0     0     0
          raidz2-0               DEGRADED     0     0     0
            c8t0d0               ONLINE       0     0     0
            c8t1d0               ONLINE       0     0     0
            c8t2d0               ONLINE       0     0     0
            4881130428504041127  UNAVAIL      0     0     0
            c8t4d0        ONLINE       0     0     0

errors: No known data errors
# zpool online dozer c8t4d0
# zpool status -x
all pools are healthy
```

Actualización de agrupaciones de almacenamiento de ZFS

Si dispone de agrupaciones de almacenamiento ZFS de una versión anterior de Solaris, puede actualizarlas con el comando `zpool upgrade` para poder aprovechar las funciones de las agrupaciones en la versión actual. Además, el comando `zpool status` le informa cuando las agrupaciones están ejecutando versiones anteriores. Por ejemplo:

```
# zpool status
  pool: tank
 state: ONLINE
status: The pool is formatted using an older on-disk format.  The pool can
        still be used, but some features are unavailable.
action: Upgrade the pool using 'zpool upgrade'.  Once this is done, the
        pool will no longer be accessible on older software versions.
 scrub: none requested
config:
        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c1t0d0  ONLINE       0     0     0
            c1t1d0  ONLINE       0     0     0
errors: No known data errors
```

La sintaxis siguiente es válida para identificar información adicional sobre una versión concreta y compatible:

```
# zpool upgrade -v
This system is currently running ZFS pool version 33.

The following versions are supported:

VER  DESCRIPTION
---  --------------------------------------------------------
 1   Initial ZFS version
 2   Ditto blocks (replicated metadata)
 3   Hot spares and double parity RAID-Z
 4   zpool history
 5   Compression using the gzip algorithm
 6   bootfs pool property
 7   Separate intent log devices
 8   Delegated administration
 9   refquota and refreservation properties
 10  Cache devices
 11  Improved scrub performance
 12  Snapshot properties
 13  snapused property
 14  passthrough-x aclinherit
 15  user/group space accounting
 16  stmf property support
 17  Triple-parity RAID-Z
 18  Snapshot user holds
 19  Log device removal
 20  Compression using zle (zero-length encoding)
 21  Deduplication
 22  Received properties
 23  Slim ZIL
 24  System attributes
 25  Improved scrub stats
 26  Improved snapshot deletion performance
 27  Improved snapshot creation performance
 28  Multiple vdev replacements
 29  RAID-Z/mirror hybrid allocator
 30  Encryption
 31  Improved 'zfs list' performance
```

```
 32  One MB blocksize
 33  Improved share support
 34  Sharing with inheritance

For more information on a particular version, including supported releases,
see the ZFS Administration Guide.
```

A continuación puede ejecutar el comando `zpool upgrade` para actualizar todas las agrupaciones. Por ejemplo:

```
# zpool upgrade -a
```

Nota – Si moderniza la agrupación a una versión de ZFS posterior, no se podrá acceder a la agrupación en un sistema que ejecute una versión antigua de ZFS.

◆ ◆ ◆ CAPÍTULO 4

Gestión de componentes de la agrupación raíz ZFS

En este capítulo, se describe cómo gestionar los componentes de la agrupación raíz Oracle Solaris ZFS, como la conexión de un reflejo de la agrupación raíz, la clonación de un entorno de inicio ZFS, y la modificación del tamaño de los dispositivos de intercambio y volcado.

Este capítulo se divide en las secciones siguientes:

Para obtener información sobre la recuperación de agrupaciones raíz, consulte el Capítulo 11, "Archivado de instantáneas y recuperación de agrupaciones raíz".

Si desea obtener información sobre las novedades de último momento, consulte las notas de la versión Oracle Solaris 11.1.

Gestión de componentes de la agrupación raíz ZFS (descripción general)

ZFS es el sistema de archivos raíz predeterminado de la versión Oracle Solaris 11. Revise las siguientes consideraciones al instalar la versión de Oracle Solaris.

- **Instalación**: en la versión Oracle Solaris 11, puede efectuar la instalación y el inicio desde un sistema de archivos raíz ZFS de las siguientes maneras:
  - Live CD (sólo x86): instala una agrupación raíz ZFS en un solo disco. Puede utilizar el menú de partición `fdisk` durante la instalación a fin de particionar el disco para su entorno.

  - Instalación de texto (SPARC y x86): instala una agrupación raíz ZFS en un solo disco desde soportes o a través de la red. Puede utilizar el menú de partición `fdisk` durante la instalación a fin de particionar el disco para su entorno.
  - Automated Installer (AI) (SPARC y x86): instala una agrupación raíz ZFS de forma automática. Puede utilizar un manifiesto AI para determinar el disco y las particiones de disco que se utilizarán para la agrupación raíz ZFS.
- **Dispositivos de intercambio y volcado**: creados de manera automática en los volúmenes de ZFS en la agrupación raíz de ZFS mediante todos los métodos de instalación mencionados anteriormente. Para obtener más información sobre la gestión de dispositivos de intercambio y volcado ZFS, consulte "Gestión de los dispositivos de intercambio y volcado ZFS" en la página 135.
- **Configuración de agrupación raíz reflejada**: puede configurar una agrupación raíz reflejada durante una instalación automática. Para obtener más información sobre la configuración de una agrupación raíz reflejada después de la instalación, consulte "Cómo configurar una agrupación raíz reflejada (SPARC o x86/VTOC)" en la página 125.
- **Gestión del espacio de la agrupación raíz**: una vez instalado el sistema, considere la definición de una cuota en el sistema de archivos raíz ZFS para evitar que el sistema de archivos raíz se llene. Actualmente, no hay ningún espacio de la agrupación raíz ZFS reservado como protección para un sistema de archivos completo. Por ejemplo, si tiene un disco de 68 GB para la agrupación raíz, considere definir una cuota de 67 GB en el sistema de archivos raíz ZFS (`rpool/ROOT/solaris`), lo cual deja 1 GB de espacio restante en el sistema de archivos. Para obtener información sobre la configuración de cuotas, consulte "Establecimiento de cuotas en sistemas de archivos ZFS" en la página 200.

Requisitos de la agrupación raíz ZFS

Revise las siguientes secciones que describen los requisitos de espacio y configuración de la agrupación raíz ZFS.

Requisitos de espacio de la agrupación raíz ZFS

Cuando se instala un sistema, el tamaño del volumen de intercambio y el volumen de volcado dependen de la cantidad de memoria física. La cantidad mínima de espacio de agrupación disponible para un sistema de archivos raíz de ZFS iniciable depende de la cantidad de memoria física, el espacio en disco disponible y la cantidad de entornos de inicio que se vayan a crear.

Revise los siguientes requisitos de espacio de la agrupación de almacenamiento de ZFS:

- Para obtener una descripción de los requisitos de memoria para los distintos métodos de instalación, consulte *Notas de la versión de Oracle Solaris 11.1* .
- Se recomienda un mínimo de 7 a 13 GB de espacio en el disco. El espacio se consume del modo siguiente:

- **Dispositivo de volcado y área de intercambio**: los tamaños predeterminados de los volúmenes de intercambio y volcado creados por los programas de instalación de Solaris varían según la cantidad de memoria en el sistema y otras variables. El tamaño del dispositivo de volcado es de aproximadamente la mitad del tamaño de la memoria física o más, según la actividad del sistema.

  Puede ajustar los tamaños de los volúmenes de intercambio y de volcado según sea necesario, siempre y cuando los nuevos tamaños permitan el funcionamiento del sistema, durante la instalación o después de ella. Para obtener más información, consulte "Ajuste del tamaño de los dispositivos de intercambio y volcado ZFS" en la página 136.
- **Entorno de inicio**: un entorno de inicio ZFS tiene aproximadamente entre 4 y 6 GB. Cada entorno de inicio de ZFS que se clona de otro entorno de inicio de ZFS no necesita espacio en disco adicional. Tenga en cuenta que el tamaño del entorno de inicio aumentará cuando se actualice dicho entorno, en función de las actualizaciones. Todos los entornos de inicio ZFS de la misma agrupación raíz utilizan los mismos dispositivos de intercambio y volcado.
- **Componentes del sistema operativo Solaris**: todos los subdirectorios del sistema de archivos raíz que forman parte de la imagen del sistema operativo, con la excepción de `/var`, deben estar en el sistema de archivos raíz. Además, todos los componentes del sistema operativo Solaris deben residir en la agrupación raíz, con la excepción de los dispositivos de intercambio y volcado.

Requisitos de configuración de la agrupación raíz ZFS

Revise los siguientes requisitos de configuración de la agrupación de almacenamiento ZFS:

- En Oracle Solaris 11.1, el disco destinado a la agrupación raíz puede tener una etiqueta EFI (GPT) o SMI (VTOC) en un sistema basado en x86 o una etiqueta SMI (VOTC) en un sistema SPARC.
  - Los sistemas SPARC con firmware compatible con GPT actualizado instalará una etiqueta de disco EFI (GPT) en el disco de agrupación raíz o en los discos. Si el sistema SPARC no tiene el firmware actualizado, se instala una etiqueta del disco SMI (VTOC) en el disco de agrupación raíz o en los discos.
  - Un sistema basado en x86 se instala con una etiqueta EFI (GPT) en uno o varios discos de agrupación raíz, en la mayoría de los casos.

  Para obtener información sobre el aspecto de la etiqueta EFI (GPT) en un sistema basado en x86, consulte "Utilización de discos en un grupo de almacenamiento de ZFS" en la página 48.
- La agrupación debe existir en un segmento de disco o en segmentos de disco que se reflejan si hay una etiqueta SMI (VTOC) en un sistema basado en SPARC o x86. Si los discos de agrupación raíz tienen una etiqueta EFI (GPT), la agrupación puede existir en un disco entero o en discos enteros reflejados. Si intenta utilizar una configuración de agrupación no admitida durante una operación `beadm`, aparecerá un mensaje similar al siguiente:

```
ERROR: ZFS pool name does not support boot environments
```

Para obtener una descripción detallada de las configuraciones admitidas para la agrupación ZFS, consulte "Creación de una agrupación raíz ZFS" en la página 57.

- En un sistema basado en x86, el disco debe contener una partición `fdisk` de Solaris. Se crea una partición `fdisk` de Solaris automáticamente cuando se instala el sistema basado en x86. Para obtener más información acerca de las particiones `fdisk` de Solaris, consulte "directrices para la creación de una partición fdisk" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.
- Las propiedades de la agrupación o del sistema de archivos se pueden definir en una agrupación raíz durante una instalación automática. El algoritmo de compresión `gzip` no se admite en las agrupaciones raíz.
- No cambie el nombre de la agrupación raíz tras su creación en una instalación inicial. El cambio de nombre de la agrupación raíz puede impedir el inicio del sistema.

Gestión de la agrupación raíz ZFS

Las siguientes secciones proporcionan información sobre cómo instalar y actualizar una agrupación raíz ZFS, y cómo configurar una agrupación raíz reflejada.

Instalación de una agrupación raíz ZFS

El método de instalación Live CD de Oracle Solaris 11 instala una agrupación raíz ZFS predeterminada en un solo disco. Con el método de instalación automática (AI), puede crear un manifiesto AI a fin de identificar los discos o discos reflejados que se utilizan para la agrupación raíz de ZFS.

El instalador automatizado ofrece la flexibilidad de instalar una agrupación raíz ZFS en el disco de inicio predeterminado o en un disco de destino que haya identificado. Puede especificar el dispositivo lógico, como `c1t0d0`, o la ruta del dispositivo físico. Además, puede utilizar el identificador MPxIO o el ID del dispositivo que se instalará.

Tras la instalación, revise la agrupación de almacenamiento de ZFS y la información del sistema de archivos, que pueden variar según el tipo de instalación y las personalizaciones. Por ejemplo:

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        rpool       ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
```

```
            c8t0d0  ONLINE       0     0     0
            c8t1d0  ONLINE       0     0     0
# zfs list
NAME                      USED  AVAIL  REFER  MOUNTPOINT
rpool                    11.8G  55.1G  4.58M  /rpool
rpool/ROOT               3.57G  55.1G    31K  legacy
rpool/ROOT/solaris       3.57G  55.1G  3.40G  /
rpool/ROOT/solaris/var    165M  55.1G   163M  /var
rpool/VARSHARE           42.5K  55.1G  42.5K  /var/share
rpool/dump               6.19G  55.3G  6.00G  -
rpool/export               63K  55.1G    32K  /export
rpool/export/home          31K  55.1G    31K  /export/home
rpool/swap               2.06G  55.2G  2.00G  -
```

Revise la información del entorno de inicio ZFS. Por ejemplo:

```
# beadm list
BE      Active Mountpoint Space Policy Created
--      ------ ---------- ----- ------ -------
solaris NR     /          3.75G static 2012-07-20 12:10
```

En la salida anterior, el campo `Active` indica que el entorno de inicio está activo ahora y está representado por `N`; que está activo en el inicio y está representado por `R`; o ambos casos, representado por `NR`.

▼ Cómo actualizar el entorno de inicio ZFS

El entorno de inicio ZFS predeterminado se denomina `solaris` por defecto. Puede identificar los entornos de inicio mediante el comando `beadm list`. Por ejemplo:

```
# beadm list
BE      Active Mountpoint Space Policy Created
--      ------ ---------- ----- ------ -------
solaris NR     /          3.82G static 2012-07-19 13:44
```

En la salida anterior, NR significa que el entorno de inicio está activo ahora y será el entorno de inicio activo al reiniciar.

Puede utilizar el comando `pkg image` para actualizar el entorno de inicio de ZFS. Si actualiza el entorno de inicio ZFS mediante el comando `pkg update`, se crea y se activa automáticamente un nuevo entorno de inicio, a menos que las actualizaciones al entorno de inicio existente sean mínimas.

1 Actualice el entorno de inicio ZFS.

```
# pkg update


DOWNLOAD                                  PKGS       FILES    XFER (MB)
Completed                                707/707 10529/10529  194.9/194.9
.
.
.
```

Se crea y se activa automáticamente un nuevo entorno de inicio, `solaris-1`.

También puede crear y activar un EI de resguardo fuera del proceso de actualización.

```
# beadm create solaris-1
# beadm activate solaris-1
```

2 Reinicie el sistema para completar la activación del entorno de inicio. A continuación, confirme el estado del entorno de inicio.

```
# init 6
.
.
.
# beadm list
BE        Active Mountpoint Space  Policy Created
--        ------ ---------- -----  ------ -------
solaris   -      -          46.95M static 2012-07-20 10:25
solaris-1 NR     /          3.82G  static 2012-07-19 14:45
```

3 Si se produce un error cuando se inicia el nuevo entorno de inicio, active el entorno de inicio anterior y vuelva a iniciarlo.

```
# beadm activate solaris
# init 6
```

▼ Cómo montar un entorno de inicio alternativo

Es posible que necesite copiar un archivo o acceder a él desde otro entorno de inicio para fines de recuperación.

1 Conviértase en un administrador.

2 Monte el entorno de inicio alternativo.

```
# beadm mount solaris-1 /mnt
```

3 Acceda al entorno de inicio.

```
# ls /mnt
bin         export      media       pkg         rpool       tmp
boot        home        mine        platform    sbin        usr
dev         import      mnt         proc        scde        var
devices     java        net         project     shared
doe         kernel      nfs4        re          src
etc         lib         opt         root        system
```

4 Desmonte el entorno de inicio alternativo cuando haya terminado de trabajar con él.

```
# beadm umount solaris-1
```

▼ Cómo configurar una agrupación raíz reflejada (SPARC o x86/VTOC)

Si no configura una agrupación raíz reflejada durante una instalación automática, puede configurar una agrupación raíz reflejada de manera sencilla después de la instalación.

Para obtener información sobre el reemplazo de un disco en una agrupación raíz, consulte "Cómo reemplazar un disco en una agrupación raíz ZFS (SPARC o x86/VTOC)" en la página 128.

1 Muestre el estado actual de la agrupación raíz.

```
# zpool status rpool
  pool: rpool
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        rpool       ONLINE       0     0     0
          c2t0d0s0  ONLINE       0     0     0

errors: No known data errors
```

2 Si es necesario, prepare un segundo disco para anexar a la agrupación raíz.

- SPARC: confirme que el disco tiene una etiqueta del disco SMI (VTOC) y un segmento 0. Si necesita volver a etiquetar el disco y crear un segmento 0, consulte "cómo crear un segmento de disco para un sistema de archivos raíz ZFS" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.
- x86: confirme que el disco tiene una partición `fdisk`, una etiqueta SMI y un segmento 0. Si necesita volver a particionar el disco y crear un segmento 0, consulte "preparación de un disco para un sistema de archivos raíz ZFS" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.

3 Conecte un segundo disco para configurar una agrupación raíz reflejada.

```
# zpool attach rpool c2t0d0s0 c2t1d0s0
Make sure to wait until resilver is done before rebooting.
```

El etiquetado correcto de discos y los bloques de inicio se aplican automáticamente.

4 Vea el estado de la agrupación raíz para confirmar que se ha completado la reconstrucción.

```
# zpool status rpool
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
```

```
  scan: resilver in progress since Fri Jul 20 13:39:53 2012
    938M scanned out of 11.7G at 46.9M/s, 0h3m to go
    938M resilvered, 7.86% done
config:

        NAME          STATE     READ WRITE CKSUM
        rpool         DEGRADED     0     0     0
          mirror-0    DEGRADED     0     0     0
            c2t0d0s0  ONLINE       0     0     0
            c2t1d0s0  DEGRADED     0     0     0  (resilvering)
```

En la salida anterior, el proceso de reconstrucción no se ha completado. La reconstrucción se completa cuando aparecen mensajes similares al siguiente:

```
resilvered 11.6G in 0h5m with 0 errors on Fri Jul 20 13:57:25 2012
```

5 Si va conectar un disco más grande, establezca la propiedad `autoexpand` para ampliar el tamaño de la agrupación.

Determine el tamaño de la agrupación `rpool` existente:

```
# zpool list rpool
NAME   SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
rpool  29.8G   152K  29.7G   0%  1.00x  ONLINE  -

# zpool set autoexpand=on rpool
```

Revise el tamaño de la agrupación `rpool` expandida:

```
# zpool list rpool
NAME   SIZE  ALLOC  FREE  CAP  DEDUP  HEALTH  ALTROOT
rpool  279G   146K  279G   0%  1.00x  ONLINE  -
```

6 Verifique que puede iniciar el sistema correctamente desde el disco nuevo.

▼ Cómo configurar una agrupación raíz reflejada (x86/EFI [GPT])

Oracle Solaris 11.1 instala una etiqueta EFI (GPT) de manera predeterminada en un sistema basado en x86, en la mayoría de los casos.

Si no configura una agrupación raíz reflejada durante una instalación automática, puede configurar una agrupación raíz reflejada de manera sencilla después de la instalación.

Para obtener información sobre el reemplazo de un disco en una agrupación raíz, consulte "Cómo reemplazar un disco en una agrupación raíz ZFS (SPARC o x86/VTOC)" en la página 128.

1 Muestre el estado actual de la agrupación raíz.

```
# zpool status rpool
 pool:  rpool
 state: ONLINE
  scan: none requested
config:

        NAME      STATE     READ WRITE CKSUM
        rpool     ONLINE       0     0     0
          c2t0d0  ONLINE       0     0     0

errors: No known data errors
```

2 Conecte un segundo disco para configurar una agrupación raíz reflejada.

```
# zpool attach rpool c2t0d0 c2t1d0
Make sure to wait until resilver is done before rebooting.
```

El etiquetado correcto de discos y los bloques de inicio se aplican automáticamente.

Si personalizó particiones en el disco de agrupación raíz, posiblemente necesita una sintaxis similar a la siguiente:

```
# zpool attach rpool c2t0d0s0 c2t1d0
```

3 Vea el estado de la agrupación raíz para confirmar que se ha completado la reconstrucción.

```
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Fri Jul 20 13:52:05 2012
    809M scanned out of 11.6G at 44.9M/s, 0h4m to go
    776M resilvered, 6.82% done
config:

        NAME        STATE     READ WRITE CKSUM
        rpool       DEGRADED     0     0     0
          mirror-0  DEGRADED     0     0     0
            c8t0d0  ONLINE       0     0     0
            c8t1d0  DEGRADED     0     0     0  (resilvering)

errors: No known data errors
```

En la salida anterior, el proceso de reconstrucción no se ha completado. La reconstrucción se completa cuando aparecen mensajes similares al siguiente:

```
resilvered 11.6G in 0h5m with 0 errors on Fri Jul 20 13:57:25 2012
```

4 **Si va conectar un disco más grande, establezca la propiedad `autoexpand` para ampliar el tamaño de la agrupación.**

Determine el tamaño de la agrupación `rpool` existente:

```
# zpool list rpool
NAME   SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
rpool  29.8G   152K  29.7G   0%  1.00x  ONLINE  -

# zpool set autoexpand=on rpool
```

Revise el tamaño de la agrupación `rpool` expandida:

```
# zpool list rpool
NAME   SIZE  ALLOC  FREE  CAP  DEDUP  HEALTH  ALTROOT
rpool  279G   146K  279G   0%  1.00x  ONLINE  -
```

5 **Verifique que puede iniciar el sistema correctamente desde el disco nuevo.**

▼ Cómo reemplazar un disco en una agrupación raíz ZFS (SPARC o x86/VTOC)

Es posible que necesite sustituir un disco en la agrupación raíz, por los siguientes motivos:

- La agrupación raíz es demasiado pequeña y desea sustituirla por un disco mayor.
- El disco de la agrupación raíz no funciona correctamente. En una agrupación no redundante, si el disco falla y el sistema no se inicia, deberá realizar el inicio desde un medio alternativo, como un CD o la red, antes de sustituir el disco de la agrupación raíz.
- Si utiliza el comando `zpool replace` para reemplazar un disco en una agrupación raíz, debe aplicar manualmente los bloques de inicio.

En una configuración de agrupación raíz reflejada, es posible que pueda sustituir un disco sin tener que iniciar el sistema desde un medio alternativo. Puede sustituir un disco dañado mediante el comando `zpool replace` o, si tiene un disco adicional, puede utilizar el comando `zpool attach`. Consulte los siguientes pasos para obtener un ejemplo de cómo conectar un disco adicional y cómo desconectar un disco de la agrupación raíz.

Los sistemas con discos SATA requieren que se desconecte el disco y se anule su configuración antes de intentar la operación `zpool replace` para sustituir un disco dañado. Por ejemplo:

```
# zpool offline rpool c1t0d0s0
# cfgadm -c unconfigure c1::dsk/c1t0d0
<Physically remove failed disk c1t0d0>
<Physically insert replacement disk c1t0d0>
# cfgadm -c configure c1::dsk/c1t0d0
<Confirm that the new disk has an SMI label and a slice 0>
# zpool online rpool c1t0d0s0
# zpool replace rpool c1t0d0s0
# zpool status rpool
```

```
<Let disk resilver before installing the boot blocks>
# bootadm install-bootloader
```

En algunos dispositivos de hardware, no es necesario conectar ni volver a configurar el disco de sustitución después de insertarlo.

1 Conecte físicamente el disco de sustitución.

2 Si es necesario, prepare un segundo disco para anexar a la agrupación raíz.

- SPARC: confirme que el disco de reemplazo (nuevo) tenga una etiqueta SMI (VTOC) y un segmento 0. Para obtener información sobre el reetiquetado de un disco que está diseñado para la agrupación raíz, consulte "Cómo etiquetar un disco" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.
- x86: confirme que el disco tiene una partición `fdisk`, una etiqueta SMI y un segmento 0. Si necesita volver a particionar el disco y crear un segmento 0, consulte "cómo configurar un disco para un sistema de archivos raíz ZFS" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.

3 Conecte el nuevo disco a la agrupación raíz.

Por ejemplo:

```
# zpool attach rpool c2t0d0s0 c2t1d0s0
Make sure to wait until resilver is done before rebooting.
```

El etiquetado correcto de discos y los bloques de inicio se aplican automáticamente.

4 Confirme el estado de la agrupación raíz.

Por ejemplo:

```
# zpool status rpool
  pool: rpool
 state: ONLINE
 scan: resilvered 11.7G in 0h5m with 0 errors on Fri Jul 20 13:45:37 2012
config:

        NAME          STATE     READ WRITE CKSUM
        rpool         ONLINE       0     0     0
          mirror-0    ONLINE       0     0     0
            c2t0d0s0  ONLINE       0     0     0
            c2t1d0s0  ONLINE       0     0     0

errors: No known data errors
```

5 Verifique que puede iniciar el sistema desde el nuevo disco después de que se ha completado la reconstrucción.

Por ejemplo, en un sistema basado en SPARC:

```
ok boot /pci@1f,700000/scsi@2/disk@1,0
```

Identifique los nombres de ruta del dispositivo de inicio de los discos nuevo y actual para poder probar el inicio desde el disco de sustitución e iniciar además el sistema manualmente desde el disco existente, si fuera necesario, si el disco de sustitución falla. En el ejemplo siguiente, el disco de la agrupación raíz actual (`c2t0d0s0`) es:

```
/pci@1f,700000/scsi@2/disk@0,0
```

En el ejemplo siguiente, el disco de inicio de sustitución (`c2t1d0s0`) es:

```
boot /pci@1f,700000/scsi@2/disk@1,0
```

6 Si el sistema se inicia desde el nuevo disco, desconecte el disco antiguo.

Por ejemplo:

```
# zpool detach rpool c2t0d0s0
```

7 Si va conectar un disco más grande, establezca la propiedad `autoexpand` para ampliar el tamaño de la agrupación.

```
# zpool set autoexpand=on rpool
```

O bien, amplíe el dispositivo:

```
# zpool online -e c2t1d0s0
```

8 Configure el sistema para que se inicie automáticamente desde el disco nuevo.

- SPARC: configure el sistema para que se inicie automáticamente desde el disco nuevo, mediante el comando `eeprom` o el comando `setenv` desde la PROM de inicio.
- x86: vuelva a configurar el BIOS del sistema.

▼ Cómo reemplazar un disco en una agrupación raíz ZFS (SPARC o x86/EFI [GPT])

Oracle Solaris 11.1 instala una etiqueta EFI (GPT) de manera predeterminada en un sistema basado en x86, en la mayoría de los casos.

Es posible que necesite sustituir un disco en la agrupación raíz, por los siguientes motivos:

- La agrupación raíz es demasiado pequeña y desea sustituirla por un disco mayor.
- El disco de la agrupación raíz no funciona correctamente. En una agrupación no redundante, si el disco falla y el sistema no se inicia, deberá realizar el inicio desde un medio alternativo, como un CD o la red, antes de sustituir el disco de la agrupación raíz.
- Si utiliza el comando `zpool replace` para reemplazar un disco en una agrupación raíz, debe aplicar manualmente los bloques de inicio.

En una configuración de agrupación raíz reflejada, es posible que pueda sustituir un disco sin tener que iniciar el sistema desde un medio alternativo. Puede sustituir un disco dañado mediante el comando `zpool replace` o, si tiene un disco adicional, puede utilizar el comando `zpool attach`. Consulte los siguientes pasos para obtener un ejemplo de cómo conectar un disco adicional y cómo desconectar un disco de la agrupación raíz.

Los sistemas con discos SATA requieren que se desconecte el disco y se anule su configuración antes de intentar la operación `zpool replace` para sustituir un disco dañado. Por ejemplo:

```
# zpool offline rpool c1t0d0
# cfgadm -c unconfigure c1::dsk/c1t0d0
<Physically remove failed disk c1t0d0>
<Physically insert replacement disk c1t0d0>
# cfgadm -c configure c1::dsk/c1t0d0
# zpool online rpool c1t0d0
# zpool replace rpool c1t0d0
# zpool status rpool
<Let disk resilver before installing the boot blocks>
x86# bootadm install-bootloader
```

En algunos dispositivos de hardware, no es necesario conectar ni volver a configurar el disco de sustitución después de insertarlo.

1 Conecte físicamente el disco de sustitución.

2 Conecte el nuevo disco a la agrupación raíz.

Por ejemplo:

```
# zpool attach rpool c2t0d0 c2t1d0
Make sure to wait until resilver is done before rebooting.
```

El etiquetado correcto de discos y los bloques de inicio se aplican automáticamente.

3 Confirme el estado de la agrupación raíz.

Por ejemplo:

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: resilvered 11.6G in 0h5m with 0 errors on Fri Jul 20 12:06:07 2012
config:

        NAME        STATE     READ WRITE CKSUM
        rpool       ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c2t0d0  ONLINE       0     0     0
            c2t1d0  ONLINE       0     0     0

errors: No known data errors
```

4 Verifique que puede iniciar el sistema desde el nuevo disco después de que se ha completado la reconstrucción.

5 **Si el sistema se inicia desde el nuevo disco, desconecte el disco antiguo.**

Por ejemplo:

```
# zpool detach rpool c2t0d0
```

6 **Si va conectar un disco más grande, establezca la propiedad `autoexpand` para ampliar el tamaño de la agrupación.**

```
# zpool set autoexpand=on rpool
```

O bien, amplíe el dispositivo:

```
# zpool online -e c2t1d0
```

7 **Configure el sistema para que se inicie automáticamente desde el disco nuevo.**

Reconfigure el BIOS del sistema.

▼ Cómo crear un entorno de inicio en otra agrupación raíz (SPARC o x86/VTOC)

Si desea volver a crear el entorno de inicio en otra agrupación raíz, siga los siguientes pasos. Puede modificar los pasos en función de si desea dos agrupaciones raíz con entornos de inicio similares que tienen dispositivos de intercambio y volcado independientes, o si sólo desea un entorno de inicio en otra agrupación raíz que comparte los dispositivos de intercambio y volcado.

Después de activar la segunda agrupación raíz y realizar el inicio desde el nuevo entorno de inicio de dicha agrupación, esta no tendrá información sobre el entorno de inicio anterior en la primera agrupación raíz. Si desea volver a iniciar el entorno de inicio original, deberá iniciar el sistema manualmente desde el disco de inicio de la agrupación raíz original.

1 **Cree una segunda agrupación raíz con un disco con etiqueta SMI (VTOC). Por ejemplo:**

```
# zpool create rpool2 c4t2d0s0
```

2 **Cree el nuevo entorno de inicio en la segunda agrupación raíz. Por ejemplo:**

```
# beadm create -p rpool2 solaris2
```

3 **Defina la propiedad `bootfs` en la segunda agrupación raíz. Por ejemplo:**

```
# zpool set bootfs=rpool2/ROOT/solaris2 rpool2
```

4 **Active el nuevo entorno de inicio. Por ejemplo:**

```
# beadm activate solaris2
```

5. **Inicie desde el nuevo entorno de inicio, pero debe iniciar, específicamente, desde el dispositivo de inicio de la segunda agrupación raíz.**

   ```
   ok boot disk2
   ```

   El sistema se debe ejecutar con el nuevo entorno de inicio.

6. **Vuelva a crear el volumen de intercambio. Por ejemplo:**

   ```
   # zfs create -V 4g rpool2/swap
   ```

7. **Actualice la entrada `/etc/vfstab` para el nuevo dispositivo de intercambio. Por ejemplo:**

   ```
   /dev/zvol/dsk/rpool2/swap       -               -               swap -     no      -
   ```

8. **Vuelva a crear el volumen de volcado. Por ejemplo:**

   ```
   # zfs create -V 4g rpool2/dump
   ```

9. **Restablezca el dispositivo de volcado. Por ejemplo:**

   ```
   # dumpadm -d /dev/zvol/dsk/rpool2/dump
   ```

10. **Restablezca el dispositivo de inicio predeterminado para iniciar el sistema desde el disco de inicio de la segunda agrupación raíz.**

    - SPARC: configure el sistema para que se inicie automáticamente desde el disco nuevo, mediante el comando `eeprom` o el comando `setenv` desde la PROM de inicio.
    - x86: vuelva a configurar el BIOS del sistema.

11. **Reinicie el sistema para borrar los dispositivos de intercambio y volcado de la agrupación raíz original.**

    ```
    # init 6
    ```

▼ Cómo crear un entorno de inicio en otra agrupación raíz (SPARC o x86/EFI [GPT])

Oracle Solaris 11.1 instala una etiqueta EFI (GPT) de manera predeterminada en un sistema basado en x86, en la mayoría de los casos.

Si desea volver a crear el entorno de inicio en otra agrupación raíz, siga los siguientes pasos. Puede modificar los pasos en función de si desea dos agrupaciones raíz con entornos de inicio similares que tienen dispositivos de intercambio y volcado independientes, o si sólo desea un entorno de inicio en otra agrupación raíz que comparte los dispositivos de intercambio y volcado.

Después de activar la segunda agrupación raíz y realizar el inicio desde el nuevo entorno de inicio de dicha agrupación, esta no tendrá información sobre el entorno de inicio anterior en la

primera agrupación raíz. Si desea volver a iniciar el entorno de inicio original, deberá iniciar el sistema manualmente desde el disco de inicio de la agrupación raíz original.

1 **Cree la agrupación raíz alternativa.**

```
# zpool create -B rpool2 c2t2d0
```

También puede crear una agrupación raíz alternativa reflejada. Por ejemplo:

```
# zpool create -B rpool2 mirror c2t2d0 c2t3d0
```

2 **Cree el nuevo entorno de inicio en la segunda agrupación raíz. Por ejemplo:**

```
# beadm create -p rpool2 solaris2
```

3 **Aplique la información de inicio a la segunda agrupación raíz. Por ejemplo:**

```
# bootadm install-bootloader -P rpool2
```

4 **Defina la propiedad `bootfs` en la segunda agrupación raíz. Por ejemplo:**

```
# zpool set bootfs=rpool2/ROOT/solaris2 rpool2
```

5 **Active el nuevo entorno de inicio. Por ejemplo:**

```
# beadm activate solaris2
```

6 **Inicie desde el nuevo entorno de inicio.**

- SPARC: configure el sistema para que se inicie automáticamente desde el disco nuevo, mediante el comando `eeprom` o el comando `setenv` desde la PROM de inicio.
- x86: vuelva a configurar el BIOS del sistema.

El sistema se debe ejecutar con el nuevo entorno de inicio.

7 **Vuelva a crear el volumen de intercambio. Por ejemplo:**

```
# zfs create -V 4g rpool2/swap
```

8 **Actualice la entrada `/etc/vfstab` para el nuevo dispositivo de intercambio. Por ejemplo:**

```
/dev/zvol/dsk/rpool2/swap      -               -               swap -     no      -
```

9 **Vuelva a crear el volumen de volcado. Por ejemplo:**

```
# zfs create -V 4g rpool2/dump
```

10 **Restablezca el dispositivo de volcado. Por ejemplo:**

```
# dumpadm -d /dev/zvol/dsk/rpool2/dump
```

11 **Reinicie el sistema para borrar los dispositivos de intercambio y volcado de la agrupación raíz original.**

```
# init 6
```

Gestión de los dispositivos de intercambio y volcado ZFS

Durante el proceso de instalación, se crea un área de intercambio en un volumen ZFS de la agrupación raíz ZFS. Por ejemplo:

```
# swap -l
swapfile                  dev    swaplo   blocks     free
/dev/zvol/dsk/rpool/swap 145,2        16 16646128 16646128
```

Durante el proceso de instalación, se crea un dispositivo de volcado en un volumen ZFS de la agrupación raíz ZFS. En general, un dispositivo de volcado no requiere administración porque se configura automáticamente en el momento de la instalación. Por ejemplo:

```
# dumpadm
      Dump content: kernel pages
       Dump device: /dev/zvol/dsk/rpool/dump (dedicated)
Savecore directory: /var/crash/
  Savecore enabled: yes
   Save compressed: on
```

Si desactiva y elimina el dispositivo de volcado, deberá activarlo con el comando `dumpadm` una vez que se haya creado nuevamente. En la mayoría de los casos, sólo deberá ajustar el tamaño del dispositivo de volcado mediante el comando `zfs`.

Para obtener información sobre el tamaño de los volúmenes de intercambio y volcado creados por los programas de instalación, consulte "Requisitos de la agrupación raíz ZFS" en la página 120.

Tanto el tamaño del volumen de intercambio como el tamaño del volumen de volcado se pueden ajustar después de la instalación. Para obtener más información, consulte "Ajuste del tamaño de los dispositivos de intercambio y volcado ZFS" en la página 136.

Al trabajar con dispositivos de intercambio y volcado de ZFS, debe tener en cuenta los problemas siguientes:

- Para el área de intercambio y los dispositivos de volcado deben utilizarse volúmenes ZFS distintos.
- En la actualidad, no es posible utilizar un archivo de intercambio en un sistema de archivos ZFS.
- Si necesita cambiar el área de intercambio o el dispositivo de volcado después de instalar el sistema, utilice los comandos `swap` y `dumpadm` como en las versiones anteriores de Solaris. Para obtener más información, consulte el Capítulo 16, "Configuración de espacio de intercambio adicional (tareas)" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos* y el Capítulo 1, "Gestión de información sobre la caída del sistema (tareas)" de *Resolución de problemas típicos en Oracle Solaris 11.1*.

Ajuste del tamaño de los dispositivos de intercambio y volcado ZFS

Es posible que deba ajustar el tamaño de los dispositivos de intercambio y volcado después de la instalación o, posiblemente, volver a crear los volúmenes de intercambio y volcado.

- Puede volver a configurar la propiedad `volsize` del dispositivo de volcado tras haber instalado un sistema. Por ejemplo:

```
# zfs set volsize=2G rpool/dump
# zfs get volsize rpool/dump
NAME        PROPERTY  VALUE       SOURCE
rpool/dump  volsize   2G          -
```

- Puede cambiar el tamaño del volumen de intercambio, pero debe reiniciar el sistema para ver el aumento del tamaño de intercambio. Por ejemplo:

```
# swap -d /dev/zvol/dsk/rpool/swap
# zfs set volsize=2G rpool/swap
# swap -a /dev/zvol/dsk/rpool/swap
# init 6
```

  Para obtener información sobre cómo eliminar un dispositivo de intercambio en un sistema activo, consulte "Cómo agregar espacio de intercambio en un entorno raíz ZFS de Oracle Solaris" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.

- Si necesita más espacio de intercambio en un sistema que ya está instalado y el dispositivo de intercambio está ocupado, simplemente agregue otro volumen de intercambio. Por ejemplo:

```
# zfs create -V 2G rpool/swap2
```

- Active el nuevo volumen de intercambio. Por ejemplo:

```
# swap -a /dev/zvol/dsk/rpool/swap2
# swap -l
swapfile                  dev  swaplo   blocks   free
/dev/zvol/dsk/rpool/swap  256,1      16 1058800 1058800
/dev/zvol/dsk/rpool/swap2 256,3      16 4194288 4194288
```

- Agregue una entrada para el segundo volumen de intercambio en el archivo `/etc/vfstab`. Por ejemplo:

```
/dev/zvol/dsk/rpool/swap2     -            -            swap    -    no    -
```

Resolución de problemas de dispositivos de volcado ZFS

Revise los siguientes elementos si tiene problemas al capturar un volcado de bloqueo del sistema o al cambiar el tamaño del dispositivo de volcado.

- Si no se creó automáticamente un volcado de bloqueo, puede utilizar el comando `savecore` para guardar el volcado de bloqueo.

- Se crea automáticamente un dispositivo de volcado cuando se instala por primera vez un sistema de archivos raíz ZFS o se migra a un sistema de archivos raíz ZFS. En la mayoría de los casos, sólo será necesario ajustar el tamaño del dispositivo de volcado si su tamaño predeterminado es demasiado pequeño. Por ejemplo, en un sistema con mucha memoria, el tamaño del dispositivo de volcado se incrementa a 40 GB de la siguiente manera:

  ```
  # zfs set volsize=40G rpool/dump
  ```

  El cambio de tamaño de un dispositivo de volcado grande puede ser un proceso largo.

  Si, por cualquier motivo, debe activar un dispositivo de volcado tras crear un dispositivo de volcado manualmente, utilice una sintaxis similar a la siguiente:

  ```
  # dumpadm -d /dev/zvol/dsk/rpool/dump
        Dump content: kernel pages
         Dump device: /dev/zvol/dsk/rpool/dump (dedicated)
  Savecore directory: /var/crash/
    Savecore enabled: yes
     Save compressed: on
  ```

- Un sistema con 128 GB de memoria o más necesitará un dispositivo de volcado más grande que el dispositivo de volcado que se crea de forma predeterminada. Si el dispositivo de volcado es demasiado pequeño para capturar un volcado de bloqueo existente, se muestra un mensaje similar al siguiente:

  ```
  # dumpadm -d /dev/zvol/dsk/rpool/dump
  dumpadm: dump device /dev/zvol/dsk/rpool/dump is too small to hold a system dump
  dump size 36255432704 bytes, device size 34359738368 bytes
  ```

  Para obtener información sobre el cambio de tamaño de los dispositivos de intercambio y volcado, consulte "Planificación para espacio de intercambio" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.

- No se puede agregar actualmente un dispositivo de volcado a una agrupación con varios dispositivos de nivel superior. Verá un mensaje similar al siguiente:

  ```
  # dumpadm -d /dev/zvol/dsk/datapool/dump
  dump is not supported on device '/dev/zvol/dsk/datapool/dump':
  'datapool' has multiple top level vdevs
  ```

  Agregue el dispositivo de volcado a la agrupación raíz, que no puede tener varios dispositivos de nivel superior.

Inicio desde un sistema de archivos raíz ZFS

Los sistemas basados en SPARC y basados en x86 se inician con un archivo de almacenamiento de inicio, que consiste en una imagen del sistema de archivos que contiene los archivos necesarios para el inicio. Si el inicio se realiza desde un sistema de archivos raíz ZFS, los nombres de ruta del archivo de almacenamiento de inicio y del archivo de núcleo se resuelven en el sistema de archivos raíz seleccionado para el inicio.

El inicio desde un sistema de archivos ZFS es diferente del inicio desde un sistema de archivos UFS porque, con ZFS, un especificador de dispositivos identifica una agrupación de almacenamiento, no un solo sistema de archivos raíz. Una agrupación de almacenamiento puede contener varios sistemas de archivos raíz ZFS de inicio. Si el inicio se realiza desde ZFS, debe especificar un dispositivo de inicio y un sistema de archivos raíz en la agrupación identificada por el dispositivo de inicio.

De forma predeterminada, el sistema de archivos seleccionado para el inicio es el sistema identificado por la propiedad `bootfs` de la agrupación. Para anular esta selección predeterminada, especifique un sistema de archivos de inicio alternativo que se incluya en el comando `boot -Z` en un sistema basado en SPARC o seleccione un dispositivo de inicio alternativo del BIOS en un sistema basado en x86.

Inicio desde un disco alternativo en una agrupación raíz ZFS reflejada

Puede conectar un disco para crear una agrupación raíz ZFS reflejada después de la instalación. Para obtener más información sobre la creación de una agrupación raíz reflejada, consulte "Cómo configurar una agrupación raíz reflejada (SPARC o x86/VTOC)" en la página 125.

Revise los siguientes problemas conocidos relativos a agrupaciones raíz ZFS reflejadas:

- Puede iniciar desde distintos dispositivos en una agrupación raíz ZFS reflejada. Según la configuración de hardware, quizá deba actualizar la PROM o el BIOS para especificar otro dispositivo de inicio.

  Por ejemplo, puede iniciar desde cualquier disco (`c1t0d0s0` o `c1t1d0s0`) de esta agrupación.

  ```
  # zpool status
    pool: rpool
   state: ONLINE
   scrub: none requested
  config:

          NAME          STATE     READ WRITE CKSUM
          rpool         ONLINE       0     0     0
            mirror-0    ONLINE       0     0     0
              c1t0d0s0  ONLINE       0     0     0
              c1t1d0s0  ONLINE       0     0     0
  ```

  En un sistema basado en SPARC, especifique el disco alternativo en el indicador `ok`.

  ```
  ok boot /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/disk@1
  ```

  Tras reiniciar el sistema, confirme el dispositivo de inicio activo. Por ejemplo:

  ```
  SPARC# prtconf -vp | grep bootpath
          bootpath:  '/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/disk@1,0:a'
  ```

  En un sistema basado en x86, utilice una sintaxis similar a la siguiente:

```
x86# prtconf -v|sed -n '/bootpath/,/value/p'
        name='bootpath' type=string items=1
            value='/pci@0,0/pci8086,25f8@4/pci108e,286@0/disk@0,0:a'
```

- En un sistema basado en x86, seleccione un disco alternativo en la agrupación raíz ZFS reflejada en el pertinente menú BIOS.
- **SPARC o x86**: si reemplaza un disco de agrupación raíz mediante el comando `zpool replace`, debe instalar la información de inicio en el disco recientemente reemplazado mediante el comando `bootadm`. Si crea una agrupación raíz ZFS reflejada con el método de instalación inicial o si utiliza el comando `zpool attach` para adjuntar un disco a la agrupación raíz, este paso no es necesario. La sintaxis de `bootadm` es la siguiente:

  ```
  # bootadm install-bootloader
  ```

  Si desea instalar el cargador de inicio en una agrupación raíz alternativa, utilice la opción `-P`.

  ```
  # bootadm install-bootloader -P rpool2
  ```

  Si desea instalar el cargador de inicio GRUB antiguo, utilice el comando `installgrub` antiguo.

  ```
  x86# installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdsk/c0t1d0s0
  ```

- Puede iniciar desde distintos dispositivos en una agrupación raíz ZFS reflejada. Según la configuración de hardware, quizá deba actualizar la PROM o el BIOS para especificar otro dispositivo de inicio.

  Por ejemplo, puede iniciar desde cualquier disco (`c1t0d0s0` o `c1t1d0s0`) de esta agrupación.

  ```
  # zpool status
    pool: rpool
   state: ONLINE
   scrub: none requested
  config:

          NAME          STATE     READ WRITE CKSUM
          rpool         ONLINE       0     0     0
            mirror-0    ONLINE       0     0     0
              c1t0d0s0  ONLINE       0     0     0
              c1t1d0s0  ONLINE       0     0     0
  ```

  En un sistema basado en SPARC, especifique el disco alternativo en el indicador `ok`.

  ```
  ok boot /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/disk@1
  ```

  Tras reiniciar el sistema, confirme el dispositivo de inicio activo. Por ejemplo:

  ```
  SPARC# prtconf -vp | grep bootpath
          bootpath:  '/pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/disk@1,0:a'
  ```

  En un sistema basado en x86, utilice una sintaxis similar a la siguiente:

  ```
  x86# prtconf -v|sed -n '/bootpath/,/value/p'
          name='bootpath' type=string items=1
              value='/pci@0,0/pci8086,25f8@4/pci108e,286@0/disk@0,0:a'
  ```

- En un sistema basado en x86, seleccione un disco alternativo en la agrupación raíz ZFS reflejada en el pertinente menú BIOS.

Inicio de un sistema de archivos raíz ZFS en un sistema basado en SPARC

En un sistema basado en SPARC con varios entornos de inicio ZFS, puede iniciar el sistema desde cualquier entorno de inicio mediante el comando `beadm activate`.

Durante el proceso de instalación y activación con `beadm`, el sistema de archivos raíz ZFS se designa automáticamente con la propiedad `bootfs`.

En una agrupación pueden existir varios sistemas de archivos de inicio. De forma predeterminada, la entrada del sistema de archivos de inicio del archivo */boot/menu.lst/*`boot/menu.lst` se identifica mediante la propiedad `bootfs` de la agrupación. Sin embargo, una entrada de `menu.lst` puede contener un comando `bootfs`, que especifica un sistema de archivos alternativo de la agrupación. De esta manera, el archivo `menu.lst` puede contener entradas de varios sistemas de archivos raíz dentro de la agrupación.

Cuando se instala un sistema con un sistema de archivos raíz ZFS, se agrega una entrada similar a la siguiente al archivo `menu.lst`:

```
title Oracle Solaris 11.1 SPARC
bootfs rpool/ROOT/solaris
```

Cuando se crea un nuevo entorno de inicio, se actualiza automáticamente el archivo `menu.lst`.

```
title Oracle Solaris 11.1 SPARC
bootfs rpool/ROOT/solaris
title solaris
bootfs rpool/ROOT/solaris2
```

En un sistema basado en SPARC, puede seleccionar el entorno de inicio desde el cual desea iniciar, de la siguiente manera:

- Después de activar un entorno de inicio ZFS, puede utilizar el comando boot `-L` para obtener una lista de los sistemas de archivos de inicio en una agrupación ZFS. A continuación, puede seleccionar en la lista uno de los sistemas de archivos de inicio. Se muestran instrucciones detalladas para iniciar dicho sistema de archivos. El sistema de archivos seleccionado se puede iniciar siguiendo esas instrucciones.
- Utilice el comando boot `-Z` *sistema de archivos* para iniciar un sistema de archivos ZFS específico.

Este método de inicio no activa el entorno de inicio de forma automática. Después de iniciar el entorno de inicio con la sintaxis `-L` y `-Z`, deberá activar este entorno de inicio para seguir realizado los inicios desde allí automáticamente.

EJEMPLO 4–1 Inicio desde un entorno de inicio ZFS específico

Si dispone de varios entornos de inicio ZFS en una agrupación de almacenamiento ZFS en el dispositivo de inicio del sistema, puede utilizar el comando `beadm activate` para especificar un entorno de inicio predeterminado.

Por ejemplo, los siguientes entornos de inicio ZFS están disponibles como se describe en la salida de `beadm`:

```
# beadm list
BE        Active Mountpoint Space Policy Created
--        ------ ---------- ----- ------ -------
solaris   NR     /          3.80G static 2012-07-20 10:25
solaris-2 -      -          7.68M static 2012-07-19 13:44
```

Si dispone de varios entornos de inicio ZFS en el sistema basado en SPARC, puede utilizar el comando `boot -L`. Por ejemplo:

```
ok boot -L
Boot device: /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/disk@0,0:a  File and args: -L
1 Oracle Solaris 11.1 SPARC
2 solaris
Select environment to boot: [ 1 - 2 ]: 1

To boot the selected entry, invoke:
boot [<root-device>] -Z rpool/ROOT/solaris-2

Program terminated
ok boot -Z rpool/ROOT/solaris-2
```

Tenga en cuenta que un entorno de inicio que se inicia con el comando anterior no está activado para el siguiente reinicio. Si desea seguir iniciando el sistema de forma automática desde el entorno de inicio seleccionado durante la operación `boot -Z`, deberá activarlo.

Inicio desde un sistema de archivos raíz ZFS en un sistema basado en x86

En Oracle Solaris 11, un sistema x86 está instalado con GRUB antiguo. Las siguientes entradas se agregan al archivo */pool-name* `/boot/grub/menu.lst` durante el proceso de instalación o la operación `beadm activate` para iniciar ZFS automáticamente:

```
title solaris
bootfs rpool/ROOT/solaris
kernel$ /platform/i86pc/kernel/amd64/unix -B $ZFS-BOOTFS
module$ /platform/i86pc/amd64/boot_archive
title solaris-1
bootfs rpool/ROOT/solaris-1
kernel$ /platform/i86pc/kernel/amd64/unix -B $ZFS-BOOTFS
module$ /platform/i86pc/amd64/boot_archive
```

Si el dispositivo que GRUB identifica como dispositivo de inicio contiene una agrupación de almacenamiento ZFS, el archivo `menu.lst` se utiliza para crear el menú GRUB.

En el caso de un sistema basado en x86 con varios entornos de inicio ZFS, el entorno de inicio se puede seleccionar en el menú GRUB. Si el sistema de archivos raíz correspondiente a esta entrada de menú es un sistema de archivos ZFS, se agrega la siguiente opción.

```
-B $ZFS-BOOTFS
```

En Oracle Solaris 11.1, un sistema basado en x86 se instala con GRUB2. El archivo `menu.lst` se reemplaza con el archivo `/rpool/boot/grub/grub.cfg` , pero este archivo no debe editarse manualmente. Utilice los subcomandos de `bootadm` para agregar, cambiar y eliminar entradas de menú.

Para obtener más información sobre la modificación de elementos del menú de GRUB, consulte *Inicio y cierre de sistemas Oracle Solaris 11.1*.

EJEMPLO 4–2 x86: inicio de un sistema de archivos ZFS

Al realizar el inicio desde un sistema de archivos raíz ZFS en un sistema GRUB2, el dispositivo raíz se especifica de la siguiente manera:

```
# bootadm list-menu
the location of the boot loader configuration files is: /rpool/boot/grub
default 0
console text
timeout 30
0 Oracle Solaris 11.1
```

Cuando el inicio se realiza desde un sistema de archivos ZFS en un sistema GRUB antiguo, el dispositivo raíz se especifica mediante el parámetro de inicio `-B $ZFS-BOOTFS`. Por ejemplo:

```
title solaris
bootfs rpool/ROOT/solaris
kernel$ /platform/i86pc/kernel/amd64/unix -B $ZFS-BOOTFS
module$ /platform/i86pc/amd64/boot_archive
title solaris-1
bootfs rpool/ROOT/solaris-1
kernel$ /platform/i86pc/kernel/amd64/unix -B $ZFS-BOOTFS
module$ /platform/i86pc/amd64/boot_archive
```

EJEMPLO 4–3 x86: reinicio rápido de un sistema de archivos raíz ZFS

La función de reinicio rápido permite reiniciar un sistema en cuestión de segundos en los sistemas basados en x86. Con la función de reinicio rápido, puede reiniciar un sistema en un nuevo núcleo sin las demoras prolongadas que pueden generar el BIOS y el cargador de inicio. La capacidad de reinicio rápido de un sistema reduce significativamente el tiempo de inactividad y mejora la eficacia.

EJEMPLO 4–3 x86: reinicio rápido de un sistema de archivos raíz ZFS *(Continuación)*

Debe utilizar de todos modos el comando `init 6` en las transiciones entre entornos de inicio con el comando `beadm activate`. Para otras operaciones del sistema en las que el comando `reboot` resulta adecuado, puede utilizar el comando `reboot -f`. Por ejemplo:

```
# reboot -f
```

Inicio para fines de recuperación en un entorno raíz ZFS

Utilice el procedimiento siguiente si necesita iniciar el sistema para recuperarse de la pérdida de una contraseña root o de un problema similar.

▼ Cómo iniciar el sistema para fines de recuperación

Proceda de la siguiente manera para resolver una dificultad con un problema `menu.lst` o un problema de contraseña de usuario root. Si necesita reemplazar un disco en una agrupación raíz, consulte "Cómo reemplazar un disco en una agrupación raíz ZFS (SPARC o x86/VTOC)" en la página 128. En el caso de que necesite realizar una restauración completa (bare-metal) del sistema consulte Capítulo 11, "Archivado de instantáneas y recuperación de agrupaciones raíz".

1 Seleccione el método de inicio apropiado:

- x86: Live Media: inicie desde el medio de instalación y utilice un terminal de GNOME para el procedimiento de recuperación.
- Instalación de texto en SPARC: inicie desde el medio de instalación o desde la red, y seleccione la opción `3 Shell` desde la pantalla de instalación de texto.
- Instalación de texto en x86: en el menú de GRUB, seleccione la entrada de inicio `Text Installer and command line` y, a continuación, seleccione la opción `3 Shell` desde la pantalla de instalación de texto.
- Instalación automatizada en SPARC: utilice el siguiente comando para iniciar directamente desde un menú de instalación que permita salir a un shell.

  ```
  ok boot net:dhcp
  ```
- Instalación automatizada en x86: el inicio desde un servidor de instalación en la red requiere un inicio de PXE. Seleccione la entrada `Text Installer and command line` del menú de GRUB. A continuación, seleccione la opción `3 Shell` desde la pantalla de instalación de texto.

Por ejemplo, después de iniciar el sistema, seleccione la opción `3 Shell`.

```
1  Install Oracle Solaris
2  Install Additional Drivers
```

```
        3  Shell
        4  Terminal type (currently xterm)
        5  Reboot

Please enter a number [1]: 3
To return to the main menu, exit the shell
#
```

2 Seleccione el problema de recuperación de inicio:

- Para resolver un shell de raíz incorrecta, inicie el sistema en modo de un solo usuario y corrija la entrada shell en el archivo `/etc/passwd`.

  En un sistema x86, edite la entrada de inicio seleccionada y agregue la opción `-s`.

  Por ejemplo, en un sistema SPARC, apague el sistema e inicie en modo de usuario único. Una vez que se haya conectado como usuario root, edite el archivo `/etc/passwd` y corrija la entrada de shell raíz.

  ```
  # init 0
  ok boot -s
  Boot device: /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/disk@0,0:a ...
  SunOS Release 5.11 Version 11.1 64-bit
  Copyright (c) 1983, 2012, Oracle and/or its affiliates. All rights reserved.
  Booting to milestone "milestone/single-user:default".
  Hostname: tardis.central
  Requesting System Maintenance Mode
  SINGLE USER MODE

  Enter user name for system maintenance (control-d to bypass): root
  Enter root password (control-d to bypass): xxxx
  single-user privilege assigned to root on /dev/console.
  Entering System Maintenance Mode

  Aug  3 15:46:21 su: 'su root' succeeded for root on /dev/console
  Oracle Corporation      SunOS 5.11      11.1    October 2012
  su: No shell /usr/bin/mybash.  Trying fallback shell /sbin/sh.
  root@tardis.central:~# TERM =vt100; export TERM
  root@tardis.central:~# vi /etc/passwd
  root@tardis.central:~# <Press control-d>
  logout
  svc.startd: Returning to milestone all.
  ```

- Resuelva un problema que impide que un sistema basado en x86 se inicio. .

  En primer lugar, debe iniciar desde el medio o la red mediante uno de los métodos de inicio que se describen en el paso 1. Luego, importe la agrupación raíz y corrija una entrada GRUB, por ejemplo.

  Puede utilizar el comando `bootadm list-menu` para enumerar y modificar las entradas de GRUB2. También puede utilizar el subcomando `set-menú` para cambiar una entrada de inicio. Para obtener más información, consulte `bootadm(1M)`.

  ```
  x86# zpool import -f rpool
  x86# bootadm list-menu
  x86# bootadm set-menu default=1
  x86# exit
          1  Install Oracle Solaris
  ```

```
        2  Install Additional Drivers
        3  Shell
        4  Terminal type (currently sun-color)
        5  Reboot

Please enter a number [1]: 5
```

Confirme que el sistema se inicie correctamente.

- Resuelva una contraseña de usuario root desconocida que impide que se conecte al sistema.

  En primer lugar, debe iniciar desde el medio o la red mediante uno de los métodos de inicio que se describen en el paso 1. A continuación, importe la agrupación raíz (`rpool`) y monte el entorno de inicio para eliminar la entrada de la contraseña root. Este proceso es idéntico tanto en las plataformas SPARC como x86.

```
# zpool import -f rpool
# beadm list
be_find_current_be: failed to find current BE name
be_find_current_be: failed to find current BE name
BE        Active Mountpoint Space  Policy Created
--        ------ ---------- -----  ------ -------
solaris   -      -          46.95M static 2012-07-20 10:25
solaris-2 R      -          3.81G  static 2012-07-19 13:44
# mkdir /a
# beadm mount solaris-2 /a
# TERM=vt100
# export TERM
# cd /a/etc
# vi shadow
<Carefully remove the unknown password>
# cd /
# beadm umount solaris-2
# halt
```

  Vaya al siguiente paso para configurar la contraseña root.

3 Para configurar la contraseña root, inicie en modo de un solo usuario y defina la contraseña.

En este paso se asume que ha eliminado una contraseña root desconocida en el paso anterior.

En un sistema basado en x86, edite la entrada de inicio seleccionada y agregue la opción `-s`.

En un sistema basado en SPARC, inicie el sistema en modo de un solo usuario, inicie sesión como usuario root y establezca la contraseña de usuario root. Por ejemplo:

```
ok boot -s
Boot device: /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/disk@0,0:a ...
SunOS Release 5.11 Version 11.1 64-bit
Copyright (c) 1983, 2012, Oracle and/or its affiliates. All rights reserved
Booting to milestone "milestone/single-user:default".

Enter user name for system maintenance (control-d to bypass): root
Enter root password (control-d to bypass): <Press return>
single-user privilege assigned to root on /dev/console.
Entering System Maintenance Mode

Jul 20 14:09:59 su: 'su root' succeeded for root on /dev/console
```

```
Oracle Corporation      SunOS 5.11      11.1    October 2012
root@tardis.central:~# passwd -r files root
New Password: xxxxxx
Re-enter new Password: xxxxxx
passwd: password successfully changed for root
root@tardis.central:~# <Press control-d>
logout
svc.startd: Returning to milestone all.
```

◆ ◆ ◆ CAPÍTULO 5

Administración de sistemas de archivos ZFS de Oracle Solaris

Este capítulo ofrece información detallada sobre la administración de sistemas de archivos ZFS de Oracle Solaris. En este capítulo se incluyen conceptos como la disposición jerárquica del sistema de archivos, la herencia de propiedades, así como la administración automática de puntos de montaje y cómo compartir interacciones.

Este capítulo se divide en las secciones siguientes:

Administración de sistemas de archivos AFS (descripción general)

Un sistema de archivos ZFS se genera encima de una agrupación de almacenamiento. Los sistemas de archivos se pueden crear y destruir dinámicamente sin necesidad de asignar ni dar formato a ningún espacio en el disco subyacente. Debido a que los sistemas de archivos son tan ligeros y a que son el punto central de administración en ZFS, puede crear muchos de ellos.

Los sistemas de archivos ZFS se administran mediante el comando `zfs`. El comando `zfs` ofrece un conjunto de subcomandos que ejecutan operaciones específicas en los sistemas de archivos. Este capítulo describe estos subcomandos detalladamente. Las instantáneas, los volúmenes y los

clones también se administran mediante este comando, pero estas funciones sólo se explican brevemente en este capítulo. Para obtener información detallada sobre instantáneas y clones, consulte el Capítulo 6, "Uso de clones e instantáneas de Oracle Solaris ZFS". Para obtener información detallada sobre volúmenes ZFS, consulte "Volúmenes de ZFS" en la página 285.

Nota – El término *conjunto de datos* se utiliza en este capítulo como término genérico para referirse a un sistema de archivos, instantánea, clon o volumen

Creación, destrucción y cambio de nombre de sistemas de archivos ZFS

Los sistemas de archivos ZFS se pueden crear y destruir mediante los comandos `zfs create` y `zfs destroy`, respectivamente. Mediante el comando `zfs rename` se puede cambiar el nombre a los sistemas de archivos ZFS.

Creación de un sistema de archivos ZFS

Los sistemas de archivos ZFS se crean mediante el comando `zfs create`. El subcomando `create` toma un único argumento: el nombre del sistema de archivos que crear. El nombre del sistema de archivos se especifica como nombre de ruta que comienza por el nombre de la agrupación:

pool-name/[filesystem-name/]filesystem-name

El nombre de grupo y los nombres del sistema de archivos inicial de la ruta identifican la ubicación en la jerarquía donde se creará el nuevo sistema de archivos. El último nombre de la ruta identifica el nombre del sistema de archivos que se creará. El nombre del sistema de archivos debe seguir las convenciones de denominación establecidas en "Requisitos de asignación de nombres de componentes de ZFS" en la página 32.

El cifrado de un sistema de archivos ZFS debe estar habilitado cuando se crea el sistema de archivos. Para obtener más información sobre el cifrado de un sistema de archivos ZFS, consulte "Cifrado de sistemas de archivos ZFS" en la página 205.

En el ejemplo siguiente, un sistema de archivos denominado `jeff` se crea en el sistema de archivos `tank/home`.

```
# zfs create tank/home/jeff
```

ZFS monta de forma automática el sistema de archivos recién creado si se crea correctamente. De forma predeterminada, los sistemas de archivos se montan como */conjunto de datos*, mediante la ruta proporcionada para el nombre del sistema de archivos en el subcomando `create`. En este ejemplo, el sistema de archivos recién creado `jeff` se monta en `/tank/home/jeff`. Para obtener más información sobre puntos de montaje que se administran automáticamente, consulte "Administración de puntos de montaje de ZFS" en la página 183.

Para obtener más información sobre el comando `zfs create`, consulte `zfs(1M)`.

Las propiedades del sistema de archivos pueden establecerse al crear dicho sistema de archivos.

En el ejemplo siguiente se crea un punto de montaje de `/export/zfs` para el sistema de archivos `tank/home`:

```
# zfs create -o mountpoint=/export/zfs tank/home
```

Para obtener más información sobre las propiedades del sistema de archivos, consulte "Introducción a las propiedades de ZFS" en la página 151.

Destrucción de un sistema de archivos ZFS

Para destruir un sistema de archivos ZFS, utilice el comando `zfs destroy`. El sistema de archivos destruido se desmonta automáticamente y se anula la compartición. Para obtener más información sobre puntos de montaje o recursos compartidos administrados automáticamente, consulte "Puntos de montaje automáticos" en la página 184.

En el ejemplo siguiente, se destruye el sistema de archivos `tank/home/mark`:

```
# zfs destroy tank/home/mark
```

Precaución – No aparece ningún mensaje de confirmación con el subcomando `destroy`. Utilícelo con extrema precaución.

Si el sistema de archivos que se desea destruir está ocupado y no se puede desmontar, el comando `zfs destroy` falla. Para destruir un sistema de archivos activo, utilice la opción `-f`. Úsela con precaución, puesto que puede desmontar, destruir y anular la compartición de sistemas de archivos activos, lo que provoca un comportamiento inesperado de la aplicación.

```
# zfs destroy tank/home/matt
cannot unmount 'tank/home/matt': Device busy

# zfs destroy -f tank/home/matt
```

El comando `zfs destroy` también falla si un sistema de archivos tiene descendientes. Para destruir repetidamente un sistema de archivos y todos sus descendientes, utilice la opción `-r`. Una destrucción repetitiva también destruye las instantáneas, por lo que debe utilizar esta opción con precaución.

```
# zfs destroy tank/ws
cannot destroy 'tank/ws': filesystem has children
use '-r' to destroy the following datasets:
tank/ws/jeff
tank/ws/bill
tank/ws/mark
# zfs destroy -r tank/ws
```

Si el sistema de archivos que se debe destruir tiene elementos dependientes indirectos, falla incluso el comando de destrucción repetitiva. Para forzar la destrucción de *todos* los dependientes, incluidos los sistemas de archivos clonados fuera de la jerarquía de destino, se debe utilizar la opción `-R`. Esta opción se debe utilizar con sumo cuidado.

```
# zfs destroy -r tank/home/eric
cannot destroy 'tank/home/eric': filesystem has dependent clones
use '-R' to destroy the following datasets:
tank//home/eric-clone
# zfs destroy -R tank/home/eric
```

Precaución – No aparece ningún mensaje de confirmación con las opciones `-f`, `-r` o `-R` para el comando `zfs destroy`, por lo que debe utilizarlas con cuidado.

Para obtener información detallada sobre instantáneas y clones, consulte el Capítulo 6, "Uso de clones e instantáneas de Oracle Solaris ZFS".

Cambio de nombre de un sistema de archivos ZFS

Mediante el comando `zfs rename` se puede cambiar el nombre a los sistemas de archivos. Con el subcomando `rename` se pueden efectuar las operaciones siguientes:

- Cambiar el nombre de un sistema de archivos.
- Cambiar la ubicación del sistema de archivos en la jerarquía ZFS.
- Cambiar el nombre de un sistema de archivos y cambiar su ubicación dentro de la jerarquía ZFS.

En el ejemplo siguiente, se utiliza el subcomando `rename` para cambiar el nombre del sistema de archivos `eric` por `eric_old`:

```
# zfs rename tank/home/eric tank/home/eric_old
```

El ejemplo siguiente muestra cómo utilizar `zfs rename` para cambiar la ubicación de un sistema de archivos:

```
# zfs rename tank/home/mark tank/ws/mark
```

En este ejemplo, el sistema de archivos `mark` se reubica de `tank/home` a `tank/ws`. Si reubica un sistema de archivos mediante rename, la nueva ubicación debe estar en la misma agrupación y tener espacio suficiente en el disco para albergar este nuevo sistema de archivos. Si la nueva ubicación no tiene espacio suficiente en el disco, posiblemente por haber alcanzado su cuota, la operación `rename` fallará.

Para obtener más información sobre las cuotas, consulte "Configuración de cuotas y reservas de ZFS" en la página 199.

La operación `rename` intenta una secuencia de desmontar/volver a montar para el sistema de archivos y los sistemas de archivos descendientes. El comando `rename` falla si la operación no puede desmontar un sistema de archivos activo. Si se produce este problema, deberá forzar el desmontaje del sistema de archivos.

Para obtener más información sobre el cambio de nombre de las instantáneas, consulte "Cambio de nombre de instantáneas de ZFS" en la página 222.

Introducción a las propiedades de ZFS

Las propiedades son para el mecanismo principal que utiliza para controlar el comportamiento de los sistemas de archivos, volúmenes, instantáneas y clones. A menos que se indique lo contrario, las propiedades que se definen en esta sección se aplican a todos los tipos de conjuntos de datos.

- "Propiedades nativas de sólo lectura de ZFS" en la página 167
- "Propiedades nativas de ZFS configurables" en la página 168
- "Propiedades de usuario de ZFS" en la página 175

Las propiedades se dividen en dos tipos: nativas y definidas por el usuario. Las propiedades nativas proporcionan estadísticas internas o controlan el comportamiento del sistema de archivos ZFS. Asimismo, las propiedades nativas son configurables o de sólo lectura. Las propiedades del usuario no repercuten en el comportamiento del sistema de archivos ZFS, pero puede usarlas para anotar conjuntos de datos de forma que tengan sentido en su entorno. Para obtener más información sobre las propiedades del usuario, consulte "Propiedades de usuario de ZFS" en la página 175.

La mayoría de las propiedades configurables también se pueden heredar. Una propiedad que se puede heredar es la que, cuando se establece en un sistema de archivos principal, se propaga a todos sus descendientes.

Todas las propiedades heredables tienen un origen asociado que indica la forma en que se ha obtenido una propiedad. El origen de una propiedad puede tener los valores siguientes:

`local`	Indica que la propiedad se ha establecido explícitamente en el conjunto de datos mediante el comando `zfs set`, tal como se describe en "Configuración de propiedades de ZFS" en la página 178.
`inherited from` *dataset-name*	Indica que la propiedad se ha heredado del superior nombrado.
`default`	Indica que el valor de la propiedad no se ha heredado o establecido localmente. Este origen es el resultado de que ningún superior tiene la propiedad como `local` de origen.

La tabla siguiente identifica las propiedades del sistema de archivos ZFS nativo configurable y de sólo lectura. Las propiedades nativas de sólo lectura se identifican como tales. Todas las demás propiedades nativas que se enumeran en esta tabla son configurables. Para obtener información sobre las propiedades del usuario, consulte "Propiedades de usuario de ZFS" en la página 175.

TABLA 5–1 Descripciones de propiedades nativas de ZFS

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`aclinherit`	Cadena	`secure`	Controla cómo se heredan las entradas de lista de control de acceso (ACL) cuando se crean los archivos y los directorios. Los valores son `discard`, `noallow`, `secure` y `passthrough`. Para obtener una descripción de estos valores, consulte "Propiedades de ACL" en la página 248.
`aclmode`	Cadena	`groupmask`	Controla cómo se modifica una entrada de lista de control de acceso (ACL) durante una operación de `chmod`. Los valores son `discard`, `groupmask` y `passthrough`. Para obtener una descripción de estos valores, consulte "Propiedades de ACL" en la página 248.
`atime`	Booleano	`on`	Controla si la hora de acceso de los archivos se actualiza cuando se leen. Si se desactiva esta propiedad, se evita la generación de tráfico de escritura al leer archivos y se puede mejorar considerablemente el rendimiento, si bien esto podría confundir a los programas de envío de correo y otras utilidades similares.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`available`	Número	N/A	Propiedad de sólo lectura que identifica la cantidad de espacio disponible en el disco para un sistema de archivos y todos los subordinados, suponiendo que no hay otra actividad en la agrupación. Como el espacio en el disco se comparte en una agrupación, el espacio disponible puede verse limitado por varios factores, como el tamaño físico de la agrupación, las cuotas, las reservas u otros conjuntos de datos de la agrupación. La abreviatura de la propiedad es `avail`. Para obtener más información sobre el cálculo de espacio, consulte "Cálculo del espacio de ZFS" en la página 34.
`canmount`	Booleano	`on`	Controla si un sistema de archivos determinado se puede montar con el comando `zfs mount`. Esta propiedad se puede establecer en cualquier sistema de archivos y la propiedad no es heredable. No obstante, cuando esta propiedad está establecida en `off`, los sistemas de archivos descendientes se pueden heredar, pero el sistema de archivos nunca se monta. Si se establece la opción `noauto`, un sistema de archivos sólo se puede montar y desmontar de manera explícita. El sistema de archivos no se monta automáticamente al crearlo o importarlo, ni se monta con el comando `zfs mount- a` ni se desmonta con el comando `zfs unmount-a`. Para obtener más información, consulte "Propiedad `canmount`" en la página 170.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`casesensitivity`	Cadena	`mixed`	Esta propiedad indica si el algoritmo que coincide con el nombre de archivo utilizado por el sistema de archivos debe ser `casesensitive` o `caseinsensitive`, o si debe permitir una combinación de ambos estilos de coincidencia (`mixed`). Tradicionalmente, los sistemas de archivos UNIX y POSIX tienen nombres de archivo que distinguen entre mayúsculas y minúsculas. El valor `mixed` para esta propiedad indica que el sistema de archivos puede admitir solicitudes de comportamientos de coincidencias con y sin distinción de mayúsculas y minúsculas. Actualmente, el comportamiento de coincidencia con distinción de mayúsculas y minúsculas en un sistema de archivos que admite un comportamiento mixto está limitado al producto de servidor Oracle Solaris SMB. Para obtener más información sobre el uso del valor `mixed`, consulte "Propiedad `casesensitivity`" en la página 170. Independientemente de la configuración de la propiedad `casesensitivity`, el sistema de archivos conserva las mayúsculas y minúsculas del nombre especificado para crear un archivo. Esta propiedad no se podrá cambiar una vez creado el sistema de archivos.
`suma de comprobación`	Cadena	`on`	Controla la suma de comprobación utilizada para verificar la integridad de los datos. El valor predeterminado es `on`, que selecciona automáticamente un algoritmo adecuado, actualmente `fletcher4`. Los valores son `on, off, fletcher2, fletcher4, sha256` y `sha256+mac`. El valor `off` desactiva la comprobación de integridad en los datos del usuario. No se recomienda el valor `off`.
`compression`	Cadena	`off`	Activa o desactiva la compresión de este conjunto de datos. Los valores son `on`, `off` y `lzjb`, `gzip` o `gzip-`*N*. En la actualidad, configurar esta propiedad en `lzjb`, `gzip` o `gzip-` *N* equivale a establecerla en `on`. Activar la compresión en un sistema de archivos en el que ya hay datos sólo comprime los datos nuevos. Los datos que existan están sin comprimir. La abreviatura de la propiedad es `compress`.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`compressratio`	Número	N/A	Propiedad de sólo lectura que identifica el índice de compresión alcanzado para un conjunto de datos, expresado como multiplicador. La compresión se puede activar ejecutando el comando `zfs set compression=on` *dataset*. El valor se calcula a partir del tamaño lógico de todos los archivos y la cantidad de datos físicos a los que se hace referencia. Incluye grabaciones explícitas mediante el uso de la propiedad `compression`.
`copies`	Número	`1`	Establece la cantidad de copias de datos de usuarios por sistema de archivos. Los valores disponibles son `1`, `2`, o `3`. Estas copias son adicionales a cualquier redundancia de agrupación. El espacio en el disco que utilicen varias copias de datos de usuarios se carga en los pertinentes archivo y conjunto de datos, y se contabiliza en relación con las cuotas y reservas. Además, la propiedad `used` se actualiza si se activan varias copias. La configuración de esta propiedad debe considerarse al crear el sistema de archivos, puesto que, si se modifica la propiedad en cualquier sistema ya creado, sólo se afecta a los datos nuevos que se escriban.
`creation`	Cadena	N/A	Propiedad de sólo lectura que identifica la fecha y la hora de creación de este conjunto de datos.
`dedup`	Cadena	`off`	Controla la capacidad de eliminar datos duplicados en un sistema de archivos ZFS. Los valores posibles son `on`, `off`, `verify` y `sha256[,verify]`. La suma de comprobación predeterminada para la eliminación de datos duplicados es `sha256`. Para obtener más información, consulte "Propiedad dedup" en la página 172.
`devices`	Booleano	`on`	Controla si se pueden abrir los archivos de dispositivos en un sistema de archivos.
`cifrado`	Booleano	`off`	Controla si un sistema de archivos está cifrado. Un sistema de archivos cifrado significa que los datos están codificados y que el propietario del sistema necesita una clave para acceder a los datos.
`exec`	Booleano	`on`	Controla si se permite ejecutar programas en un sistema de archivos. Asimismo, si se establece en `off`, no se permiten las llamadas de `mmap(2)` con `PROT_EXEC`.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`keychangedate`	Cadena	none	Identifica la fecha del última cambio de clave de ajuste de una operación `zfs key -c` para el sistema de archivos especificado. Si no se produjo ninguna operación de cambio de clave, el valor de esta propiedad de sólo lectura es igual a la fecha de creación del sistema de archivos.
`keysource`	Cadena	none	Identifica el formato y la ubicación de la clave que se ajusta a las claves del sistema de archivos. Los valores de propiedad válidos son `raw`, `hex`, `passphrase,prompt` o *archivo*. La clave debe estar presente cuando el sistema de archivos se crea, se monta o se carga mediante el comando `zfs key -l`. Si el cifrado ha sido activado para un sistema de archivos nuevo, el valor de `keysource` predeterminado es `passphrase,prompt`.
`keystatus`	Cadena	none	Propiedad de sólo lectura que identifica el estado de la clave de cifrado del sistema de archivos. La disponibilidad de la clave de un sistema de archivos se indica mediante `available` o `unavailable`. Para los sistemas de archivos que no tienen activado el cifrado, se muestra la opción `none`.
`logbias`	Cadena	`latency`	Controla de qué manera ZFS optimiza las solicitudes síncronas para este sistema de archivos. Si `logbias` se establece en `latency`, ZFS utiliza los dispositivos de registro independientes de la agrupación, si los hay, para manejar las solicitudes con latencia baja. Si `logbias` se establece en `throughput`, ZFS no utiliza los dispositivos de registro independientes de la agrupación. En su lugar, ZFS optimiza las operaciones síncronas para el rendimiento global de la agrupación y el uso eficiente de recursos. El valor predeterminado es `latency`.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`mlslabel`	Cadena	None (Nada)	Consulte la propiedad `multilevel` para obtener una descripción del comportamiento de la propiedad `mlslabel` en sistemas de archivos de varios niveles. La siguiente descripción de `mlslabel` se aplica sistemas de archivos que no tienen varios niveles. Proporciona una etiqueta de sensibilidad que determina si un sistema de archivos se puede montar en una zona de Trusted Extensions. Si el sistema de archivos etiquetado coincide con la zona etiquetada, el sistema de archivos se puede montar y es posible acceder a él desde la zona etiquetada. El valor predeterminado es`none`. Esta propiedad se puede modificar solamente cuando Trusted Extensions está activado y solamente con el privilegio adecuado.
`mounted`	Booleano	N/A	Propiedad de sólo lectura que indica si este sistema de archivos, un clon o una instantánea se encuentra montada. Esta propiedad no se aplica a los volúmenes. El valor puede ser `yes` o `no`.
`mountpoint`	Cadena	N/A	Controla el punto de montaje utilizado para este sistema de archivos. Si la propiedad `mountpoint` se cambia para un sistema de archivos, se desmontan éste y cualquier descendiente que herede el punto de montaje. Si el valor nuevo es `legacy`, permanecen desmontados. En cambio, se vuelven a montar automáticamente en la nueva ubicación si la propiedad era `legacy` o `none`, o bien si estaban montados antes de que cambiara la propiedad. Asimismo, cualquier sistema de archivos compartidos está sin compartir y compartido en la nueva ubicación. Para obtener más información sobre el uso de esta propiedad, consulte "Administración de puntos de montaje de ZFS" en la página 183.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`multilevel`	Booleano	`off`	Esta propiedad solamente se puede utilizar en un sistema con Trusted Extensions activado. El valor predeterminado es off. Los objetos de un sistema de archivos de varios niveles son etiquetados individualmente con un atributo de etiqueta que se genera de forma automática. Los objetos pueden volver a etiquetarse cambiando este atributo de etiqueta, con las interfaces `setlabel` o `setflabel`. Un sistema de archivos raíz, un sistema de archivos de Oracle Solaris o un sistema de archivos que contiene un código de Solaris empaquetado no pueden tener varios niveles. En un sistema de archivos de varios niveles, existen diferencias en la propiedad `mlslabel`. El valor `mlslabel` define la etiqueta más alta posible para los objetos del sistema de archivos. No se permite el intento de crear un archivo (o cambiar la etiqueta de un archivo) con una etiqueta más alta que el valor `mlslabel`. La política de montaje basada en el valor `mlslabel` no se aplica a un sistema de archivos de varios niveles. Para un sistema de archivos de varios niveles, la propiedad `mlslabel` se puede definir de forma explícita al crear el sistema de archivos. De lo contrario, se crea automáticamente la propiedad `mlslabel` predeterminada de `ADMIN_HIGH`. Después de crear un sistema de archivos de varios niveles, la propiedad `mlslabel` se puede cambiar, pero no se puede definir en una etiqueta inferior (`none`), ni tampoco se puede eliminar.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`primarycache`	Cadena	`all`	Controla la información que se guarda en la caché primaria (ARC). Los valores posibles son `all`, `none` y `metadata`. Si se establece en `all`, los datos de usuario y los metadatos se almacenan en la memoria caché. Si se establece en `none`, no se completan datos de usuario ni los metadatos se almacenan en la memoria caché. Si se establece en `metadata`, sólo los metadatos se almacenan en la memoria caché. Cuando estas propiedades se establecen en sistemas de archivos existentes, sólo la nueva E/S se basa en la memoria caché en función del valor de estas propiedades. Algunos entornos de la base de datos pueden beneficiarse de no almacenar datos de usuario en la memoria caché. Debe determinar si es adecuado configurar las propiedades de memoria caché para su entorno.
`nbmand`	Booleano	`off`	Controla si el sistema de archivos debe montarse con bloqueos `nbmand` (obligatorio sin bloqueo). Esta propiedad es sólo para clientes de SMB. Los cambios realizados en esta propiedad sólo surten efecto cuando el sistema de archivos se desmonta y se vuelve a montar.
`normalization`	Cadena	None (Nada)	Esta propiedad indica si un sistema de archivos debe realizar una normalización de los nombres de archivo de unicode cuando se comparan dos nombres de archivo, e indica qué algoritmo de normalización debería utilizarse. Los nombres de archivo siempre se almacenan sin modificaciones, y los nombres están normalizados como parte de cualquier proceso de comparación. Si se establece esta propiedad en un valor legal que no es `none` y la propiedad `utf8only` no se especificó, la propiedad `utf8only` se configura automáticamente en `on`. El valor predeterminado de la propiedad `normalization` es `none`. Esta propiedad no se podrá cambiar una vez creado el sistema de archivos.
`origin`	Cadena	N/A	Propiedad de sólo lectura para volúmenes o sistemas de archivos clónicos que identifica la instantánea a partir de la cual se ha creado el clon. No se puede destruir el origen (ni siquiera con las opciones `-r` o `-f`) en tanto exista un clon. Los sistemas de archivos no clónicos tienen la propiedad de origen establecida en `none`.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`quota`	Número (o `none`)	`none`	Limita la cantidad de espacio en el disco que un sistema de archivos y sus descendientes pueden consumir. Esta propiedad fuerza un límite físico sobre la cantidad de espacio utilizado, incluido todo el espacio consumido por descendientes, como los sistemas de archivos y las instantáneas. La configuración de una cuota en un descendiente de un sistema de archivos que ya tiene una no anula la cuota del antecesor, sino que impone un límite adicional. Las cuotas no se pueden establecer en volúmenes, ya que la propiedad `volsize` representa una cuota implícita. Para obtener información sobre la configuración de cuotas, consulte "Establecimiento de cuotas en sistemas de archivos ZFS" en la página 200.
`rekeydate`	Cadena	N/A	Propiedad de sólo lectura que indica la fecha del último cambio de clave de cifrado de datos de una operación `zfs key -K` o `zfs clone - K` en este sistema de archivos. Si no se ha realizado ninguna operación de `rekey`, el valor de esta propiedad es el mismo que el de la fecha de `creation`.
`readonly`	Booleano	`off`	Controla si un conjunto de datos se puede modificar. Si se establece en `on`, no se pueden efectuar modificaciones. La abreviatura de la propiedad es `rdonly`.
`recordsize`	Número	`128K`	Especifica un tamaño de bloque sugerido para los archivos del sistema de archivos. La abreviatura de la propiedad es `recsize`. Para obtener información detallada, consulte "Propiedad `recordsize`" en la página 173.
`referenced`	Número	N/A	Propiedad de sólo lectura que identifica la cantidad de datos a los que puede acceder un conjunto de datos, que se pueden compartir o no con otros conjuntos de datos de la agrupación. Cuando se crea una instantánea o un clon, inicialmente hace referencia a la misma cantidad de espacio en el disco que la instantánea o el sistema de archivos del que se creó, porque su contenido es idéntico. La abreviatura de la propiedad es `refer`.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`refquota`	Número (o none)	`none`	Establece la cantidad de espacio en el disco que puede consumir un conjunto de datos. Esta propiedad impone un límite físico en la cantidad de espacio que se usa. Este límite físico no incluye el espacio en el disco usado por los descendientes, como instantáneas y clones.
`refreservation`	Número (o none)	`none`	Establece la cantidad mínima de espacio en el disco que se garantiza a un conjunto de datos, sin incluir descendientes como las instantáneas o los clones. Cuando la cantidad de espacio en el disco utilizado aparece bajo este valor, se considera que el conjunto de datos utiliza la cantidad de espacio especificado por `refreservation`. La reserva de `refreservation` se representa mediante el espacio en el disco utilizado del conjunto de datos principal, y repercute en las reservas y cuotas del conjunto de datos principal. Si se establece `refreservation`, sólo se permite una instantánea en caso de que, fuera de esta reserva, exista espacio libre en la agrupación para alojar la cantidad actual de bytes a los que se hace *referencia* en el conjunto de datos. La abreviatura de la propiedad es `refreserv`.
`reservation`	Número (o none)	`none`	Establece la cantidad de espacio mínimo en el disco garantizada para un sistema de archivos y sus descendientes. Cuando la cantidad de espacio utilizado aparece bajo este valor, se considera que el sistema de archivos utiliza la cantidad de espacio especificado por su reserva. Las reservas se registran en el espacio de disco del sistema de archivos principal utilizado y repercuten en las reservas y las cuotas del sistema de archivos principal. La abreviatura de la propiedad es `reserv`. Para obtener más información, consulte "Establecimiento de reservas en sistemas de archivos ZFS" en la página 204.
`rstchown`	Booleano	activado	Indica si el propietario del sistema de archivos puede otorgar cambios de propiedad de archivos. El valor predeterminado es restringir las operaciones de `chown`. Cuando `rstchown` se establece en off, el usuario tiene el privilegio `PRIV_FILE_CHOWN_SELF` para las operaciones `chown`.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`secondarycache`	Cadena	`all`	Controla la información que se almacena en la memoria caché secundaria (L2ARC). Los valores posibles son `all`, `none` y `metadata`. Si se establece en `all`, los datos de usuario y los metadatos se almacenan en la memoria caché. Si se establece en `none`, no se completan datos de usuario ni los metadatos se almacenan en la memoria caché. Si se establece en `metadata`, sólo los metadatos se almacenan en la memoria caché.
`setuid`	Booleano	`on`	Controla si el bit de `setuid` se cumple en un sistema de archivos.
`shadow`	Cadena	`None`	Identifica un sistema de archivos de ZFS como *shadow* del sistema de archivos descrito por el *URI*. Los datos se migran a un sistema de archivos shadow con esta propiedad establecida desde el sistema de archivos identificado por el URI. El sistema de archivos que se migrará deben ser de sólo lectura para una realizar una migración completa.
`share.nfs`	Cadena	`off`	Controla si se crea y se publica un recurso compartido NFS de un sistema de archivos ZFS y las opciones que se utilizan. También puede publicar y anular la publicación de un recurso compartido NFS utilizando los comandos `zfs share` y `zfs unshare`. El uso del comando `zfs share` para publicar un recurso compartido NFS requiere la configuración de una propiedad de recurso compartido NFS. Para obtener información sobre la configuración de propiedades de recursos compartidos NFS, consulte "Cómo compartir y anular la compartición de sistemas de archivos ZFS" en la página 188. Para obtener más información sobre cómo compartir los sistemas de archivos ZFS, consulte "Cómo compartir y anular la compartición de sistemas de archivos ZFS" en la página 188.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`share.smb`	Cadena	`off`	Controla si se crea y se publica un recurso compartido SMB de un sistema de archivos ZFS y las opciones que se utilizan. También puede publicar y anular la publicación de un recurso compartido SMB utilizando los comandos `zfs share` y `zfs unshare`. El uso del comando `zfs share` para publicar un recurso compartido SMB requiere la configuración de una propiedad de recurso compartido SMB. Para obtener información sobre la configuración de propiedades de recursos compartidos SMB, consulte "Cómo compartir y anular la compartición de sistemas de archivos ZFS" en la página 188.
`snapdir`	Cadena	`hidden`	Controla si el directorio `.zfs` está oculto o visible en el directorio raíz del sistema de archivos. Para obtener más información sobre el uso de instantáneas, consulte "Información general de instantáneas de ZFS" en la página 219.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`sync`	Cadena	`standard`	Determina el comportamiento síncrono de las transacciones de un sistema de archivos. Los valores posibles son: ■ El valor predeterminado `standard` implica transacciones síncronas del sistema de archivos, como `fsync`, `O_DSYNC`, `O_SYNC`, etc., que se escriben en el registro de intentos. ■ `always` garantiza que *todas* las transacciones del sistema de archivos se hayan escrito y vaciado en una ubicación de almacenamiento estable mediante la devolución de una llamada del sistema. Este valor tiene un penalización de rendimiento significativa. ■ `disabled` significa que las solicitudes síncronas están desactivadas. Las transacciones del sistema de archivos solamente se ejecutan en una ubicación de almacenamiento estable en la ejecución del siguiente grupo de transacciones, que puede ser después de varios segundos. Este valor ofrece el mejor rendimiento, sin riesgo de dañar a la agrupación. **Precaución –** Este valor `disabled` es muy peligroso debido a que ZFS ignora las demandas de aplicaciones de la transacción síncrona, como de bases de datos u operaciones NFS. Al definir este valor en la raíz activa actual o en el sistema de archivos `/var`, se puede producir un comportamiento inesperado, una pérdida de datos de la aplicación o un aumento de la vulnerabilidad a ataques de reproducción. Sólo debe utilizar este valor si comprende todos los riesgos asociados.
`type`	Cadena	N/A	Propiedad de sólo lectura que identifica el tipo de conjunto de datos como `filesystem` (sistema de archivos o clónico), `volume` o `snapshot`.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`used`	Número	N/A	Propiedad de sólo lectura que identifica la cantidad de espacio que consumen el conjunto de datos y todos sus descendientes. Para obtener información detallada, consulte "Propiedad `used`" en la página 167.
`usedbychildren`	Número	`off`	Propiedad de sólo lectura que identifica la cantidad de espacio en el disco utilizado por subordinados de este conjunto de datos, que se liberaría si todos los subordinados del conjunto de datos se destruyeran. La abreviatura de la propiedad es `usedchild`.
`usedbydataset`	Número	`off`	Propiedad de sólo lectura que identifica la cantidad de espacio en el disco que utiliza este conjunto de datos en sí, que se liberaría si se destruyera el conjunto de datos, después de eliminar primero las instantáneas y los `refreservation`. La abreviatura de la propiedad es `usedds`.
`usedbyrefreservation`	Número	`off`	Propiedad de sólo lectura que identifica la cantidad de espacio en el disco que utiliza un `refreservation` establecido en un conjunto de datos, que se liberaría si el `refreservation` se eliminara. La abreviatura de la propiedad es `usedrefreserv`.
`usedbysnapshots`	Número	`off`	Propiedad de sólo lectura que identifica la cantidad de espacio en el disco que consumen las instantáneas de este conjunto de datos. En concreto, es la cantidad de espacio en el disco que se liberaría si todas las instantáneas de este conjunto de instantáneas se destruyeran. Tenga en cuenta que esto no es simplemente la suma de las propiedades `used` de las instantáneas, ya que varias instantáneas pueden compartir el espacio. La abreviatura de la propiedad es `usedsnap`.
`version`	Número	N/A	Identifica la versión de disco de un sistema de archivos, que es independiente de la versión de la agrupación. Esta propiedad sólo se puede establecer en una versión posterior que está disponible en la versión del software admitida. Para obtener más información, consulte el comando `zfs upgrade`.

TABLA 5–1 Descripciones de propiedades nativas de ZFS *(Continuación)*

Nombre de propiedad	Tipo	Valor predeterminado	Descripción
`utf8only`	Booleano	`Off`	Esta propiedad indica si un sistema de archivos debe rechazar nombres de archivos que incluyen caracteres que no están presentes en el conjunto de códigos de caracteres UTF-8. Si esta propiedad se establece de forma explícita en `off`, la propiedad `normalization` no se debe establecer de forma explícita o debe estar definida en `none`. El valor predeterminado de la propiedad `utf8only` es `off`. Esta propiedad no se podrá cambiar una vez creado el sistema de archivos.
`volsize`	Número	N/A	En el caso de volúmenes, especifica el tamaño lógico del volumen. Para obtener información detallada, consulte "Propiedad `volsize`" en la página 174.
`volblocksize`	Número	`8 KB`	En volúmenes, especifica el tamaño del bloque del volumen. El tamaño del bloque no se puede cambiar cuando el volumen se ha escrito, por lo que debe establecer el tamaño del bloque en el momento de la creación del volumen. El tamaño de bloque predeterminado para volúmenes es de 8 Kbytes. Es válida cualquier potencia de 2 desde 512 bytes hasta 128 Kbytes. La abreviatura de la propiedad es `volblock`.
`vscan`	Booleano	`Off`	Controla si los archivos normales deben ser analizados en busca de virus al abrirse y al cerrarse. Además de activar esta propiedad, también se debe activar un servicio de análisis de virus para que se realice un análisis si cuenta con un software de análisis de virus de terceros. El valor predeterminado es `off`.
`zoned`	Booleano	N/A	Indica si este sistema de archivos se ha agregado a una zona no global. Si se establece esta propiedad, el punto de montaje no recibe permisos en la zona global y ZFS no puede montar dicho sistema de archivos cuando se solicite. Cuando una zona se instala por primera vez, esta propiedad se establece para cualquier sistema de archivos agregado. Para obtener más información sobre el uso de ZFS con zonas instaladas, consulte "Uso de ZFS en un sistema Solaris con zonas instaladas" en la página 288.
`xattr`	Booleano	`on`	Indica si los atributos extendidos se activan (`on`) o desactivan (`off`) para este sistema de archivos.

Propiedades nativas de sólo lectura de ZFS

Las propiedades nativas de sólo lectura se pueden recuperar, pero no definir. Las propiedades nativas de sólo lectura no se heredan. Algunas propiedades nativas son específicas de un tipo concreto de conjunto de datos. En estos casos, el tipo de conjunto de datos concreto se menciona en la descripción de la Tabla 5–1.

Las propiedades nativas de sólo lectura se enumeran aquí y se describen en la Tabla 5–1.

- `available`
- `compressratio`
- `creation`
- `keystatus`
- `mounted`
- `origin`
- `referenced`
- `rekeydate`
- `type`
- `used`

  Para obtener más información, consulte "Propiedad `used`" en la página 167.
- `usedbychildren`
- `usedbydataset`
- `usedbyrefreservation`
- `usedbysnapshots`

Para obtener más información sobre el cálculo de espacio en el disco, incluidas las propiedades `used`, `referenced` y `available`, consulte "Cálculo del espacio de ZFS" en la página 34.

Propiedad `used`

La propiedad `used` es una propiedad de sólo lectura que identifica la cantidad de espacio en el disco que consume este conjunto de datos y todos sus descendientes. Este valor se comprueba con la cuota del conjunto de datos y la reserva. El espacio utilizado no incluye la reserva del conjunto de datos, pero considera la reserva de cualquier conjunto de datos descendiente. La cantidad de espacio que un conjunto de datos consume en su elemento principal, y la cantidad de espacio en el disco que se libera si el conjunto de datos se destruye repetidamente, es la mayor entre su espacio utilizado y su reserva.

Cuando se crean instantáneas, su espacio en el disco se comparte inicialmente entre la instantánea y el sistema de archivos, y posiblemente con instantáneas anteriores. Conforme cambia el sistema de archivos, el espacio en el disco que se compartía anteriormente se vuelve

exclusivo para la instantánea, y se cuenta en el espacio utilizado de la instantánea. El espacio que utiliza una instantánea representa sólo sus datos exclusivos. Asimismo, suprimir instantáneas puede aumentar la cantidad de espacio en el disco exclusivo para (y utilizado por) otras instantáneas. Para obtener información sobre los problemas de espacio y las instantáneas, consulte "Comportamiento de falta de espacio" en la página 35.

La cantidad de espacio en el disco utilizado, disponible y con referencia no incluye los cambios pendientes. Los cambios pendientes suelen calcularse en pocos segundos. Si se confirma un cambio en un disco mediante la función `fsync(3c)` u `O_SYNC`, no se garantiza necesariamente que la información de uso del espacio en el disco se actualice de inmediato.

La información de las propiedades `usedbychildren`, `usedbydataset` , `usedbyrefreservation` y `usedbysnapshots` se puede mostrar mediante el comando `zfs list -o space`. Estas propiedades identifican la propiedad `used` en espacio en el disco que consumen los descendientes. Para obtener más información, consulte la Tabla 5–1.

Propiedades nativas de ZFS configurables

Las propiedades nativas configurables son aquellas cuyos valores se pueden recuperar y establecer. Las propiedades nativas configurables se establecen mediante el comando `zfs set`, como se describe en "Configuración de propiedades de ZFS" en la página 178, o mediante el comando `zfs create`, como se describe en "Creación de un sistema de archivos ZFS" en la página 148. Salvo las cuotas y las reservas, las propiedades nativas configurables se heredan. Si desea más información sobre cuotas y reservas, consulte "Configuración de cuotas y reservas de ZFS" en la página 199.

Algunas propiedades nativas configurables son específicas de un tipo concreto de conjunto de datos. En estos casos, el tipo de conjunto de datos concreto se menciona en la descripción de la Tabla 5–1. Si no se menciona específicamente, una propiedad se aplica a todos los tipos de conjuntos de datos: sistemas de archivos, clones, volúmenes e instantáneas.

Las propiedades configurables aparecen aquí y se describen en la Tabla 5–1.

- `aclinherit`

  Para obtener una descripción detallada, consulte "Propiedades de ACL" en la página 248.
- `atime`
- `canmount`
- `casesensitivity`
- `suma de comprobación`
- `compression`
- `copies`
- `devices`

- dedup
- cifrado
- exec
- keysource
- logbias
- mlslabel
- mountpoint
- nbmand
- normalization
- primarycache
- quota
- readonly
- recordsize

  Para obtener información detallada, consulte "Propiedad recordsize" en la página 173.
- refquota
- refreservation
- reservation
- rstchown
- secondarycache
- share.smb
- share.nfs
- setuid
- snapdir
- version
- vscan
- utf8only
- volsize

  Para obtener información detallada, consulte "Propiedad volsize" en la página 174.
- volblocksize
- zoned
- xattr

Propiedad canmount

Si esta propiedad se establece en off, el sistema de archivos no se puede montar mediante los comandos zfs mount ni zfs mount -a. Establecer esta propiedad en off es como establecer la propiedad mountpoint en none, excepto que el sistema de archivos todavía tiene una propiedad mountpoint normal que se puede heredar. Por ejemplo, puede establecer esta propiedad en off, así como establecer propiedades heredables para los sistemas de archivos descendientes. Sin embargo, el sistema de archivos principal no se puede montar nunca, ni los usuarios pueden acceder a él. En este caso, el sistema de archivos principal sirve como *contenedor* para poder establecer propiedades en el contenedor, pero nunca se puede acceder al contenedor en sí.

En el ejemplo siguiente, se crea userpool y su propiedad canmount se establece en off. Los puntos de montaje para los sistemas de archivos de usuario descendientes se establecen en un punto de montaje común, /export/home. Los sistemas de archivo descendientes heredan las propiedades que se establecen en el sistema de archivos superior, pero el sistema de archivos superior no se monta nunca.

```
# zpool create userpool mirror c0t5d0 c1t6d0
# zfs set canmount=off userpool
# zfs set mountpoint=/export/home userpool
# zfs set compression=on userpool
# zfs create userpool/user1
# zfs create userpool/user2
# zfs mount
userpool/user1                  /export/home/user1
userpool/user2                  /export/home/user2
```

Si la propiedad canmount se establece en noauto, el sistema de archivos sólo se puede montar de manera explícita, no automáticamente.

Propiedad casesensitivity

Esta propiedad indica si el algoritmo que coincide con el nombre de archivo utilizado por el sistema de archivos debe ser casesensitive o caseinsensitive, o si debe permitir una combinación de ambos estilos de coincidencia (mixed).

Cuando una solicitud de coincidencia que no distingue mayúsculas de minúsculas está compuesta por un sistema de archivos de sensibilidad *mixta*, el comportamiento es generalmente el mismo que se espera de un sistema de archivos que no distingue mayúsculas de minúsculas. La diferencia es que un sistema de archivos de sensibilidad mixta puede contener directorios con varios nombres que son únicos desde una perspectiva de distinción entre mayúsculas y minúsculas, pero no desde una perspectiva de no distinción entre mayúsculas y minúsculas.

Por ejemplo, un directorio puede contener archivos foo, Foo y FOO. Si se realiza una solicitud de coincidencia sin distinción entre mayúsculas y minúsculas con cualquiera de las posibles formas de foo, (por ejemplo foo, FOO, FoO, fOo, etc.), uno de los tres archivos existentes se elige

como coincidencia en función del algoritmo de coincidencia. No se garantiza exactamente qué archivo selecciona el algoritmo como coincidencia, pero lo que sí se garantiza es que el mismo archivo se selecciona como una coincidencia de cualquiera de las formas de `foo`. El archivo elegido como coincidencia sin distinción entre mayúsculas y minúsculas para `foo`, `FOO` , `foO`, `Foo`, etc., es siempre el mismo, siempre que el directorio permanezca sin cambios.

La propiedades `utf8only`, `normalization` y `casesensitivity` también proporcionan nuevos permisos que se pueden asignar a usuarios sin privilegios mediante la administración delegada de ZFS. Para obtener más información, consulte "Delegación de permisos de ZFS" en la página 272.

Propiedad `copies`

Como función de fiabilidad, los metadatos de sistemas de archivos ZFS se almacenan automáticamente varias veces en distintos discos, si es posible. Esta función se conoce como *bloques ditto*.

En esta versión también se pueden almacenar varias copias de los datos de usuario por sistema de archivos utilizando el comando `zfs set copies`. Por ejemplo:

```
# zfs set copies=2 users/home
# zfs get copies users/home
NAME        PROPERTY  VALUE       SOURCE
users/home  copies    2           local
```

Los valores disponibles son 1, 2 ó 3. El valor predeterminado es 1. Estas copias son adicionales a cualquier redundancia de nivel de grupo, por ejemplo en una configuración RAID-Z o reflejada.

Las ventajas de almacenar varias copias de los datos de usuario ZFS son:

- Mejora la retención de datos al permitir la recuperación de fallos de lectura de bloques irrecuperables, como los fallos de medios (conocidos como *bit rot) para todas las configuraciones ZFS.*
- Proporciona protección de datos, incluso cuando sólo hay disponible un disco.
- Permite seleccionar las directivas de protección de datos por sistema de archivos, más allá de las posibilidades de la agrupación de almacenamiento.

Nota – Según la asignación de los bloques ditto en la agrupación de almacenamiento, varias copias se podrían colocar en un solo disco. Un posible fallo posterior en el disco podría hacer que todos los bloques ditto no estuvieran disponibles.

Los bloques ditto pueden ser útiles cuando de forma involuntaria se crea una agrupación no redundante y se deben establecer políticas de retención de datos.

Propiedad dedup

La propiedad dedup controla si los datos duplicados se eliminan de un sistema de archivos. Si un sistema de archivos tiene activada la propiedad dedup, los bloques de datos duplicados se eliminan de forma sincrónica. El resultado es que se almacenan solamente los datos exclusivos y los componentes comunes se comparten entre archivos.

No active la propiedad dedup de los sistemas de archivos que residen en sistemas de producción hasta que se revisen las siguientes consideraciones:

1. Determine si los datos se beneficiarían con el ahorro de espacio que proporciona la anulación de la duplicación. Si los datos duplicados no se pueden eliminar, no tiene sentido activar la eliminación de datos duplicados. Por ejemplo:

```
# zdb -S tank
Simulated DDT histogram:
bucket              allocated                       referenced
______   ______________________________   ______________________________
refcnt   blocks   LSIZE   PSIZE   DSIZE   blocks   LSIZE   PSIZE   DSIZE
------   ------   -----   -----   -----   ------   -----   -----   -----
     1    2.27M    239G    188G    194G    2.27M    239G    188G    194G
     2     327K   34.3G   27.8G   28.1G     698K   73.3G   59.2G   59.9G
     4    30.1K   2.91G   2.10G   2.11G     152K   14.9G   10.6G   10.6G
     8    7.73K    691M    529M    529M    74.5K   6.25G   4.79G   4.80G
    16      673   43.7M   25.8M   25.9M    13.1K    822M    492M    494M
    32      197   12.3M   7.02M   7.03M    7.66K    480M    269M    270M
    64       47   1.27M    626K    626K    3.86K    103M   51.2M   51.2M
   128       22    908K    250K    251K    3.71K    150M   40.3M   40.3M
   256        7    302K     48K   53.7K    2.27K   88.6M   17.3M   19.5M
   512        4    131K   7.50K   7.75K    2.74K    102M   5.62M   5.79M
    2K        1      2K      2K      2K    3.23K   6.47M   6.47M   6.47M
    8K        1    128K      5K      5K    13.9K   1.74G   69.5M   69.5M
 Total    2.63M    277G    218G    225G    3.22M    337G    263G    270G

dedup = 1.20, compress = 1.28, copies = 1.03, dedup * compress / copies = 1.50
```

   Si la razón estimada de dedup es mayor que 2, puede que se produzca un ahorro de espacio con dedup.

   En el ejemplo anterior, la relación de eliminación de datos duplicados es menor que 2; por lo tanto, no se recomienda activar la eliminación de datos duplicados.

2. Asegúrese de que el sistema tenga memoria suficiente para admitir dedup.
   - Cada entrada de la tabla de dedup incorporada en el núcleo central es de aproximadamente 320 bytes.
   - Multiplique el número de bloques asignados por 320. Por ejemplo:

     ```
     in-core DDT size = 2.63M x 320 = 841.60M
     ```

3. El rendimiento de dedup es mejor cuando la tabla de anulación de la duplicación se ajusta a la memoria. Si la tabla de dedup se tiene que escribir en el disco, el rendimiento disminuirá. Por ejemplo, la eliminación de un sistema de archivos de gran tamaño con la eliminación de datos duplicados activada disminuirá significativamente el rendimiento del sistema si el sistema no cumple con los requisitos de memoria descritos anteriormente.

Cuando dedup está activado, el algoritmo de suma de comprobación dedup sustituye la propiedad checksum. Establecer el valor de la propiedad en verify es lo mismo que especificar sha256,verify. Si la propiedad se define en verify y dos bloques tienen la misma firma, ZFS realiza una comparación por bytes con el bloque existente para asegurarse de que los contenidos son idénticos.

Esta propiedad se puede activar por sistema de archivos. Por ejemplo:

```
# zfs set dedup=on tank/home
```

Puede utilizar el comando zfs get para determinar si se ha establecido la propiedad dedup.

Aunque la eliminación de datos duplicados se establece como una propiedad del sistema de archivos, el alcance se extiende a todas las agrupaciones. Por ejemplo, se puede identificar la relación de eliminación de datos duplicados. Por ejemplo:

```
# zpool list tank
NAME    SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
rpool   136G  55.2G  80.8G    40%  2.30x  ONLINE  -
```

La columna DEDUP indica cuántos datos duplicados se han eliminado. Si la propiedad dedup no está activada en un sistema de archivos, o si la propiedad dedup fue activada en el sistema de archivos en ese momento, la relación de DEDUP es 1.00x.

Puede utilizar el comando zpool get para determinar el valor de la propiedad dedupratio. Por ejemplo:

```
# zpool get dedupratio export
NAME   PROPERTY    VALUE  SOURCE
rpool  dedupratio  3.00x  -
```

Esta propiedad de agrupación ilustra la cantidad de datos duplicados que ha eliminado esta agrupación.

Propiedad encryption

Puede utilizar la propiedad de cifrado para cifrar sistemas de archivos ZFS. Para obtener más información, consulte “Cifrado de sistemas de archivos ZFS” en la página 205.

Propiedad recordsize

La propiedad recordsize especifica un tamaño de bloque sugerido para los archivos del sistema de archivos.

Esta propiedad se designa exclusivamente para utilizarse con cargas de trabajo de la base de datos que acceden a los archivos en registros de tamaño fijo. ZFS ajusta automáticamente el tamaño de los bloques de acuerdo con algoritmos internos optimizados para los patrones de acceso habituales. En cuanto a las bases de datos que crean archivos muy grandes pero que

acceden a los archivos en pequeños bloques aleatorios, estos algoritmos quizá funcionen por debajo de su nivel habitual. Si se especifica un valor de `recordsize` mayor o igual que el tamaño de grabación de la base de datos, el rendimiento puede mejorar considerablemente. El uso de esta propiedad se desaconseja de manera especial en los sistemas de archivos de finalidad general; puede afectar negativamente al rendimiento. El tamaño especificado debe ser una potencia de 2 mayor o igual que 512 y menor o igual que 128 KB. El cambio del valor `recordsize` en los sistemas de archivos sólo afecta a los archivos creados posteriormente. No afecta a los archivos ya creados.

La abreviatura de la propiedad es `recsize`.

La propiedad share.smb

Esta propiedad permite compartir sistemas de archivos ZFS con el servicio Oracle Solaris SMB, e identifica las opciones que se pueden utilizar.

Cuando la propiedad cambia de off a on, los recursos compartidos que heredan la propiedad se vuelven a compartir con sus opciones actuales. Cuando la propiedad se establece en off, los recursos compartidos que heredan la propiedad no se comparten.Para ver ejemplos de uso de la propiedad `share.smb`, consulte "Cómo compartir y anular la compartición de sistemas de archivos ZFS" en la página 188.

Propiedad `volsize`

La propiedad `volsize` especifica el tamaño lógico del volumen. De forma predeterminada, la creación de un volumen establece una reserva para la misma cantidad. Cualquier cambio en `volsize` se refleja en un cambio equivalente en la reserva. Estas comprobaciones se utilizan para evitar un comportamiento inesperado para los usuarios. Un volumen que contenga menos espacio del que indica como disponible puede provocar un comportamiento indefinido o corrupción en los datos, según cómo se utilice el volumen. Estos efectos también pueden darse si el tamaño del volumen se cambia durante su uso, especialmente si se reduce el tamaño. Al ajustar el tamaño del volumen se debe ir con sumo cuidado.

Aunque no se recomienda, puede crear un volumen disperso si especifica el indicador `-s` en el comando `zfs create -V` o si cambia la reserva después de crear el volumen. Un *volumen disperso* se define como un volumen donde la reserva no es igual al tamaño del volumen. En un volumen disperso, los cambios en `volsize` no se reflejan en la reserva.

Para obtener más información sobre el uso de volúmenes, consulte "Volúmenes de ZFS" en la página 285.

Propiedades de usuario de ZFS

Además de las propiedades nativas, ZFS es compatible con las propiedades aleatorias del usuario. Las propiedades del usuario no repercuten en el comportamiento del sistema de archivos ZFS, pero puede usarlas para anotar información de manera que tenga sentido en su entorno.

Los nombres de propiedad del usuario deben ajustarse a las características siguientes:

- Deben contener un signo de dos puntos (': ') para distinguirlos de las propiedades nativas.
- Además, deben contener letras minúsculas, números o los signos de puntuación siguientes: ':', '+','.', '_'.
- La longitud máxima de un nombre de propiedad de usuario es 256 caracteres.

La convención habitual es que el nombre de la propiedad se divida en los dos componentes siguientes, pero este espacio de nombre no lo aplica ZFS:

module:property

Cuando haga un uso programático de las propiedades del usuario, utilice un nombre de dominio DNS inverso para el componente *módulo* de nombres de propiedades con vistas a reducir la posibilidad de que dos paquetes desarrollados independientemente utilicen el mismo nombre de propiedad para fines diferentes. Los nombres de propiedad que comienzan con `com.oracle.` se reservan para su uso por Oracle Corporation.

Los valores de las propiedades de usuario deben ajustarse a las convenciones siguientes:

- Deben constar de cadenas aleatorias que se heredan siempre y que nunca se validan.
- La longitud máxima de la propiedad de usuario es 1024 caracteres.

Por ejemplo:

```
# zfs set dept:users=finance userpool/user1
# zfs set dept:users=general userpool/user2
# zfs set dept:users=itops userpool/user3
```

Todos los comandos que se utilizan en propiedades, como `zfs list`, `zfs get`, `zfs set`, etc., se pueden utilizar para manipular las propiedades nativas y las del usuario.

Por ejemplo:

```
zfs get -r dept:users userpool
NAME            PROPERTY    VALUE           SOURCE
userpool        dept:users  all             local
userpool/user1  dept:users  finance         local
userpool/user2  dept:users  general         local
userpool/user3  dept:users  itops           local
```

Para borrar una propiedad de usuario, utilice el comando `zfs inherit.` Por ejemplo:

```
# zfs inherit -r dept:users userpool
```

Si la propiedad no se define en ningún conjunto de datos superior, se elimina por completo.

Consulta de información del sistema de archivos ZFS

El comando `zfs list` ofrece un mecanismo ampliable para ver y consultar información del conjunto de datos. En esta sección se explican las consultas básicas y complejas.

Visualización de información básica de ZFS

Puede visualizar información básica del conjunto de datos mediante el comando `zfs list` sin opciones. Este comando muestra los nombres de todos los conjuntos de datos en el sistema y los de sus propiedades `used`, `available`, `referenced` y `mountpoint`. Para obtener más información sobre estas propiedades, consulte "Introducción a las propiedades de ZFS" en la página 151.

Por ejemplo:

```
# zfs list
users                   2.00G  64.9G    32K  /users
users/home              2.00G  64.9G    35K  /users/home
users/home/cindy         548K  64.9G   548K  /users/home/cindy
users/home/mark         1.00G  64.9G  1.00G  /users/home/mark
users/home/neil         1.00G  64.9G  1.00G  /users/home/neil
```

También puede utilizar este comando para visualizar conjuntos de datos específicos si proporciona el nombre del conjunto de datos en la línea de comandos. Asimismo, utilice la opción `-r` para mostrar repetidamente todos los descendientes del conjunto de datos. Por ejemplo:

```
# zfs list -t all -r users/home/mark
NAME                        USED  AVAIL  REFER  MOUNTPOINT
users/home/mark            1.00G  64.9G  1.00G  /users/home/mark
users/home/mark@yesterday      0      -  1.00G  -
users/home/mark@today          0      -  1.00G  -
```

Puede utilizar el comando `lista zfs` con el punto de montaje de un sistema de archivos. Por ejemplo:

```
# zfs list /user/home/mark
NAME              USED  AVAIL  REFER  MOUNTPOINT
users/home/mark  1.00G  64.9G  1.00G  /users/home/mark
```

El ejemplo siguiente muestra cómo visualizar información básica sobre `tank/home/gina` y todos sus sistemas de archivos descendientes:

```
# zfs list -r users/home/gina
NAME                         USED  AVAIL  REFER  MOUNTPOINT
users/home/gina              2.00G  62.9G    32K  /users/home/gina
```

```
users/home/gina/projects      2.00G  62.9G    33K  /users/home/gina/projects
users/home/gina/projects/fs1  1.00G  62.9G  1.00G  /users/home/gina/projects/fs1
users/home/gina/projects/fs2  1.00G  62.9G  1.00G  /users/home/gina/projects/fs2
```

Para obtener más información sobre el comando `zfs list`, consulte `zfs(1M)`.

Creación de consultas de ZFS complejas

La salida `zfs list` se puede personalizar mediante las opciones `-o`, `-t` y `-H`.

Puede personalizar la salida del valor de las propiedades mediante la opción `-o` y una lista separada por comas de las propiedades en cuestión. También puede proporcionar una propiedad del conjunto de datos como argumento válido. Para obtener una lista de todas las propiedades de conjuntos de datos compatibles, consulte "Introducción a las propiedades de ZFS" en la página 151. Además de las propiedades que se definen, la lista de la opción `-o` también puede contener el `name` literal para indicar que la salida debe incluir el nombre del conjunto de datos.

En el siguiente ejemplo, se utiliza `zfs list` para mostrar el nombre del conjunto de datos, junto con los valores de propiedad `share.nfs` y `mountpoint`.

```
# zfs list -r -o name,share.nfs,mountpoint users/home
NAME                          NFS       MOUNTPOINT
users/home                    on        /users/home
users/home/cindy              on        /users/home/cindy
users/home/gina               on        /users/home/gina
users/home/gina/projects      on        /users/home/gina/projects
users/home/gina/projects/fs1  on        /users/home/gina/projects/fs1
users/home/gina/projects/fs2  on        /users/home/gina/projects/fs2
users/home/mark               on        /users/home/mark
users/home/neil               on        /users/home/neil
```

Puede utilizar la opción `-t` para especificar los tipos de conjuntos de datos que se deben mostrar. Los tipos válidos se describen en la tabla siguiente.

TABLA 5–2 Tipos de objetos ZFS

Tipo	Descripción
`filesystem`	Sistemas de archivos y clones
`volumen`	Volúmenes
`share`	Recurso compartido del sistema de archivos
`instantánea`	Instantáneas

Las opciones `-t` toman una lista separada por comas de los tipos de conjuntos de datos que mostrar. El ejemplo siguiente utiliza las opciones `-t` y `-o` simultáneamente para mostrar el nombre y la propiedad `used` para todos los sistemas:

```
# zfs list -r -t filesystem -o name,used users/home
NAME                          USED
users/home                   4.00G
users/home/cindy              548K
users/home/gina              2.00G
users/home/gina/projects     2.00G
users/home/gina/projects/fs1  1.00G
users/home/gina/projects/fs2  1.00G
users/home/mark              1.00G
users/home/neil              1.00G
```

Puede utilizar la opción `-H` para omitir la cabecera `zfs list` de la salida que se ha generado. Con la opción `-H`, todos los espacios en blanco se sustituyen por el carácter de tabulación. Puede usar esta opción si necesita una salida analizable; por ejemplo, con las secuencias de comandos. El ejemplo siguiente muestra la salida generada a partir del uso del comando `zfs list` con la opción `-H`:

```
# zfs list -r -H -o name users/home
users/home
users/home/cindy
users/home/gina
users/home/gina/projects
users/home/gina/projects/fs1
users/home/gina/projects/fs2
users/home/mark
users/home/neil
```

Administración de propiedades de ZFS

Las propiedades del conjunto de datos se administran mediante los subcomandos `set`, `inherit` y `get` del comando `zfs`.

- "Configuración de propiedades de ZFS" en la página 178
- "Herencia de propiedades de ZFS" en la página 179
- "Consulta de las propiedades de ZFS" en la página 180

Configuración de propiedades de ZFS

Puede utilizar el comando `zfs set` para modificar cualquier propiedad configurable del conjunto de datos. También puede usar el comando `zfs create` para establecer las propiedades cuando se crea el conjunto de datos. Para obtener una lista de propiedades del conjunto de datos configurables, consulte "Propiedades nativas de ZFS configurables" en la página 168.

El comando `zfs set` toma una secuencia de propiedad/valor con el formato de *propiedad=valor* y un nombre de conjunto de datos. Sólo se puede establecer o modificar una propiedad durante cada invocación de `zfs set`.

El ejemplo siguiente establece la propiedad `atime` en `off` para `tank/home`.

```
# zfs set atime=off tank/home
```

Además, cualquier propiedad del sistema de archivos se puede establecer al crear el sistema. Por ejemplo:

```
# zfs create -o atime=off tank/home
```

Puede especificar valores numéricos de propiedades mediante el uso de los siguientes sufijos sencillos (en orden creciente de importancia): `BKMGTPEZ`. Cualquiera de estos sufijos puede ir seguido de una `b` opcional que indica los bytes, con la excepción del sufijo `B`, que ya indica los bytes. Las cuatro invocaciones siguientes de `zfs set` son expresiones numéricas equivalentes que indican que la propiedad `quota` se puede establecer en el valor de 20 GB en el sistema de archivos `users/home/mark`:

```
# zfs set quota=20G users/home/mark
# zfs set quota=20g users/home/mark
# zfs set quota=20GB users/home/mark
# zfs set quota=20gb users/home/mark
```

Si intenta definir una propiedad de un sistema de archivos que esté 100% lleno, aparece en pantalla un mensaje similar al siguiente:

```
# zfs set quota=20gb users/home/mark
cannot set property for '/users/home/mark': out of space
```

Los valores de propiedades no numéricas distinguen mayúsculas de minúsculas y deben estar en minúsculas, excepto `mountpoint`. Los valores de esta propiedad pueden tener caracteres en mayúscula y minúscula.

Para obtener más información sobre el comando `zfs set`, consulte `zfs(1M)`.

Herencia de propiedades de ZFS

Todas las propiedades configurables, con la excepción de cuotas y reservas, heredan el valor del sistema de archivos superior, a menos que en el descendiente se establezca explícitamente una cuota o reserva. Si ningún superior tiene un valor explícito establecido para una propiedad heredada, se usa el valor predeterminado para la propiedad. Puede utilizar el comando `zfs inherit` para eliminar un valor de propiedad y, de este modo, hacer que el valor se herede del elemento superior.

El ejemplo siguiente utiliza el comando `zfs set` para activar la compresión para el sistema de archivos `tank/home/jeff`. A continuación, `zfs inherit` se utiliza para desconfigurar la propiedad `compression`; de este modo, la propiedad hereda el valor predeterminado de `off`. Como ni `home` ni `tank` tienen la propiedad `compression` configurada localmente, se utiliza el valor predeterminado. Si ambos tienen activada la compresión, se utiliza el valor configurado en el superior más inmediato (`home` en este ejemplo).

```
# zfs set compression=on tank/home/jeff
# zfs get -r compression tank/home
NAME                  PROPERTY     VALUE     SOURCE
tank/home             compression  off       default
tank/home/eric        compression  off       default
tank/home/eric@today  compression  -         -
tank/home/jeff        compression  on        local
# zfs inherit compression tank/home/jeff
# zfs get -r compression tank/home
NAME                  PROPERTY     VALUE     SOURCE
tank/home             compression  off       default
tank/home/eric        compression  off       default
tank/home/eric@today  compression  -         -
tank/home/jeff        compression  off       default
```

El subcomando `inherit` se aplica de forma recursiva cuando se especifica la opción `-r`. En el ejemplo siguiente, el comando hace que el valor de la propiedad `compression:` sea heredado por `tank/home` y cualquier descendiente que pudiera haber:

```
# zfs inherit -r compression tank/home
```

Nota – Si se utiliza la opción `-r`, se borra la configuración actual de la propiedad en todos los sistemas de archivos descendientes.

Para obtener más información sobre el comando `zfs inherit`, consulte `zfs(1M)`.

Consulta de las propiedades de ZFS

La forma más sencilla de consultar los valores de las propiedades es mediante el comando `zfs list`. Para obtener más información, consulte "Visualización de información básica de ZFS" en la página 176. Sin embargo, en el caso de consultas y secuencias de comandos complejas, use el comando `zfs get` para proporcionar información detallada en un formato personalizado.

Puede utilizar el comando `zfs get` para recuperar cualquier propiedad del conjunto de datos. El ejemplo siguiente muestra la manera de recuperar un solo valor de propiedad en un conjunto de datos:

```
# zfs get checksum tank/ws
NAME            PROPERTY       VALUE                      SOURCE
tank/ws         checksum       on                         default
```

La cuarta columna, `SOURCE`, indica el origen de este valor de propiedad. La tabla siguiente define los posibles valores de origen.

TABLA 5–3 Valores posibles de SOURCE (`zfs get`)

Valor de origen	Descripción
`default`	Este valor de propiedad nunca se ha configurado explícitamente para este conjunto de datos ni sus superiores. En esta propiedad se utiliza el valor predeterminado.
`inherited from` *dataset-name*	El valor de esta propiedad se hereda del superior, tal como especifica *dataset-name*.
`local`	El valor de esta propiedad se ha configurado explícitamente para este conjunto de datos mediante `zfs set`.
`temporary`	El valor de esta propiedad se ha establecido mediante la opción `zfs mount -o` y sólo es válida durante el ciclo de vida del montaje. Para obtener más información sobre las propiedades de puntos de montaje temporales, consulte "Uso de propiedades de montaje temporales" en la página 187.
- (none)	Esta propiedad es de sólo lectura. Su valor lo ha generado ZFS.

Puede utilizar la palabra clave especial `all` para recuperar todos los valores de propiedades del conjunto de datos. Los ejemplos siguientes usan la palabra clave `all`:

```
# zfs get all tank/home
NAME       PROPERTY              VALUE                  SOURCE
tank/home  aclinherit            restricted             default
tank/home  aclmode               discard                default
tank/home  atime                 on                     default
tank/home  available             66.9G                  -
tank/home  canmount              on                     default
tank/home  casesensitivity       mixed                  -
tank/home  checksum              on                     default
tank/home  compression           off                    default
tank/home  compressratio         1.00x                  -
tank/home  copies                1                      default
tank/home  creation              Fri May 11 10:58 2012  -
tank/home  dedup                 off                    default
tank/home  devices               on                     default
tank/home  encryption            off                    -
tank/home  exec                  on                     default
tank/home  keysource             none                   default
tank/home  keystatus             none                   -
tank/home  logbias               latency                default
tank/home  mlslabel              none                   -
tank/home  mounted               yes                    -
tank/home  mountpoint            /tank/home             default
tank/home  multilevel            off                    -
tank/home  nbmand                off                    default
tank/home  normalization         none                   -
tank/home  primarycache          all                    default
tank/home  quota                 none                   default
tank/home  readonly              off                    default
tank/home  recordsize            128K                   default
```

```
tank/home  referenced           43K                  -
tank/home  refquota             none                 default
tank/home  refreservation       none                 default
tank/home  rekeydate            -                    default
tank/home  reservation          none                 default
tank/home  rstchown             on                   default
tank/home  secondarycache       all                  default
tank/home  setuid               on                   default
tank/home  shadow               none                 -
tank/home  share.*              ...                  local
tank/home  snapdir              hidden               default
tank/home  sync                 standard             default
tank/home  type                 filesystem           -
tank/home  used                 8.54M                -
tank/home  usedbychildren       8.49M                -
tank/home  usedbydataset        43K                  -
tank/home  usedbyrefreservation 0                    -
tank/home  usedbysnapshots      0                    -
tank/home  utf8only             off                  -
tank/home  version              6                    -
tank/home  vscan                off                  default
tank/home  xattr                on                   default
tank/home  zoned                off                  default
```

La opción `-s` de `zfs get` permite especificar, por tipo de origen, las propiedades que mostrar. Esta opción toma una lista separada por comas que indica los tipos de origen deseados. Sólo aparecen las propiedades con el tipo de origen especificado. Los tipos de origen válidos son `local`, `default`, `inherited`, `temporary` y `none`. El ejemplo siguiente muestra todas las propiedades que se han establecido localmente en `tank/ws`.

```
# zfs get -s local all tank/ws
NAME     PROPERTY              VALUE                  SOURCE
tank/ws  compression           on                     local
```

Cualquiera de las opciones anteriores se puede combinar con la opción `-r` para mostrar de forma recursiva las propiedades especificadas en todos los subordinados del sistema de archivos indicado. En el ejemplo siguiente, todas las propiedades temporales de todos los sistemas de archivos en `tank/home` aparecen recursivamente:

```
# zfs get -r -s temporary all tank/home
NAME             PROPERTY       VALUE                      SOURCE
tank/home          atime          off                      temporary
tank/home/jeff     atime          off                      temporary
tank/home/mark     quota          20G                      temporary
```

Puede consultar los valores de las propiedades mediante el comando `zfs get` sin especificar un sistema de archivos de destino, lo cual significa que el comando funciona en todas las agrupaciones o los sistemas de archivos. Por ejemplo:

```
# zfs get -s local all
tank/home               atime          off                  local
tank/home/jeff          atime          off                  local
tank/home/mark          quota          20G                  local
```

Para obtener más información sobre el comando `zfs get`, consulte `zfs(1M)`.

Consulta de propiedades de ZFS para secuencias de comandos

El comando `zfs get` admite las opciones `-H` y `-o`, diseñadas para secuencias de comandos. Puede utilizar la opción `-H` para omitir información de cabecera y sustituir un espacio en blanco con el carácter de tabulación. El espacio en blanco uniforme permite el fácil análisis de los datos. Puede utilizar la opción `-o` para personalizar la salida de los modos siguientes:

- El `name` literal se puede utilizar con una lista separada por comas de propiedades como se definen en la sección "Introducción a las propiedades de ZFS" en la página 151.
- Una lista separada por comas de los campos literales, `name`, `value`, `property` y `source`, que deben salir seguidos por un espacio y un argumento, que es una lista separada por comas de las propiedades.

El ejemplo siguiente muestra la forma de recuperar un valor simple mediante las opciones `-H` y `-o` de `zfs get`:

```
# zfs get -H -o value compression tank/home
on
```

La opción `-p` informa de valores numéricos como sus valores exactos. Por ejemplo, 1 MB se especifica como 1000000. Esta opción puede usarse de la forma siguiente:

```
# zfs get -H -o value -p used tank/home
182983742
```

Puede utilizar la opción `-r` junto con una de las opciones anteriores para recuperar de forma recursiva los valores solicitados para todos los descendientes. El ejemplo siguiente utiliza las opciones `-H`, `-o` y `-r` para recuperar el nombre del sistema de archivos y el valor de la propiedad `used` para `export/home` y sus descendientes, mientras se omite la salida del encabezado:

```
# zfs get -H -o name,value -r used export/home
```

Montaje de sistemas de archivos ZFS

En esta sección se describe cómo ZFS monta sistemas de archivos.

Administración de puntos de montaje de ZFS

De manera predeterminada, un sistema de archivos ZFS se monta automáticamente cuando se crea. Puede determinar un comportamiento de punto de montaje específico para un sistema de archivos, tal y como se describe en esta sección.

También puede establecer el punto de montaje predeterminado para el sistema de archivos de una agrupación en el momento de la creación mediante la opción `-m` del comando `zpool create`. Para obtener más información sobre la creación de agrupaciones, consulte "Creación de grupos de almacenamiento de ZFS" en la página 55.

De forma predeterminada, todos los sistemas de archivos ZFS se montan con ZFS en el inicio mediante el servicio `svc://system/filesystem/local` de la Utilidad de gestión de servicios (SMF). Los sistemas de archivos se montan en */ruta*, donde *ruta* corresponde al nombre del sistema de archivos.

Puede anular el punto de montaje predeterminado si utiliza el comando `zfs set` para establecer la propiedad `mountpoint` en una ruta específica. ZFS crea automáticamente el punto de montaje especificado, si es preciso, y monta de manera automática el sistema de archivos asociado.

Los sistemas de archivos ZFS se montan automáticamente en el momento del inicio sin necesidad de que el usuario edite el archivo `/etc/vfstab`.

La propiedad `mountpoint` se hereda. Por ejemplo, si `pool/home` tiene la propiedad `mountpoint` configurada en `/export/stuff`, entonces `pool/home/user` hereda `/export/stuff/user` para su propiedad `mountpoint`.

Para evitar que se monte un sistema de archivos, establezca la propiedad `mountpoint` en `none`. Además, la propiedad `canmount` se puede utilizar para controlar si se puede montar un sistema de archivos. Para obtener información sobre la propiedad `canmount`, consulte "Propiedad `canmount`" en la página 170.

Los sistemas de archivos también se administran a través de las interfaces de montaje heredadas utilizando `zfs establecido` para definir la propiedad `mountpoint` en `legacy`. De este modo, se impide que ZFS monte y administre automáticamente un sistema de archivos. En su lugar se deben utilizar las herramientas heredadas que incluyen los comandos `mount` y `umount`, así como el archivo `/etc/vfstab`. Para obtener más información sobre montajes heredados, consulte "Puntos de montaje heredados" en la página 185.

Puntos de montaje automáticos

- Cuando cambie la propiedad `mountpoint` de `legacy` o `none` a una ruta específica, ZFS monta automáticamente el sistema de archivos.
- Si ZFS administra el sistema de archivos pero éste se encuentra desmontado, y se cambia la propiedad `mountpoint`, el sistema de archivos permanece sin montar.

Cualquier sistema de archivos cuya propiedad `mountpoint` no es `legacy` es gestionado por ZFS. En el ejemplo siguiente se crea un sistema de archivos cuyo punto de montaje es administrado automáticamente por ZFS:

```
# zfs create pool/filesystem
# zfs get mountpoint pool/filesystem
NAME             PROPERTY      VALUE                     SOURCE
```

```
pool/filesystem  mountpoint    /pool/filesystem          default
# zfs get mounted pool/filesystem
NAME             PROPERTY      VALUE                     SOURCE
pool/filesystem  mounted       yes                       -
```

También puede configurar explícitamente la propiedad `mountpoint` tal como se muestra en el ejemplo siguiente:

```
# zfs set mountpoint=/mnt pool/filesystem
# zfs get mountpoint pool/filesystem
NAME             PROPERTY      VALUE                     SOURCE
pool/filesystem  mountpoint    /mnt                      local
# zfs get mounted pool/filesystem
NAME             PROPERTY      VALUE                     SOURCE
pool/filesystem  mounted       yes                       -
```

Si se cambia la propiedad `mountpoint`, el sistema de archivos se desmonta automáticamente del punto de montaje anterior y se vuelve a montar en el nuevo punto de montaje. Se crean directorios de punto de montaje según sea preciso. Si ZFS no puede desmontar un sistema de archivos porque está activo, se informa de un error y se debe forzar un desmontaje manual.

Puntos de montaje heredados

Puede administrar los sistemas de archivos ZFS con herramientas heredadas si la propiedad `mountpoint` se configura como `legacy`. Los sistemas de archivos heredados se deben administrar mediante los comandos `mount` y `umount`, así como el archivo `/etc/vfstab`. ZFS no monta automáticamente sistemas de archivos heredados en el inicio, y los comandos `mount` y `umount` de ZFS no funcionan en sistemas de archivos de este tipo. Los ejemplos siguientes muestran cómo configurar y gestionar un sistema de archivos ZFS en el modo de herencia:

```
# zfs set mountpoint=legacy tank/home/eric
# mount -F zfs tank/home/eschrock /mnt
```

Para montar automáticamente un sistema de archivos heredado en el inicio, debe agregar una entrada al archivo `/etc/vfstab`. El ejemplo siguiente muestra el aspecto que podría tener la entrada en el archivo `/etc/vfstab`:

```
#device         device        mount           FS      fsck    mount   mount
#to mount       to fsck       point           type    pass    at boot options
#

tank/home/eric  -        /mnt           zfs        -        yes        -
```

Las entradas `device to fsck` y `fsck pass` se establecen en `-` porque el comando `fsck` no es aplicable a los sistemas de archivos ZFS. Para obtener más información sobre integridad de datos de ZFS, consulte “Semántica transaccional” en la página 28.

Montaje de sistemas de archivos ZFS

ZFS monta automáticamente sistemas de archivos cuando éstos se crean o cuando el sistema inicia. El uso del comando `zfs mount` sólo es necesario cuando se deben cambiar las opciones de montaje, o explícitamente montar o desmontar sistemas de archivos.

El comando `zfs mount` sin argumentos muestra todos los sistemas de archivos montados administrados por ZFS. No se muestran los puntos de montaje administrados de herencia. Por ejemplo:

```
# zfs mount | grep tank/home
zfs mount | grep tank/home
tank/home                       /tank/home
tank/home/jeff                  /tank/home/jeff
```

Puede utilizar la opción `-a` para montar todos los sistemas de archivos ZFS administrados. Los sistemas de archivos administrados de herencia no están montados. Por ejemplo:

```
# zfs mount -a
```

De forma predeterminada, ZFS no permite el montaje en la parte superior de un directorio que no está vacío. Por ejemplo:

```
# zfs mount tank/home/lori
cannot mount 'tank/home/lori': filesystem already mounted
```

Los puntos de montaje heredados se deben administrar mediante las herramientas de herencia. Intentar usar herramientas de ZFS genera un error. Por ejemplo:

```
# zfs mount tank/home/bill
cannot mount 'tank/home/bill': legacy mountpoint
use mount(1M) to mount this filesystem
# mount -F zfs tank/home/billm
```

Cuando se monta un sistema de archivos, utiliza un conjunto de opciones de montaje basadas en los valores de propiedad asociados con el sistema de archivos. La correspondencia entre las propiedades y las opciones de montaje es la siguiente:

TABLA 5–4 Propiedades relacionadas con el montaje de ZFS y opciones de montaje

Propiedad	Opción de montaje
atime	Atime/noatime
devices	devices/nodevices
exec	exec/noexec
nbmand	Nbmand/nonbmand
readonly	ro/rw

TABLA 5–4 Propiedades relacionadas con el montaje de ZFS y opciones de montaje *(Continuación)*

Propiedad	Opción de montaje
setuid	setuid/nosetuid
xattr	Xattr/noaxttr

La opción de montaje `nosuid` es un alias de `nodevices, nosetuid`.

Puede utilizar las funciones de montaje reflejado NFSv4 que le ayudan a gestionar mejor los directorios de inicio ZFS montados en NFS.

Cuando se crean sistemas de archivos en el servidor NFS, el cliente NFS puede descubrir automáticamente estos sistemas de archivos recién creados en el montaje existente de un sistema de archivos superior.

Por ejemplo, si el servidor `neo` ya comparte el sistema de archivos `tank` y el cliente `zee` lo tiene montado, `/tank/baz` se hace visible automáticamente en el cliente después de crearlo en el servidor.

```
zee# mount neo:/tank /mnt
zee# ls /mnt
baa    bar

neo# zfs create tank/baz

zee% ls /mnt
baa    bar    baz
zee% ls /mnt/baz
file1    file2
```

Uso de propiedades de montaje temporales

Si alguna de las opciones anteriores se configura explícitamente mediante la opción `-o` con el comando `zfs mount`, el valor de propiedad asociado se anula de manera temporal. Estos valores de propiedades se indican como `temporary` mediante el comando `zfs get` y recuperan la configuración original cuando se desmonta el sistema de archivos. Si se cambia un valor de propiedad mientras se monta el sistema de archivos, el cambio surte efecto inmediatamente y se anula cualquier configuración temporal.

En el ejemplo siguiente, la opción de montaje de sólo lectura se configura temporalmente en el sistema de archivos `tank/home/neil`. Se supone que el sistema de archivos está desmontado.

```
# zfs mount -o ro users/home/neil
```

Para cambiar temporalmente una propiedad de un sistema de archivos que está montado, debe usar la opción especial `remount`. En el ejemplo siguiente, la propiedad `atime` se cambia temporalmente a `off` para un sistema de archivos que esté montado:

```
# zfs mount -o remount,noatime users/home/neil
NAME             PROPERTY  VALUE  SOURCE
users/home/neil  atime     off    temporary
# zfs get atime users/home/perrin
```

Para obtener más información sobre el comando `zfs mount`, consulte `zfs(1M)`.

Desmontaje de los sistemas de archivos ZFS

Los sistemas de archivos ZFS se pueden desmontar mediante el subcomando `zfs unmount.` El comando `unmount` puede considerar como argumentos el punto de montaje o el nombre del sistema de archivos.

En el ejemplo siguiente, el nombre del sistema de archivos desmonta un sistema de archivos:

```
# zfs unmount users/home/mark
```

En el ejemplo siguiente, el punto de montaje desmonta el sistema de archivos:

```
# zfs unmount /users/home/mark
```

El comando `unmount` falla si el sistema de archivos está ocupado. Para forzar el desmontaje de un sistema de archivos, puede usar la opción `-f`. Tenga cuidado al forzar el desmontaje de un sistema de archivos si su contenido está en uso. La aplicación se puede comportar de manera imprevista.

```
# zfs unmount tank/home/eric
cannot unmount '/tank/home/eric': Device busy
# zfs unmount -f tank/home/eric
```

Para ofrecer compatibilidad con versiones anteriores, el comando `umount` se puede usar para desmontar sistemas de archivos ZFS. Por ejemplo:

```
# umount /tank/home/bob
```

Para obtener más información sobre el comando `zfs umount`, consulte `zfs(1M)`.

Cómo compartir y anular la compartición de sistemas de archivos ZFS

Oracle Solaris 11.1 simplifica la administración de recursos compartidos ZFS al aprovechar la herencia de propiedades ZFS. La sintaxis del nuevo recurso compartido está activada en agrupaciones que ejecutan la versión de agrupación 34.

Se pueden definir varios recursos compartidos por sistema de archivos. Un nombre de recurso compartido identifica de forma única a cada recurso compartido. Puede definir las propiedades que se utilizan para compartir una ruta determinada en un sistema de archivos. De manera

predeterminada, todos los sistemas de archivos están sin compartir. En general, los servicios del servidor NFS no se inician hasta que se crea un recurso compartido. Si crea un recurso compartido válido, los servicios NFS se inician automáticamente. Si la propiedad `mountpoint` de un sistema de archivos ZFS se establece en antigua, el sistema de archivos únicamente se puede compartir utilizando el comando `share` antiguo.

- La propiedad `share.nfs` reemplaza a la propiedad `sharenfs` en versiones anteriores para definir y publicar un recurso compartido NFS.
- La propiedad `share.smb` reemplaza a la propiedad `sharesmb` en versiones anteriores para definir y publicar un recurso compartido SMB.
- La propiedad `sharenfs` y la propiedad `sharesmb` propiedad son alias de la propiedad `share.nfs` y la propiedad `share.smb`.
- El archivo `/etc/dfs/dfstab` ya no se utiliza para compartir sistemas de archivos durante el inicio. Al configurar estas propiedades, los sistemas de archivos se comparten automáticamente. SMF gestiona información de recursos compartidos ZFS o UFS de modo que los sistemas de archivos se compartan automáticamente cuando se reinicia el sistema. Esta función significa que todos los sistemas de archivos cuya propiedad `sharenfs` o`sharesmb` no está establecida en off se comparten durante el inicio.
- La interfaz `sharemgr` ya no está disponible. El comando `share` antiguo sigue estando disponible para crear un recurso compartido antiguo. Consulte los ejemplos que se proporcionan a continuación.
- El comando `share -a` es igual al comando `share -ap` anterior, de modo que el uso compartido de un sistema de archivos es persistente. La opción `share -p` ya no está disponible.

Por ejemplo, si desea compartir el sistema de archivos `tank/home`, utilice una sintaxis similar a la siguiente:

```
# zfs set share.nfs=on tank/home
```

También puede especificar valores de propiedad adicionales o modificar valores de propiedad existentes en recursos compartidos de sistemas de archivos existentes. Por ejemplo:

```
# zfs set share.nfs.nosuid=on tank/home/userA
```

En ejemplo anterior, donde la propiedad `share.nfs` está definida en el sistema de archivos `tank/home`, el valor de propiedad `share.nfs` es heredado por los sistemas de archivos descendentes. Por ejemplo:

```
# zfs create tank/home/userA
# zfs create tank/home/userB
```

Sintaxis del uso compartido de ZFS heredados

La sintaxis de Oracle Solaris 11 aún es compatible, de modo que puede compartir sistemas de archivos en dos pasos. Esta sintaxis se admite en todas las versiones de agrupaciones.

- En primer lugar, utilice el comando `zfs set share` para crear un recurso compartido NFS o SMB del sistema de archivos ZFS.

```
# zfs create rpool/fs1
# zfs set share=name=fs1,path=/rpool/fs1,prot=nfs rpool/fs1
name=fs1,path=/rpool/fs1,prot=nfs
```

- A continuación, establezca la propiedad `sharenfs` o `sharesmb` en on para publicar el recurso compartido. Por ejemplo:

```
# zfs set sharenfs=on rpool/fs1
# grep fs1 /etc/dfs/sharetab
/rpool/fs1      fs1     nfs     sec=sys,rw
```

Los recursos compartidos del sistema de archivos se pueden mostrar con el comando `zfs get share` antiguo.

```
# zfs get share rpool/fs1
NAME       PROPERTY  VALUE  SOURCE
rpool/fs1  share     name=fs1,path=/rpool/fs1,prot=nfs  local
```

Además, el comando `share` para compartir un sistema de archivos, similar a la sintaxis en Oracle Solaris 10, aún se admite para compartir cualquier directorio dentro de un sistema de archivos. Por ejemplo, para compartir un sistema de archivos ZFS:

```
# share -F nfs /tank/zfsfs
# grep zfsfs /etc/dfs/sharetab
/tank/zfsfs    tank_zfsfs     nfs     sec=sys,rw
```

La sintaxis anterior es idéntica a la que se usa para compartir un sistema de archivos UFS:

```
# share -F nfs /ufsfs
# grep ufsfs /etc/dfs/sharetab
/ufsfs          -               nfs     rw
/tank/zfsfs     tank_zfsfs      nfs     rw
```

Sintaxis de uso compartido de ZFS nuevo

El nuevo comando `zfs set` se utiliza para compartir y publicar un sistema de archivos ZFS a través de los protocolos NFS o SMB. También puede establecer la propiedad `share.nfs` o `share.smb` al crear el sistema de archivos.

Por ejemplo, el sistema de archivos `tank/sales` se crea y se comparte. Los permisos de uso compartido predeterminados son de sólo lectura para todos los usuarios. El sistema de archivos descendente `tank/sales/logs` también se comparte automáticamente porque la propiedad

share.nfs es heredada por sistemas de archivos descendentes y el sistema de archivos tank/sales/log se establece en acceso de sólo lectura.

```
# zfs create -o share.nfs=on tank/sales
# zfs create -o share.nfs.ro=\* tank/sales/logs
# zfs get -r share.nfs tank/sales
NAME             PROPERTY   VALUE  SOURCE
tank/sales       share.nfs  on     local
tank/sales%      share.nfs  on     inherited from tank/sales
tank/sales/log   share.nfs  on     inherited from tank/sales
tank/sales/log%  share.nfs  on     inherited from tank/sales
```

Puede proporcionar acceso root a un sistema específico para un sistema de archivos compartido de la siguiente manera:

```
# zfs set share.nfs=on tank/home/data
# zfs set share.nfs.sec.default.root=neo tank/home/data
```

Uso compartido de ZFS con herencia por propiedad

En las agrupaciones que se actualizaron a la última versión de agrupación 34, hay disponible una nueva sintaxis de uso compartido que utiliza la herencia de propiedades ZFS para facilitar el mantenimiento del uso compartido. Cada característica de uso compartido se convierte en una propiedad share independiente. Las propiedades share se identifican con nombres que empiezan con el prefijo share. Algunos ejemplos de propiedades share son share.desc, share.nfs.nosuid y share.smb.guestok.

La propiedad share.nfs controla si se activó el uso compartido de NFS. La propiedad share.smb controla si se activó el uso compartido de SMB. Los nombres de propiedad sharenfs y sharesmb antiguos se pueden seguir utilizando porque, en las nuevas agrupaciones, sharenfs es un alias de share.nfs y sharesmb es un alias de share.smb. Si desea compartir el sistema de archivos tank/home, utilice una sintaxis similar a la siguiente:

```
# zfs set share.nfs=on tank/home
```

En este ejemplo, el valor de propiedad share.nfs es heredado por cualquier sistema de archivos descendente. Por ejemplo:

```
# zfs create tank/home/userA
# zfs create tank/home/userB
# grep tank/home /etc/dfs/sharetab
/tank/home      tank_home       nfs     sec=sys,rw
/tank/home/userA        tank_home_userA nfs     sec=sys,rw
/tank/home/userB        tank_home_userB nfs     sec=sys,rw
```

Herencia de uso compartido de ZFS en agrupaciones más antiguas

En agrupaciones más antiguas, únicamente las propiedades sharenfs y sharesmb son heredadas por sistemas de archivos descendentes. Otras características de uso compartido se almacenan en el archivo .zfs/shares para cada recurso compartido y no se heredan.

Una regla especial es que siempre que se crea un nuevo sistema de archivos que hereda `sharenfs` o `sharesmb` de su elemento principal, se crea un recurso compartido predeterminado para ese sistema de archivos desde el valor `sharenfs` o `sharesmb`. Tenga en cuenta que cuando `sharenfs` simplemente está establecido en on, el recurso compartido predeterminado que se crea en un sistema de archivos descendente únicamente tiene las características NFS predeterminadas. Por ejemplo:

```
# zpool get version tank
NAME  PROPERTY  VALUE  SOURCE
tank  version   33     default
# zfs create -o sharenfs=on tank/home
# zfs create tank/home/userA
# grep tank/home /etc/dfs/sharetab
/tank/home      tank_home       nfs     sec=sys,rw
/tank/home/userA        tank_home_userA nfs     sec=sys,r
```

Recursos compartidos ZFS designados

También puede crear un recurso compartido *designado*, lo cual proporciona más flexibilidad para configurar permisos y propiedades en un entorno SMB. Por ejemplo:

```
# zfs share -o share.smb=on tank/workspace%myshare
```

En el ejemplo anterior, el comando `zfs share` crea un recurso compartido SMB denominado `myshare` del sistema de archivos `tank/workspace`. Puede acceder al recurso compartido SMB y mostrar o configurar permisos específicos o ACL mediante el directorio `.zfs/shares` del sistema de archivos. Cada recurso compartido SMB es representado mediante un archivo `.zfs/shares` independiente. Por ejemplo:

```
# ls -lv /tank/workspace/.zfs/shares
-rwxrwxrwx+  1 root     root           0 May 15 10:31 myshare
     0:everyone@:read_data/write_data/append_data/read_xattr/write_xattr
         /execute/delete_child/read_attributes/write_attributes/delete
         /read_acl/write_acl/write_owner/synchronize:allow
```

Los recursos compartidos designados heredan propiedades de uso compartido del sistema de archivos principal. Si agrega la propiedad `share.smb.guestok` al sistema de archivos principal en el ejemplo anterior, esta propiedad es heredada por el recurso compartido designado. Por ejemplo:

```
# zfs get -r share.smb.guestok tank/workspace
NAME                    PROPERTY           VALUE  SOURCE
tank/workspace          share.smb.guestok  on     inherited from tank
tank/workspace%myshare  share.smb.guestok  on     inherited from tank
```

Los recursos compartidos designados pueden ser útiles en el entorno NFS al definir recursos compartidos para un subdirectorio del sistema de archivos. Por ejemplo:

```
# zfs create -o share.nfs=on -o share.nfs.anon=99 -o share.auto=off tank/home
# mkdir /tank/home/userA
# mkdir /tank/home/userB
```

```
# zfs share -o share.path=/tank/home/userA tank/home%userA
# zfs share -o share.path=/tank/home/userB tank/home%userB
# grep tank/home /etc/dfs/sharetab
/tank/home/userA        userA   nfs     anon=99,sec=sys,rw
/tank/home/userB        userB   nfs     anon=99,sec=sys,rw
```

El ejemplo anterior también muestra que, al establecer `share.auto` en off para un sistema de archivos, se desactiva el uso compartido automático para ese sistema de archivos y se mantiene intacta la herencia de otras propiedades. A diferencia de la mayoría de las propiedades de uso compartido, la propiedad `share.auto` no se puede heredar.

Los recursos compartidos designados también se utilizan al crear un recurso compartido NFS público. Un recurso compartido público solamente se puede crear en un recurso compartido NFS designado. Por ejemplo:

```
# zfs create -o mountpoint=/pub tank/public
# zfs share -o share.nfs=on -o share.nfs.public=on tank/public%pubshare
# grep pub /etc/dfs/sharetab
/pub    pubshare        nfs     public,sec=sys,rw
```

Consulte `share_nfs(1M)` and `share_smb(1M)` para obtener una descripción detallada de las propiedades de recursos compartidos NFS y SMB.

Recursos compartidos ZFS automáticos

Cuando se crea un recurso compartido automático (auto), se crea un nombre de recurso único a partir del nombre del sistema de archivos. El nombre creado es una copia del nombre del sistema de archivos, con la excepción de que los caracteres del nombre del sistema de archivos, los cuales serían ilegales en el nombre del recurso, se reemplazan con caracteres de subrayado (_). Por ejemplo, el nombre del recurso de `data/home/john` es `data_home_john`.

La configuración de un nombre de propiedad `share.autoname` permite reemplazar el nombre del sistema de archivos con un nombre específico al crear el recurso compartido automático. El nombre también se utiliza para reemplazar el nombre del sistema de archivos de prefijo en caso de herencia. Por ejemplo:

```
# zfs create -o share.smb=on -o share.autoname=john data/home/john
# zfs create data/home/john/backups
# grep john /etc/dfs/sharetab
/data/home/john john    smb
/data/home/john/backups john_backups    smb
```

Si un comando `share` antiguo o el comando `zfs set share` se utilizan en un sistema de archivos que aún no fue compartido, el valor `share.auto` se establece automáticamente en `off`. Los comandos antiguos siempre crean recursos compartidos designados. Esta regla especial impide que el uso compartido automático interfiera con el recurso compartido designado que se crea.

Visualización de información de recurso compartido ZFS

Visualice el valor de las propiedades de uso compartido de archivos con el comando `zfs get`. En el siguiente ejemplo, se muestra cómo visualizar la propiedad `share.nfs` para un solo sistema de archivos:

```
# zfs get share.nfs tank/sales
NAME        PROPERTY   VALUE  SOURCE
tank/sales  share.nfs  on     local
```

En el siguiente ejemplo, se muestra cómo visualizar la propiedad `share.nfs` para sistemas de archivos descendentes:

```
# zfs get -r share.nfs tank/sales
NAME             PROPERTY   VALUE  SOURCE
tank/sales       share.nfs  on     local
tank/sales%      share.nfs  on     inherited from tank/sales
tank/sales/log   share.nfs  on     inherited from tank/sales
tank/sales/log%  share.nfs  on     inherited from tank/sales
```

La información extendida de la propiedad de recurso compartido no está disponible en la sintaxis del comando `zfs get all`.

Puede visualizar detalles específicos sobre la información del recurso compartido NFS o SMB con la siguiente sintaxis:

```
# zfs get share.nfs.all tank/sales
NAME        PROPERTY             VALUE  SOURCE
tank/sales  share.nfs.aclok      off    default
tank/sales  share.nfs.anon              default
tank/sales  share.nfs.charset.*  ...    default
tank/sales  share.nfs.cksum             default
tank/sales  share.nfs.index             default
tank/sales  share.nfs.log               default
tank/sales  share.nfs.noaclfab   off    default
tank/sales  share.nfs.nosub      off    default
tank/sales  share.nfs.nosuid     off    default
tank/sales  share.nfs.public     -      -
tank/sales  share.nfs.sec               default
tank/sales  share.nfs.sec.*      ...    default
```

Dado que existen muchas propiedades de recursos compartidos, considere la posibilidad de visualizar las propiedades con un valor que no es el predeterminado. Por ejemplo:

```
# zfs get -e -s local,received,inherited share.all tank/home
NAME       PROPERTY           VALUE  SOURCE
tank/home  share.auto         off    local
tank/home  share.nfs          on     local
tank/home  share.nfs.anon     99     local
tank/home  share.protocols    nfs    local
tank/home  share.smb.guestok  on     inherited from tank
```

Cambio de valores de propiedad de un recurso compartido ZFS

Puede cambiar los valores de propiedad de un recurso compartido especificando propiedades nuevas o modificadas en un recurso compartido de sistema de archivos. Por ejemplo, si la propiedad de sólo lectura se establece cuando se crea el sistema de archivos, se puede establecer en off.

```
# zfs create -o share.nfs.ro=\* tank/data
# zfs get share.nfs.ro tank/data
NAME       PROPERTY              VALUE  SOURCE
tank/data  share.nfs.sec.sys.ro  on     local
# zfs set share.nfs.ro=none tank/data
# zfs get share.nfs.ro tank/data
NAME       PROPERTY              VALUE  SOURCE
tank/data  share.nfs.sec.sys.ro  off    local
```

Si crea un recurso compartido SMB, también puede agregar el protocolo del recurso compartido NFS. Por ejemplo:

```
# zfs set share.smb=on tank/multifs
# zfs set share.nfs=on tank/multifs
# grep multifs /etc/dfs/sharetab
/tank/multifs   tank_multifs    nfs     sec=sys,rw
/tank/multifs   tank_multifs    smb     -
```

Elimine el protocolo SMB:

```
# zfs set share.smb=off tank/multifs
# grep multifs /etc/dfs/sharetab
/tank/multifs   tank_multifs    nfs     sec=sys,rw
```

Puede cambiar el nombre de un recurso compartido designado. Por ejemplo:

```
# zfs share -o share.smb=on tank/home/abc%abcshare
# grep abc /etc/dfs/sharetab
/tank/home/abc  abcshare        smb     -
# zfs rename tank/home/abc%abcshare tank/home/abc%a1share
# grep abc /etc/dfs/sharetab
/tank/home/abc  a1share        smb     -
```

Publicación y anulación de la publicación de recursos compartidos ZFS

Puede dejar de compartir temporalmente un recurso compartido designado sin destruirlo utilizando el comando `zfs unshare`. Por ejemplo:

```
# zfs unshare tank/home/abc%a1share
# grep abc /etc/dfs/sharetab
#
# zfs share tank/home/abc%a1share
# grep abc /etc/dfs/sharetab
/tank/home/abc  a1share smb     -
```

Cuando se emite el comando `zfs unshare`, se dejan de compartir todos los recursos compartidos del sistema de archivos. Estos recursos compartidos permanecen sin compartir hasta que se emite el comando `zfs share` para el sistema de archivos o se configura la propiedad `share.nfs` o `share.smb` para el sistema de archivos.

Los recursos compartidos definidos no se eliminan cuando se emite el comando `zfs unshare` y se vuelven a compartir la próxima vez que se emite el comando `zfs share` para el sistema de archivos o se configura la propiedad `share.nfs` o `share.smb` para el sistema de archivos.

Eliminación de un recurso compartido ZFS

Puede dejar de compartir un recurso compartido del sistema de archivos configurando la propiedad `share.nfs` o `share.smb` en off. Por ejemplo:

```
# zfs set share.nfs=off tank/multifs
# grep multifs /etc/dfs/sharetab
#
```

Puede eliminar permanentemente un recurso compartido designado utilizando el comando `zfs destroy`. Por ejemplo:

```
# zfs destroy tank/home/abc%a1share
```

Uso compartido de archivos ZFS en una zona no global

A partir de Oracle Solaris 11, puede crear y publicar recursos compartidos NFS en una zona no global de Oracle Solaris.

- Si un sistema de archivos ZFS está montado y se encuentra disponible en una zona no global, dicho sistema puede ser compartido en esa zona.
- Un sistema de archivos se puede compartir en la zona global si no está delegado a una zona no global o si no está montado en una zona no global. Si un sistema de archivos se agrega a una zona no global, solamente se puede compartir utilizando el comando `share` antiguo.

Por ejemplo, los sistemas de archivos `/export/home/data` y `/export/home/data1` están disponibles en `zfszone`.

```
zfszone# share -F nfs /export/home/data
zfszone#  cat /etc/dfs/sharetab

zfszone# zfs set share.nfs=on tank/zones/export/home/data1
zfszone# cat /etc/dfs/sharetab
```

Migración del uso compartido de ZFS y problemas de transición

Identifique los problemas de transición en esta sección.

- **Importación de sistemas de archivos con propiedades de uso compartido anteriores**: al importar una agrupación o recibir un flujo de sistema de archivos creado antes de Oracle Solaris 11, las propiedades `sharenfs` y `sharesmb` incluyen todas las propiedades del recurso compartido directamente en el valor de propiedad. En la mayoría de los casos, estas propiedades antiguas del recurso compartido se convierten a un conjunto equivalente de recursos compartidos designados en el momento en que se comparte cada sistema de archivos. Dado que las operaciones de importación originan el montaje y el uso compartido en la mayoría de los casos, la conversión a recursos compartidos designados se realiza directamente durante el proceso de importación.
- **Actualización de Oracle Solaris 11**: el primer uso compartido del sistema de archivos después de una actualización de la agrupación a la versión 34 puede tardar mucho tiempo, ya que los recursos compartidos designados se convierten al nuevo formato. Los recursos compartidos designados creados por el proceso de actualización son correctos, pero no pueden aprovechar la herencia de propiedades del recurso compartido.
  - Visualice los valores de propiedad del recurso compartido:

    ```
    # zfs get share.nfs filesystem
    # zfs get share.smb filesystem
    ```
  - Si vuelve a un entorno de inicio anterior, restablezca las propiedades `sharenfs` y `sharesmb` a sus valores originales.
- **Actualización de Oracle Solaris 11 Express**: en Oracle Solaris 11 y 11.1, las propiedades `sharenfs` y `sharesmb` únicamente pueden tener valores `off` y `on`. Estas propiedades ya no se utilizan para definir características de uso compartido.

  El archivo `/etc/dfs/dfstab` ya no se utiliza para compartir sistemas de archivos durante el inicio. Durante el inicio, todos los sistemas de archivos ZFS montados que incluyen recursos compartidos del sistema se comparten automáticamente. Un recurso compartido se activa cuando `sharenfs` o `sharesmb` se configuran en `on`.

  La interfaz `sharemgr` ya no está disponible. El comando `share` antiguo sigue estando disponible para crear un recurso compartido antiguo. El comando `share -a` es igual al comando `share -ap` anterior, de modo que el uso compartido de un sistema de archivos es persistente. La opción `share -p` ya no está disponible.
- **Actualización del sistema**: los recursos compartidos ZFS serán incorrectos si vuelve a iniciar un entorno de inicio Oracle Solaris 11 Express debido a cambios de propiedad en esta versión. Los recursos compartidos que no son de ZFS no se ven afectados. Si tiene previsto volver a iniciar un entorno de inicio anterior, primero debe guardar una copia de la configuración del recurso compartido existente previa a la operación `pkg update` a fin de poder restaurar la configuración del recurso compartido en los conjuntos de datos ZFS.

  En los entornos de inicio anteriores, utilice el comando `sharemgr show -vp` para enumerar todos los recursos compartidos y su configuración.

  Utilice los siguientes comandos para visualizar valores de propiedad de recursos compartidos:

```
# zfs get sharenfs filesystem
# zfs get sharesmb filesystem
```

Si vuelve a iniciar un entorno de inicio anterior, restablezca las propiedades `sharenfs` y `sharesmb`, y los recursos compartidos definidos con `sharemgr`, a sus valores originales.

- **Comportamiento de anulación de uso compartido antiguo**: el uso del comando `unshare -a` o el comando `unshareall` anula el uso compartido de un sistema de archivos, pero no actualiza el repositorio de recursos compartidos SMF. Si intenta volver a compartir el recurso existente, se comprueba si hay conflictos en el repositorio de recursos compartidos y se muestra un error.

Resolución de problemas de uso compartido de sistemas de archivos ZFS

Revise los siguientes escenarios y consideraciones de comportamiento de los recursos compartidos:

- Las propiedades de los recursos compartidos y los archivos `.zfs/shares` se tratan de forma diferente en las operaciones `zfs clone` y `zfs send`. Los archivos `.zfs/shares` se incluyen en instantáneas y se preservan en operaciones `zfs clone` y `zfs send`. Las propiedades de uso compartido, incluidos los recursos compartidos designados, no se incluyen en las instantáneas. Para obtener una descripción del comportamiento de las propiedades durante las operaciones `zfs send` y `zfs receive`, consulte "Aplicación de valores de propiedad diferentes a un flujo de instantáneas de ZFS" en la página 234. Después de una operación de clonación, todos los archivos provienen de la instantánea antes de la clonación, mientras que las propiedades se heredan de la nueva posición de la clonación en la jerarquía del sistema de archivos ZFS.
- Determinadas operaciones de uso compartido antiguas desactivan automáticamente el recurso compartido automático o convierten un recurso compartido automático existente en un recurso compartido designado equivalente. Si un sistema de archivos no se comparte como se esperaba, compruebe si el valor `share.auto` está definido en `off`.
- Si una solicitud para crear un recurso compartido designado falla porque el recurso compartido puede entrar en conflicto con el recurso compartido automático, es posible que deba desactivar el recurso compartido automático para continuar.
- Cuando una agrupación se importa como sólo lectura, no se pueden modificar sus propiedades ni sus archivos. Puede resultar imposible establecer un nuevo uso compartido en esta situación. Si el uso compartido ya estaba establecido antes de exportar la agrupación, se utilizan las características de uso compartido existentes, si es posible.

En la tabla siguiente, se identifican los estados de recursos compartidos conocidos y cómo resolverlos, si es necesario.

Estado de recurso compartido	Descripción	Resolución
INVALID	El recurso compartido no es válido porque es incoherente internamente o porque entra en conflicto con otro recurso compartido.	Intente volver a compartir el recurso no válido con el siguiente comando: `# zfs share FS%share` El uso de este comando muestra un mensaje de error acerca del aspecto del recurso compartido que no aprueba la validación. Corrija este problema y vuelva a probar el recurso compartido.
SHARED	El recurso compartido está compartido.	No es necesaria ninguna.
UNSHARED	El recurso compartido es válido, pero no está compartido.	Utilice el comando `zfs share` para volver a compartir el recurso compartido individual o el sistema de archivos principal.
UNVALIDATED	Aún no se ha validado el recurso compartido. Es posible que el sistema de archivos que contiene el recurso compartido no esté en un estado de uso compartido. Por ejemplo, no está montado o está delegado a una zona que no es la zona actual. De manera alternativa, se crearon las propiedades ZFS que representan el recurso compartido deseado, pero aún no se validaron como un recurso compartido válido.	Utilice el comando `zfs share` para volver a compartir el recurso compartido individual o el sistema de archivos principal. Si el sistema de archivos se puede compartir, el intento de volver a compartir será satisfactorio (y habrá una transición al estado compartido) o fallará (y habrá una transición al estado no válido). También puede usar el comando `share -A` para mostrar todos los recursos compartidos en todos los sistemas de archivos montados. Esto hará que todos los recursos compartidos en sistemas de archivos montados se resuelvan como no compartidos (válidos, pero no compartidos aún) o no válidos.

Configuración de cuotas y reservas de ZFS

Puede usar la propiedad `quota` para establecer un límite en la cantidad de espacio en el disco que puede usar un sistema de archivos. Asimismo, puede usar la propiedad `reservation` para garantizar que un sistema de archivos disponga de una cierta cantidad de espacio en el disco. Ambas propiedades se aplican al sistema de archivos donde se han configurado y a todos los descendientes de ese sistema de archivos.

Es decir, si una cuota se configura en el sistema de archivos `tank/home`, la cantidad total de espacio utilizado por `tank/home` *y todos sus descendientes* no puede superar la cuota. Asimismo, si se concede una reserva a `tank/home`, `tank/home` *y todos sus descendientes* se separan de esa reserva. La propiedad `used` informa de la cantidad de espacio utilizado por un sistema de archivos y todos sus descendientes.

Las propiedades `refquota` y `refreservation` están disponibles para administrar el espacio de sistemas de archivos sin tener en cuenta el espacio en el disco que consumen los descendientes, como las instantáneas y los clones.

En esta versión de Solaris, puede establecer una cuota de *usuario* o *grupo* sobre la cantidad de espacio en el disco consumida por archivos que sean propiedad de un determinado grupo o usuario. Las propiedades de cuota de usuarios y grupos no se pueden establecer en un volumen, en un sistema de archivos que sea anterior a la versión 4, o en una agrupación que sea anterior a la versión 15.

A la hora de determinar las funciones de cuota y reserva que mejor administran los sistemas de archivos se deben tener en cuenta los puntos siguientes:

- Las propiedades `quota` y `reservation` son apropiadas para administrar el espacio en el disco consumido por sistemas de archivos y sus descendientes.
- Las propiedades `refquota` y `refreservation` son apropiadas para administrar el espacio en el disco consumido por sistemas de archivos e instantáneas.
- Establecer la propiedad `refquota` o `refreservation` con un valor más alto que el de la propiedad `quota` o `reservation` no tiene repercusión alguna. Si establece las propiedades de `quota` o `refquota`, fallarán las operaciones que intenten exceder cualquier valor. Es posible exceder un valor de `quota` superior al de `refquota`. Si se ensucian algunos bloques de instantáneas, quizá se exceda realmente el valor de `quota` antes de exceder el valor de `refquota`.
- Las cuotas de usuarios y grupos proporcionan un medio de administrar más fácilmente el espacio en el disco con múltiples cuentas de usuario, como por ejemplo en un entorno universitario.

Para obtener más información sobre la configuración de cuotas y reservas, consulte "Establecimiento de cuotas en sistemas de archivos ZFS" en la página 200 y "Establecimiento de reservas en sistemas de archivos ZFS" en la página 204.

Establecimiento de cuotas en sistemas de archivos ZFS

Las cuotas en los sistemas de archivos ZFS se pueden configurar y visualizar mediante los comandos `zfs set` y `zfs get`. En el ejemplo siguiente, se establece una cuota de 10 GB para `tank/home/jeff`:

```
# zfs set quota=10G tank/home/jeff
# zfs get quota tank/home/jeff
NAME            PROPERTY  VALUE  SOURCE
tank/home/jeff  quota     10G    local
```

Las cuotas también influyen en la salida de los comandos `zfs list` y `df`. Por ejemplo:

```
# zfs list -r tank/home
NAME                USED  AVAIL  REFER  MOUNTPOINT
tank/home          1.45M  66.9G    36K  /tank/home
tank/home/eric      547K  66.9G   547K  /tank/home/eric
tank/home/jeff      322K  10.0G   291K  /tank/home/jeff
tank/home/jeff/ws    31K  10.0G    31K  /tank/home/jeff/ws
tank/home/lori      547K  66.9G   547K  /tank/home/lori
tank/home/mark       31K  66.9G    31K  /tank/home/mark
# df -h /tank/home/jeff
Filesystem            Size  Used Avail Use% Mounted on
tank/home/jeff         10G  306K   10G   1% /tank/home/jeff
```

Tenga en cuenta que, si bien `tank/home` tiene un espacio en disco disponible de 66.9 GB, `tank/home/jeff` y `tank/home/jeff/ws` sólo cuentan con 10 GB de espacio en disco disponible para cada uno, debido a la cuota de `tank/home/jeff`.

No puede configurar una cuota con una cantidad inferior a la que esté usando un sistema de archivos. Por ejemplo:

```
# zfs set quota=10K tank/home/jeff
cannot set property for 'tank/home/jeff':
size is less than current used or reserved space
```

Puede establecer un valor de `refquota` en un sistema de archivos que limite la cantidad de espacio en el disco que puede consumir el sistema de archivos. Este límite fijo no incluye el espacio en el disco consumido por descendientes. Por ejemplo, la cuota de 10 GB de studentA no se ve afectada por el espacio utilizado por las instantáneas.

```
# zfs set refquota=10g students/studentA
# zfs list -t all -r students
NAME                          USED  AVAIL  REFER  MOUNTPOINT
students                      150M  66.8G    32K  /students
students/studentA             150M  9.85G   150M  /students/studentA
students/studentA@yesterday      0      -   150M  -
# zfs snapshot students/studentA@today
# zfs list -t all -r students
students                      150M  66.8G    32K  /students
students/studentA             150M  9.90G   100M  /students/studentA
students/studentA@yesterday  50.0M      -   150M  -
students/studentA@today          0      -   100M  -
```

Para mayor comodidad, puede establecer otra cuota en un sistema de archivos para administrar mejor el espacio que consumen las instantáneas. Por ejemplo:

```
# zfs set quota=20g students/studentA
# zfs list -t all -r students
NAME                          USED  AVAIL  REFER  MOUNTPOINT
students                      150M  66.8G    32K  /students
students/studentA             150M  9.90G   100M  /students/studentA
students/studentA@yesterday  50.0M      -   150M  -
students/studentA@today          0      -   100M  -
```

En esta situación hipotética, `studentA` puede entrar en conflicto con el límite físico de `refquota` (10 GB), pero `studentA` puede eliminar archivos que recuperar aunque haya instantáneas.

En el ejemplo anterior, la menor de las dos cuotas (10 GB si se compara con 20 GB) aparece en la salida `zfs list`. Para ver el valor de las dos cuotas, use el comando `zfs get`. Por ejemplo:

```
# zfs get refquota,quota students/studentA
NAME               PROPERTY  VALUE              SOURCE
students/studentA  refquota  10G                local
students/studentA  quota     20G                local
```

Establecimiento de las cuotas de usuarios y grupos en un sistema de archivos ZFS

Puede definir la cuota de un grupo o un usuario mediante el uso de los comandos `zfs userquota` y `zfs groupquota`, respectivamente. Por ejemplo:

```
# zfs create students/compsci
# zfs set userquota@student1=10G students/compsci
# zfs create students/labstaff
# zfs set groupquota@labstaff=20GB students/labstaff
```

Visualice la cuota del grupo o la del usuario actual como se indica a continuación:

```
# zfs get userquota@student1 students/compsci
NAME              PROPERTY            VALUE               SOURCE
students/compsci  userquota@student1  10G                 local
# zfs get groupquota@labstaff students/labstaff
NAME               PROPERTY             VALUE                SOURCE
students/labstaff  groupquota@labstaff  20G                  local
```

Puede mostrar el uso general del espacio en el disco del usuario o grupo mediante la consulta de las propiedades siguientes:

```
# zfs userspace students/compsci
TYPE        NAME      USED  QUOTA
POSIX User  root      350M   none
POSIX User  student1  426M    10G
# zfs groupspace students/labstaff
TYPE         NAME      USED  QUOTA
POSIX Group  labstaff  250M    20G
POSIX Group  root      350M   none
```

Para identificar el uso individual del espacio en el disco de un usuario o grupo, consulte las propiedades siguientes:

```
# zfs get userused@student1 students/compsci
NAME              PROPERTY           VALUE              SOURCE
students/compsci  userused@student1  550M               local
# zfs get groupused@labstaff students/labstaff
NAME               PROPERTY            VALUE               SOURCE
students/labstaff  groupused@labstaff  250                 local
```

Las propiedades de cuotas de grupos y usuarios no se muestran si utiliza el comando `zfs get all` *del conjunto de datos*, que muestra una lista de todas las propiedades del sistema de archivos.

Puede eliminar la cuota de un grupo o usuario como se indica a continuación:

```
# zfs set userquota@student1=none students/compsci
# zfs set groupquota@labstaff=none students/labstaff
```

Las cuotas de usuarios o grupos en sistemas de archivos ZFS proporcionan las siguientes funciones:

- La cuota de un usuario o grupo que se define en un sistema de archivos superior no la hereda automáticamente un sistema de archivos descendiente.
- Sin embargo, la cuota del grupo o usuario se aplica cuando se crea una instantánea o un clon a partir de un sistema de archivos que tiene una cuota de grupo o usuario. Del mismo modo, se incluye una cuota de usuario o grupo en el sistema de archivos cuando se crea una secuencia mediante el comando `zfs send`, incluso sin opción `-R`.
- Los usuarios sin privilegios sólo pueden acceder al uso de su propio espacio en el disco. El usuario root o el usuario al que se le haya concedido el privilegio `userused` o `groupused`, puede acceder a la información de cálculo de espacio de grupos o usuarios de todos.
- Las propiedades `userquota` y `groupquota` no se pueden establecer en volúmenes de ZFS, en un sistema de archivos anterior a la versión 4, o en una agrupación anterior a la versión 15.

La aplicación de cuotas de usuario o grupo puede retrasarse en varios segundos. Este retraso significa que los usuarios pueden exceder su cuota antes de que el sistema perciba que se ha sobrepasado la cuota y que rechace la acción de escritura con posterioridad al mensaje de error `EDQUOT`.

Puede utilizar el comando `quota` heredado para revisar las cuotas del usuario en un entorno NFS; por ejemplo, donde se haya montado un sistema de archivos ZFS. Sin ninguna opción, el comando `quota` sólo muestra la salida si se ha superado la cuota del usuario. Por ejemplo:

```
# zfs set userquota@student1=10m students/compsci
# zfs userspace students/compsci
TYPE        NAME      USED  QUOTA
POSIX User  root      350M   none
POSIX User  student1  550M    10M
# quota student1
Block limit reached on /students/compsci
```

Si reinicia la cuota de usuario y el límite de cuota ya no se supera, podrá utilizar el comando `quota -v` para revisar la cuota del usuario. Por ejemplo:

```
# zfs set userquota@student1=10GB students/compsci
# zfs userspace students/compsci
TYPE        NAME      USED  QUOTA
POSIX User  root      350M   none
POSIX User  student1  550M    10G
# quota student1
# quota -v student1
Disk quotas for student1 (uid 102):
Filesystem     usage  quota  limit    timeleft  files  quota  limit    timeleft
```

```
/students/compsci
             563287 10485760 10485760           -      -      -      -           -
```

Establecimiento de reservas en sistemas de archivos ZFS

Una *reserva* de ZFS es una asignación de espacio en el disco de la agrupación cuya disponibilidad en un conjunto de datos está garantizada. Así, no puede reservar espacio en el disco para un conjunto de datos si ese espacio no está disponible en la agrupación. La cantidad total de todas las reservas pendientes sin consumir no puede superar la cantidad de espacio en el disco sin utilizar de la agrupación. Las reservas de ZFS se pueden configurar y visualizar mediante los comandos `zfs set` y `zfs get`. Por ejemplo:

```
# zfs set reservation=5G tank/home/bill
# zfs get reservation tank/home/bill
NAME            PROPERTY     VALUE   SOURCE
tank/home/bill  reservation  5G      local
```

Las reservas de pueden afectar a la salida del comando `zfs list`. Por ejemplo:

```
# zfs list -r tank/home
NAME                USED  AVAIL  REFER  MOUNTPOINT
tank/home          5.00G  61.9G    37K  /tank/home
tank/home/bill       31K  66.9G    31K  /tank/home/bill
tank/home/jeff      337K  10.0G   306K  /tank/home/jeff
tank/home/lori      547K  61.9G   547K  /tank/home/lori
tank/home/mark       31K  61.9G    31K  /tank/home/mark
```

`tank/home` utiliza 5 GB de espacio, aunque la cantidad total de espacio a la que hacen referencia `tank/home` y sus descendientes es mucho menor que 5 GB. El espacio utilizado refleja el espacio reservado para `tank/home/bill`. Las reservas se tienen en cuenta en el espacio en el disco utilizado del sistema de archivos superior y se contabilizan en relación con su cuota, reserva o ambas.

```
# zfs set quota=5G pool/filesystem
# zfs set reservation=10G pool/filesystem/user1
cannot set reservation for 'pool/filesystem/user1': size is greater than
available space
```

Un conjunto de datos puede usar más espacio en el disco que su reserva, siempre que haya espacio disponible en la agrupación que no esté reservado y que el uso actual del conjunto de datos esté por debajo de su cuota. Un conjunto de datos no puede consumir espacio en el disco reservado a otro conjunto de datos.

Las reservas no son acumulativas. Es decir, una segunda invocación de `zfs set` para configurar una reserva no agrega su reserva a la que ya existe, sino que la segunda reserva sustituye la primera. Por ejemplo:

```
# zfs set reservation=10G tank/home/bill
# zfs set reservation=5G tank/home/bill
# zfs get reservation tank/home/bill
NAME            PROPERTY     VALUE   SOURCE
tank/home/bill  reservation  5G      local
```

Puede establecer una reserva `refreservation` para garantizar espacio en el disco para un conjunto de datos que no incluya espacio en el disco consumido por instantáneas y clones. Esta reserva se explica en el cálculo del espacio utilizado en los conjuntos de datos principales, y repercute en las cuotas y reservas del conjunto de datos superior. Por ejemplo:

```
# zfs set refreservation=10g profs/prof1
# zfs list
NAME                     USED  AVAIL  REFER  MOUNTPOINT
profs                   10.0G  23.2G    19K  /profs
profs/prof1               10G  33.2G    18K  /profs/prof1
```

También se puede establecer una reserva en el mismo conjunto de datos para garantizar espacio de conjunto de datos e instantáneas. Por ejemplo:

```
# zfs set reservation=20g profs/prof1
# zfs list
NAME                     USED  AVAIL  REFER  MOUNTPOINT
profs                   20.0G  13.2G    19K  /profs
profs/prof1               10G  33.2G    18K  /profs/prof1
```

Las reservas regulares se explican en el cálculo del espacio utilizado en el principal.

En el ejemplo anterior, la menor de las dos cuotas (10 GB si se compara con 20 GB) aparece en la salida `zfs list`. Para ver el valor de las dos cuotas, use el comando `zfs get`. Por ejemplo:

```
# zfs get reservation,refreserv profs/prof1
NAME         PROPERTY        VALUE        SOURCE
profs/prof1  reservation     20G          local
profs/prof1  refreservation  10G          local
```

Si se establece `refreservation`, sólo se permite una instantánea en caso de que fuera de esta reserva exista suficiente espacio no reservado en la agrupación para alojar la cantidad actual de bytes *a los que se hace referencia* en el conjunto de datos.

Cifrado de sistemas de archivos ZFS

El cifrado es el proceso en que los datos se codifican para fines de privacidad, y el propietario de los datos necesita una clave para acceder a los datos codificados. Los beneficios de utilizar el cifrado ZFS son los siguientes:

- El cifrado ZFS está integrado con el conjunto de comandos ZFS. Al igual que otras operaciones de ZFS, las operaciones de cifrado, como los cambios de claves y la regeneración de claves, se realizan en línea.

- Puede utilizar las agrupaciones de almacenamiento existentes cuando se actualizan. Tiene la posibilidad de cifrar solamente determinados sistemas de archivos.
- Los sistemas de archivos subordinados pueden heredar el cifrado ZFS. La gestión de claves se puede delegar mediante la administración delegada de ZFS.
- Los datos se cifran mediante AES (estándar de cifrado avanzado) con longitudes de clave de 128, 192 y 256, en los modos de operación CCM y GCM.
- El cifrado de ZFS utiliza la estructura criptográfica de Oracle Solaris, la cual le da acceso a cualquier aceleración de hardware disponible o a implementaciones de software optimizado de los algoritmos de cifrado de forma automática.
- Actualmente, no puede cifrar el sistema de archivos raíz ZFS u otros componentes del sistema operativo, como el directorio `/var`, incluso si se trata de un sistema de archivos independiente.

Puede establecer una política de cifrado cuando se crea un sistema de archivos ZFS, pero la política no se puede cambiar. Por ejemplo, el sistema de archivos `tank/home/darren` se crea con la propiedad de cifrado activada. La política de cifrado predeterminada es solicitar una frase de contraseña, que deberá tener un mínimo de 8 caracteres de longitud.

```
# zfs create -o encryption=on tank/home/darren
Enter passphrase for 'tank/home/darren': xxxxxxx
Enter again: xxxxxxxx
```

Confirme que el sistema de archivos tiene activado el cifrado. Por ejemplo:

```
# zfs get encryption tank/home/darren
NAME              PROPERTY    VALUE        SOURCE
tank/home/darren  encryption  on           local
```

El algoritmo de cifrado predeterminado es `aes-128-ccm` cuando el valor de cifrado de un sistema de archivos está `on` (activado).

Una *clave de ajuste* se utiliza para cifrar las claves de cifrado de datos reales. La clave de ajuste es enviada del comando `zfs`, como en el ejemplo anterior cuando se crea el sistema de archivos cifrados, al núcleo. Una clave de ajuste se encuentra en un archivo (sin formato o en formato hexadecimal) o deriva de una frase de contraseña.

El formato y la ubicación de la clave de ajuste se especifican en la propiedad `keysource`, como se indica a continuación:

```
keysource=format,location
```

- El formato es uno de los siguientes:
  - `raw`: los bytes de clave sin formato.
  - `hex`: una cadena de clave hexadecimanl
  - `passphrase`: una cadena de caracteres que genera una clave.
- La ubicación es una de las siguientes:

- `prompt`: se le solicitará una clave o una frase de contraseña cuando se crea o se monta el sistema de archivos.
- `file:///`*filename*: la ubicación de un archivo de frase de contraseña o clave en un sistema de archivos
- `pkcs11`: URI que describe la ubicación de un archivo de frase de contraseña o clave en un token PKCS#11
- `https://`*location*: ubicación de un archivo de frase de contraseña o clave en un servidor seguro. No se recomienda transportar información de claves sin cifrar con este método. La operación `GET` en la URL devuelve solamente el valor de la clave o frase de contraseña, según lo solicitado en la parte de formato de la propiedad `keysource`.

  Al utilizar un localizador `https://` para `keysource`, el certificado que el servidor presenta debe ser uno que tenga una relación de confianza con `libcurl` y OpenSSL. Agregue su propio anclaje de confianza o certificado autofirmado al almacén de certificados en `/etc/openssl/certs`. Coloque el certificado con formato PEM en el directorio `/etc/certs/CA` y ejecute el comando siguiente:

  ```
  # svcadm refresh ca-certificates
  ```

Si el formato de `keysource` es *frase de contraseña*, la clave de ajuste deriva de la frase de contraseña. De lo contrario, el valor de la propiedad `keysource` apunta a la clave de ajuste real, como bytes sin formato o en formato hexadecimal. Puede especificar que la frase de contraseña se almacene en un archivo o en un flujo de bytes sin formato que se soliciten, lo cual, probablemente, sólo sea adecuado para secuencias de comandos.

Cuando los valores de la propiedad `keysource` de un sistema de archivos identifican la `passphrase`, la clave de ajuste deriva de la frase de contraseña utilizando `PKCS#5 PBKD2` y una sal generada de forma aleatoria por sistema de archivos. Esto significa que la misma frase de contraseña genera una clave de ajuste diferente si se utiliza en sistemas de archivos descendientes.

La política de cifrado de un sistema de archivos es heredada por sistemas de archivos descendientes y no se puede eliminar. Por ejemplo:

```
# zfs snapshot tank/home/darren@now
# zfs clone tank/home/darren@now tank/home/darren-new
Enter passphrase for 'tank/home/darren-new': xxxxxxx
Enter again: xxxxxxxx
# zfs set encryption=off tank/home/darren-new
cannot set property for 'tank/home/darren-new': 'encryption' is readonly
```

Si necesita copiar o migrar sistemas de archivos ZFS cifrados o no cifrados, tenga en cuenta los siguientes puntos:

- Actualmente, no se puede enviar un flujo de conjunto de datos no cifrado y recibirlo como un flujo cifrado, aunque el conjunto de datos de la agrupación receptora tenga activado el cifrado.

- Puede utilizar los siguientes comandos para migrar datos no cifrados a una agrupación o a un sistema de archivos con cifrado activado:
  - `cp -r`
  - `find | cpio`
  - `tar`
  - `rsync`
- Un flujo replicado de sistema de archivos cifrado puede ser recibido en un sistema de archivos cifrado y los datos permanecen cifrados. Para obtener más información, consulte el Ejemplo 5–4.

Cambio de claves de un sistema de archivos ZFS cifrado

Puede cambiar la clave de ajuste de un sistema de archivos cifrado utilizando el comando `zfs key -c`. La clave de ajuste existente debe haber sido cargada primero, en el momento del inicio o al cargar la clave del sistema de archivos de forma explícita (`zfs key -l`), o al montar el sistema de archivos (`zfs mount` *sistema de archivos*). Por ejemplo:

```
# zfs key -c tank/home/darren
Enter new passphrase for 'tank/home/darren': xxxxxxxx
Enter again: xxxxxxxx
```

En el siguiente ejemplo, se cambia la clave de ajuste y el valor de la propiedad `keysource` cambia para especificar que la clave de ajuste proviene de un archivo.

```
# zfs key -c -o keysource=raw,file:///media/stick/key tank/home/darren
```

La clave de cifrado de datos para un sistema de archivos cifrado se puede cambiar mediante el comando `zfs key -K`, pero la nueva clave de cifrado sólo se utiliza para los datos escritos recientemente. Esta función se puede utilizar para proporcionar conformidad con las pautas NIST 800-57 sobre el límite de tiempo de una clave de cifrado de datos. Por ejemplo:

```
# zfs key -K tank/home/darren
```

En el ejemplo anterior, la clave de cifrado de datos no está visible ni está directamente gestionada por usted. Además, necesita la delegación de `keychange` para realizar una operación de cambio de clave.

Los siguientes algoritmos de cifrado están disponibles:

- `aes-128-ccm`, `aes-192-ccm`, `aes-256-ccm`
- `aes-128-gcm`, `aes-192-gcm`, `aes-256-gcm`

La propiedad `keysource` de ZFS identifica el formato y la ubicación de la clave que se ajusta a las claves de cifrado de datos del sistema de archivos. Por ejemplo:

```
# zfs get keysource tank/home/darren
NAME              PROPERTY   VALUE              SOURCE
tank/home/darren  keysource  passphrase,prompt  local
```

La propiedad `rekeydate` de ZFS identifica la fecha de la última operación de `zfs key -K`. Por ejemplo:

```
# zfs get rekeydate tank/home/darren
NAME              PROPERTY   VALUE                  SOURCE
tank/home/darren  rekeydate  Wed Jul 25 16:54 2012  local
```

Si las propiedades `creation` y `rekeydate` de un sistema de archivos cifrado tienen el mismo valor, las claves del sistema de archivos nunca han sido regeneradas por una operación `zfs key -K`.

Gestión de claves de cifrado ZFS

Las claves de cifrado ZFS se pueden gestionar de diferentes maneras, según sus necesidades, ya sea en el sistema local o de forma remota si se necesita una ubicación centralizada.

- **Localmente**: los ejemplos anteriores muestran que la clave de ajuste puede ser una solicitud de frase de contraseña o una clave sin formato almacenada en un archivo del sistema local.
- **Remotamente**: la información de clave puede estar almacenada de forma remota mediante un sistema de gestión de claves centralizado, como Oracle Key Manager, o con un servicio web que admite una simple solicitud GET en una URI http o https. Un sistema Oracle Solaris puede acceder a la información de claves de Oracle Key Manager mediante el token PKCS#11.

Para obtener más información sobre la gestión de claves de cifrado ZFS, consulte

`http://www.oracle.com/technetwork/articles/servers-storage-admin/manage-zfs-encryption-1715034.html`

Para obtener información sobre el uso de Oracle Key Manager para gestionar información de claves, consulte:

`http://docs.oracle.com/cd/E24472_02/`

Delegación de permisos de operaciones de claves de ZFS

Revise las siguientes descripciones de permisos para delegar operaciones de claves:

- La carga o descarga de una clave del sistema de archivos utilizando los comandos `zfs key -l` y `zfs key -u` requieren el permiso de `key`. En la mayoría de los casos, también necesitará el permiso `mount`.
- El cambio de la clave de un sistema de archivos utilizando los comandos `zfs key -c` y `zfs key -K` requiere el permiso de `keychange`.

Considere delegar permisos separados para el uso de claves (carga o descarga) y para los cambios de claves, lo que le permite tener un modelo de operaciones de claves de dos personas. Por ejemplo, determine los usuarios que pueden utilizar las claves y los usuarios que pueden cambiarlas. O bien, ambos usuarios deben estar presentes para realizar cambios en las claves. Este modelo también permite crear un sistema de custodia de claves.

Montaje de un sistema de archivos ZFS cifrado

Revise las siguientes consideraciones al intentar montar un sistema de archivos ZFS cifrado:

- Si una clave del sistema de archivos cifrado no está disponible en el momento del inicio, el sistema de archivos no se monta automáticamente. Por ejemplo, un sistema de archivos con un conjunto de políticas de cifrado establecido en `passphrase,prompt` no se montará durante el momento del inicio, ya que el proceso de inicio no se interrumpe para solicitar una frase de contraseña.
- Si desea montar un sistema de archivos con un conjunto de políticas de cifrado establecido en `passphrase,prompt` en el momento del inicio, deberá montarlo de forma explícita con el comando `zfs mount` y especificar la frase de contraseña o utilizar el comando `zfs key -l` para que se le solicite la clave después de iniciar el sistema.

  Por ejemplo:

  ```
  # zfs mount -a
  Enter passphrase for 'tank/home/darren': xxxxxxxx
  Enter passphrase for 'tank/home/ws': xxxxxxxx
  Enter passphrase for 'tank/home/mark': xxxxxxxx
  ```
- Si la propiedad `keysource` de un sistema de archivos cifrado apunta a un archivo en otro sistema de archivos, el orden de montaje de los sistemas de archivos se puede afectar si el sistema de archivos cifrado se monta en el inicio y , en especial, si el archivo se encuentra en un soporte extraíble.

Actualización de sistemas de archivos ZFS cifrados

Antes de actualizar el sistema Solaris 11 al sistema Solaris 11.1, asegúrese de que se monten los sistemas de archivos cifrados. Monte los sistemas de archivos cifrados y proporcione las frases de contraseña, si se le solicitan.

```
# zfs mount -a
Enter passphrase for 'pond/amy': xxxxxxxx
Enter passphrase for 'pond/rory': xxxxxxxx
# zfs mount | grep pond
pond                            /pond
pond/amy                        /pond/amy
pond/rory                       /pond/rory
```

A continuación, actualice los sistemas de archivos cifrados.

```
# zfs upgrade -a
```

Si intenta actualizar los sistemas de archivos ZFS cifrados que no están montados, se muestra un mensaje similar al siguiente:

```
# zfs upgrade -a
cannot set property for 'pond/amy': key not present
```

Además, la salida de `zpool status` puede mostrar datos dañados.

```
# zpool status -v pond
.
.
.
        pond/amy:<0x1>
        pond/rory:<0x1>
```

Si se producen los errores anteriores, vuelva a montar los sistemas de archivos cifrados como se indica arriba. Después, solucione los errores de la agrupación.

```
# zpool scrub pond
# zpool clear pond
```

Para obtener más información sobre la actualización de sistemas de archivos, consulte "Actualización de sistemas de archivos ZFS" en la página 216.

Interacciones entre propiedades de compresión, eliminación de datos duplicados y cifrado de ZFS

Revise las siguientes consideraciones al utilizar propiedades de compresión, eliminación de datos duplicados y cifrado de ZFS:

- Cuando se escribe un archivo, se comprimen y se cifran los datos y se verifica la suma de comprobación. Luego, si es posible, se eliminan los datos duplicados.
- Cuando se lee un archivo, se verifica la suma de comprobación y se descifran los datos. Luego, de ser necesario, se descomprimen los datos.
- Si la propiedad `dedup` está activada en un sistema de archivos cifrado que también está clonado y en los clones no se ha utilizado el comando `zfs key -K` o el comando `zfs clone -K`, se eliminan los datos duplicados de todos los clones, si es posible.

Ejemplos de cifrado de sistemas de archivos ZFS

EJEMPLO 5–1 Cifrado de un sistema de archivos ZFS mediante una clave sin formato

En el siguiente ejemplo, se utiliza el comando `pktool` para generar una clave de cifrado `aes-256-ccm` escribirla en un archivo: `/cindykey.file`.

```
# pktool genkey keystore=file outkey=/cindykey.file keytype=aes keylen=256
```

Luego, el archivo `/cindykey.file` se especifica cuando se crea el sistema de archivos `tank/home/cindy`.

```
# zfs create -o encryption=aes-256-ccm -o keysource=raw,file:///cindykey.file
tank/home/cindy
```

EJEMPLO 5–2 Cifrado de un sistema de archivos ZFS con un algoritmo de cifrado diferente

Puede crear una agrupación de almacenamiento ZFS y hacer que todos los sistemas de archivos en la agrupación de almacenamiento hereden un algoritmo de cifrado. En este ejemplo, se crea la agrupación `users` y se crea el sistema de archivos `users/home` y se cifra utilizando una frase de contraseña. El algoritmo de cifrado predeterminado es `aes-128-ccm`.

Luego, se crea el sistema de archivos `users/home/mark` y se cifra utilizando el algoritmo de cifrado `aes-256-ccm`.

```
# zpool create -O encryption=on users mirror c0t1d0 c1t1d0 mirror c2t1d0 c3t1d0
Enter passphrase for 'users': xxxxxxxx
Enter again: xxxxxxxx
# zfs create users/home
# zfs get encryption users/home
NAME        PROPERTY    VALUE        SOURCE
users/home  encryption  on           inherited from users
# zfs create -o encryption=aes-256-ccm users/home/mark
# zfs get encryption users/home/mark
NAME                PROPERTY    VALUE        SOURCE
users/home/mark     encryption  aes-256-ccm  local
```

EJEMPLO 5–3 Clonación de un sistema de archivos ZFS cifrado

Si el sistema de archivos clonado hereda la propiedad `keysource` del mismo sistema de archivos que su instantánea de origen, no es necesario un nuevo `keysource` y no se le solicitará una nueva frase de contraseña si `keysource=passphrase,prompt`. El mismo `keysource` se utiliza para el clon. Por ejemplo:

De manera predeterminada, no se le solicitará una clave al clonar un descendiente de un sistema de archivos cifrado.

```
# zfs create -o encryption=on tank/ws
Enter passphrase for 'tank/ws': xxxxxxxx
Enter again: xxxxxxxx
```

EJEMPLO 5–3 Clonación de un sistema de archivos ZFS cifrado *(Continuación)*

```
# zfs create tank/ws/fs1
# zfs snapshot tank/ws/fs1@snap1
# zfs clone tank/ws/fs1@snap1 tank/ws/fs1clone
```

Si desea crear una nueva clave para el sistema de archivos clonado, utilice el comando `zfs clone -K`.

Si realiza la clonación de un sistema de archivos cifrado en lugar de un sistema de archivos cifrado descendiente, se le solicitará que proporcione una nueva clave. Por ejemplo:

```
# zfs create -o encryption=on tank/ws
Enter passphrase for 'tank/ws': xxxxxxxx
Enter again: xxxxxxxx
# zfs snapshot tank/ws@1
# zfs clone tank/ws@1 tank/ws1clone
Enter passphrase for 'tank/ws1clone': xxxxxxxx
Enter again: xxxxxxxx
```

EJEMPLO 5–4 Envío y recepción de un sistema de archivos ZFS cifrado

En el siguiente ejemplo, se crea la instantánea `tank/home/darren@snap1` a partir del sistema de archivos cifrado `/tank/home/darren`. Luego, la instantánea se envía a `bpool/snaps` con la propiedad de cifrado activada para que se cifren los datos recibidos resultantes. Sin embargo, el flujo `tank/home/darren@snap1` no se cifra durante el proceso de envío.

```
# zfs get encryption tank/home/darren
NAME              PROPERTY    VALUE        SOURCE
tank/home/darren  encryption  on           local
# zfs snapshot tank/home/darren@snap1
# zfs get encryption bpool/snaps
NAME         PROPERTY    VALUE        SOURCE
bpool/snaps  encryption  on           inherited from bpool
# zfs send tank/home/darren@snap1 | zfs receive bpool/snaps/darren1012
# zfs get encryption bpool/snaps/darren1012
NAME                     PROPERTY    VALUE        SOURCE
bpool/snaps/darren1012   encryption  on           inherited from bpool
```

En este caso, se genera una nueva clave de forma automática para el sistema de archivos cifrado recibido.

Migración de sistemas de archivos ZFS

Puede utilizar la función de migración shadow para migrar sistemas de archivos de la siguiente forma:

- Un sistema de archivos ZFS local o remoto a un sistema de archivos ZFS de destino
- Un sistema de archivos UFS local o remoto a un sistema de archivos ZFS de destino

La *migración shadow* es un proceso que extrae los datos que se van a migrar:

- Cree un sistema de archivos ZFS vacío.
- Establezca la propiedad `shadow` en un sistema de archivos ZFS vacío, que es el sistema de archivos de destino (o shadow), a fin de apuntar al sistema de archivos que se va a migrar.
- Los datos del sistema de archivos que se van a migrar se copian en el sistema de archivos shadow.

Puede utilizar el URI de la propiedad shadow para identificar el sistema de archivos que se va a migrar de dos formas:

- `shadow=file:///`*ruta* - Utilice esta sintaxis para migrar un sistema de archivos local.
- `shadow=nfs://`*host:ruta* - Utilice esta sintaxis para migrar un sistema de archivos NFS

Tenga en cuenta las siguientes consideraciones cuando migre sistemas de archivos:

- El sistema de archivos para migrar se debe definir como de sólo lectura. Si el sistema de archivos no se define como de sólo lectura, puede que no se migren los cambios que se encuentren en curso.
- El sistema de archivos de destino debe estar completamente vacío.
- Si el sistema se reinicia durante una migración, la migración continúa después de iniciar el sistema.
- El acceso al contenido del directorio que no esté completamente migrado o el acceso al contenido de los archivos contenido que no estén completamente migrados se bloquea hasta que se migre todo el contenido.
- Si desea que la información de UID, GID y ACL se migre al sistema de archivos shadow durante una migración de NFS, asegúrese de que la información del nombre de servicio esté accesible entre los sistemas locales y los remotos. Es posible que considere copiar un subconjunto de datos del sistema de archivos a fin de migrarlos para realizar una prueba de migración con el fin de comprobar que toda la información se migra de forma adecuada antes de realizar una migración de datos de gran tamaño a través de NFS.
- Migrar los datos del sistema de archivos por medio de NFS puede resultar lento según el ancho de banda de la red. Sea paciente.
- Puede utilizar el comando `shadowstat` para supervisar la migración de un sistema de archivos, lo que proporciona los siguientes datos:
  - La columna `BYTES XFRD` identifica la cantidad de bytes que se han transferido al sistema de archivos shadow.
  - La columna `BYTES LEFT` cambia continuamente hasta que la migración está prácticamente completa. ZFS no identifica al principio de la migración la cantidad de datos que se deben migrar debido a que este proceso puede tardar demasiado tiempo.
  - Considere utilizar la información de `BYTES XFRD` y `ELAPSED TIME` para calcular la duración del proceso de migración.

▼ Cómo migrar un sistema de archivos a un sistema de archivos ZFS

1 Si va a migrar datos desde un servidor NFS remoto, confirme que se pueda acceder a la información del servicio de nombres en ambos sistemas.

Para realizar una migración de gran tamaño utilizando NFS, puede considerar realizar una migración de prueba de un subconjunto de datos a fin de garantizar que la información de UID, GUID y ACL se migra correctamente.

2 Instale el paquete de migración shadow en el sistema al que se migrarán los datos, si es necesario, y active el servicio `shadowd` para facilitar el proceso de migración.

```
# pkg install shadow-migration

# svcadm enable shadowd
```

Si no activa el proceso `shadowd`, deberá restablecer la propiedad `shadow` a `none` cuando el proceso de migración se haya completado.

3 Defina como de sólo lectura el sistema de archivos local o remoto que se migrará.

Si va a migrar un sistema de archivos ZFS local, defínalo como de sólo lectura. Por ejemplo:

```
# zfs set readonly=on tank/home/data
```

Si va a migrar un sistema de archivos remoto, compártalo en modo de sólo lectura. Por ejemplo,

```
# share -F nfs -o ro /export/home/ufsdata
# share
-               /export/home/ufsdata   ro   ""
```

4 Cree un nuevo sistema de archivos ZFS con la propiedad shadow establecida en el sistema de archivos que se va a migrar.

Por ejemplo, si va a migrar un sistema de archivos ZFS local, `rpool/old`, a un nuevo sistema de archivos ZFS, `users/home/shadow`, establezca la propiedad `shadow` en `rpool/old` cuando se crea el sistema de archivos `users/home/shadow`.

```
# zfs create -o shadow=file:///rpool/old users/home/shadow
```

Por ejemplo, para migrar `/export/home/ufsdata` desde un servidor remoto, establezca la propiedad `shadow` cuando se crea el sistema de archivos ZFS.

```
# zfs create -o shadow=nfs://neo/export/home/ufsdata users/home/shadow2
```

5 Compruebe el progreso de la migración.

Por ejemplo:

```
# shadowstat
                                    EST
                          BYTES     BYTES            ELAPSED
```

```
DATASET                         XFRD    LEFT    ERRORS  TIME
users/home/shadow               45.5M   2.75M   -       00:02:31
users/home/shadow               55.8M   -       -       00:02:41
users/home/shadow               69.7M   -       -       00:02:51
No migrations in progress
```

Cuando se completa la migración, la propiedad `shadow` se establece en `none`.

```
# zfs get -r shadow users/home/shadow*
NAME                PROPERTY  VALUE   SOURCE
users/home/shadow   shadow    none    -
users/home/shadow2  shadow    none    -
```

Resolución de problemas de migraciones del sistema de archivos ZFS

Tenga en cuenta los siguientes puntos al resolver problemas de migración de ZFS:

- Si el sistema de archivos que se va a migrar no está definido como de sólo lectura, no se van a migrar todos los datos.
- Si el sistema de archivos de destino no está vacío cuando se establece la propiedad `shadow`, no se iniciará la migración de datos.
- Si cuando la migración está en curso agrega datos al sistema de archivos que se va a migrar o elimina datos de dicho sistema, es posible que esos cambios no se migren.
- Si cuando la migración está en curso intenta cambiar el montaje del sistema de archivos shadow, aparecerá el siguiente mensaje:

  ```
  # zfs set mountpoint=/users/home/data users/home/shadow3
  cannot unmount '/users/home/shadow3': Device busy
  ```

Actualización de sistemas de archivos ZFS

Si tienen sistemas de archivos ZFS de una versión anterior de Solaris, puede actualizar sus sistemas de archivos con el comando `zfs upgrade` para aprovechar las funciones de sistema de archivos de la versión actual. Además, este comando le notifica cuando los sistemas de archivos están ejecutando versiones antiguas.

Por ejemplo, este sistema de archivos está en la versión actual 5.

```
# zfs upgrade
This system is currently running ZFS filesystem version 5.

All filesystems are formatted with the current version.
```

Utilice este comando para identificar las funciones disponibles para cada versión del sistema de archivos.

```
# zfs upgrade -v
The following filesystem versions are supported:

VER  DESCRIPTION
---  --------------------------------------------------------
 1   Initial ZFS filesystem version
 2   Enhanced directory entries
 3   Case insensitive and File system unique identifier (FUID)
 4   userquota, groupquota properties
 5   System attributes

For more information on a particular version, including supported releases,
see the ZFS Administration Guide.
```

Para obtener más información sobre la actualización de sistemas de archivos cifrados, consulte"Actualización de sistemas de archivos ZFS cifrados" en la página 210

◆ ◆ ◆ CAPÍTULO 6

Uso de clones e instantáneas de Oracle Solaris ZFS

En este capítulo se explican procedimientos para crear y administrar clones e instantáneas de Oracle Solaris ZFS. También contiene información sobre cómo guardar instantáneas.

Este capítulo se divide en las secciones siguientes:

Información general de instantáneas de ZFS

Una *instantánea* es una copia de sólo lectura de un sistema de archivos o volumen. Las instantáneas se pueden crear de forma casi inmediata y al principio consumen poco espacio en el disco de la agrupación. Sin embargo, a medida que los datos dentro del conjunto de datos activo cambian, la instantánea consume espacio en el disco, ya que sigue haciendo referencia a los datos antiguos e impide que el espacio en disco se libere.

Las instantáneas de ZFS presentan las características siguientes:

- Se mantienen en sucesivos reinicios del sistema.
- El número máximo teórico de instantáneas es 2^{64}.
- Las instantáneas no utilizan un almacén de copia de seguridad independiente. Las instantáneas consumen espacio en el disco directamente de la misma agrupación de almacenamiento que el sistema de archivos o el volumen a partir del que se crearon.

- Las instantáneas recursivas se crean rápidamente como una operación atómica Las instantáneas se crean todas juntas (todas a la vez) o no se crea ninguna. La ventaja de las operaciones atómicas de instantáneas estriba en que los datos se toman siempre en un momento coherente, incluso en el caso de sistemas de archivos descendientes.

No se puede acceder directamente a las instantáneas de volúmenes, pero se pueden clonar, hacer copias de seguridad, invertir, etc. Para obtener información sobre cómo hacer copias de seguridad de una instantánea ZFS, consulte "Envío y recepción de datos ZFS" en la página 228.

- "Creación y destrucción de instantáneas de ZFS" en la página 220
- "Visualización y acceso a instantáneas de ZFS" en la página 223
- "Restablecimiento de una instantánea ZFS" en la página 224

Creación y destrucción de instantáneas de ZFS

Las instantáneas se crean con el comando `zfs snapshot` o `zfs snap` que toma como único argumento el nombre de la instantánea que se va a crear. El nombre de las instantáneas se asigna de la forma siguiente:

filesystem@snapname
volume@snapname

El nombre de la instantánea debe cumplir los requisitos de denominación establecidos en "Requisitos de asignación de nombres de componentes de ZFS" en la página 32.

En el siguiente ejemplo, se crea una instantánea de `tank/home/cindy` denominada `friday`.

```
# zfs snapshot tank/home/cindy@friday
```

Puede crear instantáneas de todos los sistemas de archivos descendientes con la opción `-r`. Por ejemplo:

```
# zfs snapshot -r tank/home@snap1
# zfs list -t snapshot -r tank/home
NAME                   USED  AVAIL  REFER  MOUNTPOINT
tank/home@snap1           0      -  2.11G  -
tank/home/cindy@snap1     0      -   115M  -
tank/home/lori@snap1      0      -  2.00G  -
tank/home/mark@snap1      0      -  2.00G  -
tank/home/tim@snap1       0      -  57.3M  -
```

Las instantáneas no tienen propiedades modificables. Las propiedades de conjuntos de datos no se pueden aplicar a una instantánea. Por ejemplo:

```
# zfs set compression=on tank/home/cindy@friday
cannot set property for 'tank/home/cindy@friday':
this property can not be modified for snapshots
```

Para destruir instantáneas se utiliza el comando `zfs destroy`. Por ejemplo:

```
# zfs destroy tank/home/cindy@friday
```

Los conjuntos de datos no se pueden destruir si tienen una instantánea. Por ejemplo:

```
# zfs destroy tank/home/cindy
cannot destroy 'tank/home/cindy': filesystem has children
use '-r' to destroy the following datasets:
tank/home/cindy@tuesday
tank/home/cindy@wednesday
tank/home/cindy@thursday
```

Además, si se han creado clones a partir de una instantánea, deben destruirse antes de poder destruir la instantánea.

Para obtener más información sobre el subcomando `destroy`, consulte "Destrucción de un sistema de archivos ZFS" en la página 149.

Conservación de instantáneas de ZFS

Si se implementan diferentes directivas de instantáneas automáticas de manera que `zfs receive` destruye accidentalmente las instantáneas más antiguas porque ya no existen en la parte remitente, debería considerar el uso de la función de conservación de instantáneas.

La función de *conservación* impide que se destruya una instantánea. Además, esta función permite que una instantánea con clones se suprima en espera de la eliminación del último clon mediante el comando `zfs destroy -d`. Cada instantánea tiene asociado un número de referencia de usuario, que se inicializa a cero. Este recuento aumenta una unidad cuando se aplica una retención a una instantánea y disminuye una unidad cuando se libera una retención.

En la versión anterior de Oracle Solaris, sólo era posible destruir una instantánea mediante el comando `zfs destroy` si ésta no tenía clones. En esta versión de Oracle Solaris, la instantánea también debe tener un recuento de referencia de usuario cero.

Se puede aplicar la función de conservación a una instantánea o a un conjunto de ellas. Por ejemplo, la siguiente sintaxis coloca una etiqueta de retención, `keep`, en `tank/home/cindy/snap@1`:

```
# zfs hold keep tank/home/cindy@snap1
```

Puede utilizar la opción `-r` para conservar las instantáneas de todos los sistemas de archivos descendientes. Por ejemplo:

```
# zfs snapshot -r tank/home@now
# zfs hold -r keep tank/home@now
```

Esta sintaxis agrega una sola referencia, `keep`, a la instantánea o al conjunto de instantáneas. Cada instantánea tiene su propio espacio de nombre de etiqueta y las etiquetas de conservación deben ser exclusivas dentro de ese espacio. Si se ha aplicado la función de conservación a una instantánea, fallará cualquier intento de destruirla mediante el comando `zfs destroy`. Por ejemplo:

```
# zfs destroy tank/home/cindy@snap1
cannot destroy 'tank/home/cindy@snap1': dataset is busy
```

Para destruir una instantánea retenida, use la opción `-d`. Por ejemplo:

```
# zfs destroy -d tank/home/cindy@snap1
```

Utilice el comando `zfs holds` para ver una lista de instantáneas conservadas. Por ejemplo:

```
# zfs holds tank/home@now
NAME           TAG   TIMESTAMP
tank/home@now  keep  Fri Aug  3 15:15:53 2012

# zfs holds -r tank/home@now
NAME                 TAG   TIMESTAMP
tank/home/cindy@now  keep  Fri Aug  3 15:15:53 2012
tank/home/lori@now   keep  Fri Aug  3 15:15:53 2012
tank/home/mark@now   keep  Fri Aug  3 15:15:53 2012
tank/home/tim@now    keep  Fri Aug  3 15:15:53 2012
tank/home@now        keep  Fri Aug  3 15:15:53 2012
```

Puede utilizar el comando `zfs release` para eliminar la conservación de una instantánea o de un conjunto de instantáneas. Por ejemplo:

```
# zfs release -r keep tank/home@now
```

Si la instantánea se libera, se podrá destruir mediante el comando `zfs destroy`. Por ejemplo:

```
# zfs destroy -r tank/home@now
```

Hay dos nuevas propiedades que identifican la información de retención de instantánea.

- La propiedad `defer_destroy` está `activada` si la instantánea se ha marcado para su destrucción posteriormente, mediante el comando `zfs destroy -d`. De lo contrario, la propiedad está `desactivada`.
- La propiedad `userrefs` indica el número de retenciones de esta instantánea, también denominado recuento de *referencia de usuario*.

Cambio de nombre de instantáneas de ZFS

Se puede cambiar el nombre de las instantáneas, pero debe hacerse en la agrupación y el conjunto de datos en que se crearon. Por ejemplo:

```
# zfs rename tank/home/cindy@snap1 tank/home/cindy@today
```

Además, la siguiente sintaxis de acceso directo es equivalente a la sintaxis anterior:

```
# zfs rename tank/home/cindy@snap1 today
```

La siguiente operación de `cambio de nombre` de instantánea no es posible porque los nombres del sistema de archivos y la agrupación de destino no coinciden con los del sistema de archivos y la agrupación a partir de los cuales se creó la instantánea:

```
# zfs rename tank/home/cindy@today pool/home/cindy@saturday
cannot rename to 'pool/home/cindy@today': snapshots must be part of same
dataset
```

El comando `zfs rename -r` permite cambiar el nombre de instantáneas de forma recursiva. Por ejemplo:

```
# zfs list -t snapshot -r users/home
NAME                        USED  AVAIL  REFER  MOUNTPOINT
users/home@now             23.5K      -  35.5K  -
users/home@yesterday           0      -    38K  -
users/home/lori@yesterday      0      -  2.00G  -
users/home/mark@yesterday      0      -  1.00G  -
users/home/neil@yesterday      0      -  2.00G  -
# zfs rename -r users/home@yesterday @2daysago
# zfs list -t snapshot -r users/home
NAME                        USED  AVAIL  REFER  MOUNTPOINT
users/home@now             23.5K      -  35.5K  -
users/home@2daysago            0      -    38K  -
users/home/lori@2daysago       0      -  2.00G  -
users/home/mark@2daysago       0      -  1.00G  -
users/home/neil@2daysago       0      -  2.00G  -
```

Visualización y acceso a instantáneas de ZFS

De forma predeterminada, las instantáneas dejan de aparecer en la salida de `zfs list`. Debe utilizar el comando `zfs list -t snapshot` para visualizar información de la instantánea. O bien, active la propiedad de agrupación `listsnapshots`. Por ejemplo:

```
# zpool get listsnapshots tank
NAME  PROPERTY       VALUE      SOURCE
tank  listsnapshots  off        default
# zpool set listsnapshots=on tank
# zpool get listsnapshots tank
NAME  PROPERTY       VALUE      SOURCE
tank  listsnapshots  on         local
```

Se puede acceder a instantáneas de sistemas de archivos del directorio `.zfs/snapshot` en la raíz del sistema de archivos. Por ejemplo, si `tank/home/cindy` se monta en `/home/cindy`, se puede acceder a los datos de la instantánea `tank/home/cindy@thursday` en el directorio `/home/cindy/.zfs/snapshot/thursday` .

```
# ls /tank/home/cindy/.zfs/snapshot
thursday   tuesday    wednesday
```

Se puede obtener una lista de instantáneas de la forma que se indica a continuación:

```
# zfs list -t snapshot -r tank/home
NAME                       USED  AVAIL  REFER  MOUNTPOINT
tank/home/cindy@tuesday     45K      -  2.11G  -
tank/home/cindy@wednesday   45K      -  2.11G  -
tank/home/cindy@thursday      0      -  2.17G  -
```

Se puede obtener una lista de las instantáneas creadas para un determinado sistema de archivos de la forma siguiente:

```
# zfs list -r -t snapshot -o name,creation tank/home
NAME                          CREATION
tank/home/cindy@tuesday       Fri Aug  3 15:18 2012
tank/home/cindy@wednesday     Fri Aug  3 15:19 2012
tank/home/cindy@thursday      Fri Aug  3 15:19 2012
tank/home/lori@today          Fri Aug  3 15:24 2012
tank/home/mark@today          Fri Aug  3 15:24 2012
```

Cálculo del espacio para instantáneas de ZFS

Cuando se crea una instantánea, al principio comparte el espacio con el sistema de archivos y, posiblemente, con instantáneas antiguas. A medida que cambia el sistema de archivos, el espacio en el disco compartido inicialmente se convierte en exclusivo de la instantánea, cosa que se contabiliza como tal en la propiedad `used`. Si se suprimen instantáneas, puede aumentarse la cantidad de espacio exclusivo destinado a (*usado* por) otras instantáneas.

Un valor de propiedad `de referencia` de espacio de instantánea es el mismo que el del sistema de archivos cuando se creó la propiedad.

Puede identificar información adicional sobre el consumo de valores de la propiedad `used`. Las nuevas propiedades del sistema de archivos de sólo lectura describen el uso de espacio en el disco de clones, sistemas de archivos y volúmenes. Por ejemplo:

```
$ zfs list -o space -r rpool
NAME                     AVAIL   USED  USEDSNAP  USEDDS  USEDREFRESERV  USEDCHILD
rpool                      124G  9.57G         0    302K              0      9.57G
rpool/ROOT                 124G  3.38G         0     31K              0      3.38G
rpool/ROOT/solaris         124G  20.5K         0       0              0      20.5K
rpool/ROOT/solaris/var     124G  20.5K         0   20.5K              0          0
rpool/ROOT/solaris-1       124G  3.38G     66.3M   3.14G              0       184M
rpool/ROOT/solaris-1/var   124G   184M     49.9M    134M              0          0
rpool/VARSHARE             124G  39.5K         0   39.5K              0          0
rpool/dump                 124G  4.12G         0   4.00G           129M          0
rpool/export               124G    63K         0     32K              0        31K
rpool/export/home          124G    31K         0     31K              0          0
rpool/swap                 124G  2.06G         0   2.00G          64.7M          0
```

Para ver una descripción de estas propiedades, consulte la Tabla 5–1.

Restablecimiento de una instantánea ZFS

Puede usar el comando `zfs rollback` para anular todos los cambios efectuados en un sistema de archivos desde que se creó una instantánea concreta. El sistema de archivos vuelve al estado en que se encontraba en el momento de realizarse la instantánea. De forma predeterminada, el comando no puede restablecer una instantánea que no sea la más reciente.

Para restablecer una instantánea anterior, hay que destruir todas las instantáneas intermedias. Puede destruir versiones anteriores de instantáneas mediante la opción `-r`.

Si una instantánea intermedia tiene clones, para destruirlos debe especificarse la opción `-R`.

Nota – El sistema de archivos que se desea restaurar se desmonta y se vuelve a montar, si actualmente está montado. Si el sistema de archivos no se puede desmontar, la restauración falla. La opción `-f` hace que se desmonte el sistema de archivos, si es necesario.

En el siguiente ejemplo, el sistema de archivos `tank/home/cindy` se restaura a la instantánea `tuesday`:

```
# zfs rollback tank/home/cindy@tuesday
cannot rollback to 'tank/home/cindy@tuesday': more recent snapshots exist
use '-r' to force deletion of the following snapshots:
tank/home/cindy@wednesday
tank/home/cindy@thursday
# zfs rollback -r tank/home/cindy@tuesday
```

En este ejemplo, las instantáneas de `wednesday` y `thursday` se destruyen porque se ha restaurado la instantánea de `tuesday`.

```
# zfs list -r -t snapshot -o name,creation tank/home/cindy
NAME                     CREATION
tank/home/cindy@tuesday  Fri Aug  3 15:18 2012
```

Identificación de diferencias entre instantáneas de ZFS (`zfs diff`)

Puede determinar las diferencias entre instantáneas de ZFS mediante el comando `zfs diff`.

Por ejemplo, considere que se crean las siguientes dos instantáneas:

```
$ ls /tank/home/tim
fileA
$ zfs snapshot tank/home/tim@snap1
$ ls /tank/home/tim
fileA  fileB
$ zfs snapshot tank/home/tim@snap2
```

Por ejemplo, para identificar las diferencias que existen entre dos instantáneas, utilice una sintaxis similar a la siguiente:

```
$ zfs diff tank/home/tim@snap1 tank/home/tim@snap2
M       /tank/home/tim/
+       /tank/home/tim/fileB
```

En la salida anterior, `M` indica que el directorio se ha modificado. El símbolo + indica que `fileB` existe en la instantánea posterior.

La `M` en la siguiente salida indica que se ha cambiado el nombre de un archivo en una instantánea.

```
$ mv /tank/cindy/fileB /tank/cindy/fileC
$ zfs snapshot tank/cindy@snap2
$ zfs diff tank/cindy@snap1 tank/cindy@snap2
M       /tank/cindy/
R       /tank/cindy/fileB -> /tank/cindy/fileC
```

En la siguiente tabla se resumen los cambios de archivo o directorio identificados mediante el comando `zfs diff`.

Cambio de archivo o directorio	Identificador
Se ha modificado un archivo o directorio o ha cambiado un enlace de archivo o directorio	`M`
Un archivo o directorio está presente en la instantánea antigua pero no en la instantánea más reciente	–
Un archivo o directorio está presente en la instantánea más reciente pero no en la instantánea antigua	+
Se ha cambiado el nombre de un archivo o directorio	`R`

Para obtener más información, consulte `zfs`(1M).

Información general sobre clones de ZFS

Un *clon* consiste en un volumen grabable o sistema de archivos cuyo contenido inicial es el mismo que el del conjunto de datos a partir del cual se ha creado. Al igual que sucede con las instantáneas, un clon se crea casi inmediatamente y al principio no consume espacio en el disco adicional. Además, se puede obtener una instantánea de un clónico.

Los clones sólo se pueden crear a partir de una instantánea. Al clonarse una instantánea, se crea una dependencia implícita entre ésta y el clon. Aunque el clon se cree en alguna otra parte de la jerarquía del sistema de archivos, la instantánea original no se podrá destruir mientras exista el clon. La propiedad `origin` muestra esta dependencia y el comando `zfs destroy` recopila todas estas dependencias, si las hay.

Los clones no heredan las dependencias del conjunto de datos a partir del que se crean. Utilice los comandos `zfs get` y `zfs set` para ver y cambiar las propiedades de un conjunto de datos clonado. Para obtener más información sobre el establecimiento de las propiedades de conjuntos de datos de ZFS, consulte “Configuración de propiedades de ZFS” en la página 178.

Debido a que al principio un clon comparte todo su espacio en el disco con la instantánea original, el valor de su propiedad `used` se establece inicialmente en cero. A medida que se efectúan cambios en el clon, consume más espacio en el disco. La propiedad `used` de la instantánea original no incluye el espacio que consume el clon en el disco.

- “Creación de un clon de ZFS” en la página 227

Creación de un clon de ZFS

Para crear un clon, utilice el comando `zfs clone`; especifique la instantánea a partir de la cual se va a crear, así como el nombre del nuevo volumen o sistema de archivos. El nuevo volumen o sistema de archivos se puede colocar en cualquier parte de la jerarquía de ZFS. El nuevo conjunto de datos es del mismo tipo (por ejemplo, volumen o sistema de archivos) que la instantánea a partir de la cual se ha creado el clon. El clon de un sistema de archivos no se puede crear en una agrupación que no sea donde se ubica la instantánea del sistema de archivos original.

En el siguiente ejemplo, se crea un nuevo clon denominado `tank/home/matt/bug123` con el mismo contenido inicial que la instantánea `tank/ws/gate@yesterday`:

```
# zfs snapshot tank/ws/gate@yesterday
# zfs clone tank/ws/gate@yesterday tank/home/matt/bug123
```

En este ejemplo, se crea un espacio de trabajo clónico a partir de la instantánea de `projects/newproject@today` para un usuario temporal denominado `projects/teamA/tempuser`. A continuación, las propiedades se establecen en el espacio de trabajo clónico.

```
# zfs snapshot projects/newproject@today
# zfs clone projects/newproject@today projects/teamA/tempuser
# zfs set share.nfs=on projects/teamA/tempuser
# zfs set quota=5G projects/teamA/tempuser
```

Destrucción de un clon de ZFS

Para destruir clones de ZFS se utiliza el comando `zfs destroy`. Por ejemplo:

```
# zfs destroy tank/home/matt/bug123
```

Para poder destruir la instantánea principal, antes hay que destruir los clones.

Sustitución de un sistema de archivos ZFS por un clon de ZFS

El comando `zfs promote` es apto para reemplazar un sistema de archivos ZFS activo por un clon de ese sistema de archivos. Esta función permite la clonación y sustitución de sistemas de archivos para que el sistema de archivos *original* se convierta en el clon del sistema de archivos

especificado. Asimismo, posibilita la destrucción del sistema de archivos a partir del cual se creó el clon. Sin la promoción de clones no es posible destruir un sistema de archivos original de clones activos. Para obtener más información sobre la destrucción de clones, consulte "Destrucción de un clon de ZFS" en la página 227.

En este ejemplo, se clona el sistema de archivos `tank/test/productA` y el sistema de archivos clónico, `tank/test/productAbeta`, se convierte en el sistema de archivos `tank/test/productA` original.

```
# zfs create tank/test
# zfs create tank/test/productA
# zfs snapshot tank/test/productA@today
# zfs clone tank/test/productA@today tank/test/productAbeta
# zfs list -r tank/test
NAME                        USED  AVAIL  REFER  MOUNTPOINT
tank/test                   104M  66.2G    23K  /tank/test
tank/test/productA          104M  66.2G   104M  /tank/test/productA
tank/test/productA@today       0      -   104M  -
tank/test/productAbeta         0  66.2G   104M  /tank/test/productAbeta
# zfs promote tank/test/productAbeta
# zfs list -r tank/test
NAME                           USED  AVAIL  REFER  MOUNTPOINT
tank/test                      104M  66.2G    24K  /tank/test
tank/test/productA                0  66.2G   104M  /tank/test/productA
tank/test/productAbeta         104M  66.2G   104M  /tank/test/productAbeta
tank/test/productAbeta@today      0      -   104M  -
```

En esta salida `zfs list` se ha sustituido la información de cálculo de espacio en el disco del sistema de archivos `productA` original por el sistema de archivos `productAbeta`.

Puede completar el proceso de sustitución de clones cambiando el nombre de los sistemas de archivos. Por ejemplo:

```
# zfs rename tank/test/productA tank/test/productAlegacy
# zfs rename tank/test/productAbeta tank/test/productA
# zfs list -r tank/test
```

Si lo desea, puede eliminar el sistema de archivos heredado. Por ejemplo:

```
# zfs destroy tank/test/productAlegacy
```

Envío y recepción de datos ZFS

El comando `zfs send` crea una representación de flujo de datos de una instantánea que se graba en una salida estándar. De forma predeterminada, se crea un flujo de datos completo. Puede redirigir la salida a un archivo u otro sistema. El comando `zfs receive` crea una instantánea cuyo contenido se especifica en el flujo de datos que figura en la entrada estándar. Si se recibe un flujo de datos completo, también se crea un sistema de archivos. Con estos comandos puede enviar y recibir datos de instantáneas ZFS y sistemas de archivos. Consulte los ejemplos de la sección siguiente.

Para guardar datos ZFS existen las soluciones de copia de seguridad siguientes:

- **Productos empresariales de copia de seguridad**: si necesita las siguientes funciones, considere la opción de una solución empresarial de copia de seguridad:
  - Restauración por archivo
  - Verificación de soportes de copia de seguridad
  - Administración de soportes
- **Instantáneas de sistemas de archivos y restauración de instantáneas**: utilice los comandos `zfs snapshot` y `zfs rollback` para crear con facilidad una copia de un sistema de archivos y restablecer su versión anterior si es preciso. Esta solución es válida, por ejemplo, para restaurar uno o varios archivos de una versión anterior.

  Para obtener más información sobre cómo crear y restaurar una versión de instantánea, consulte "Información general de instantáneas de ZFS" en la página 219.
- **Guardar instantáneas**: utilice los comandos `zfs send` y `zfs receive` para enviar y recibir una instantánea ZFS. Puede guardar cambios incrementales entre instantáneas, pero no puede restaurar archivos de manera individual. Es preciso restaurar toda la instantánea del sistema de archivos. Estos comandos no constituyen una solución de copia de seguridad completa para guardar los datos de ZFS.
- **Repetición remota**: utilice los comandos `zfs send` y `zfs receive` para copiar un sistema de archivos de un sistema a otro. Este proceso difiere del tradicional producto para la administración de volúmenes que quizá refleje dispositivos a través de una WAN. No se necesita ninguna clase de configuración ni hardware especiales. La ventaja de replicar un sistema de archivos ZFS es poder volver a crear un sistema de archivos de un grupo de almacenamiento en otro sistema y especificar distintos niveles de configuración de ese nuevo conjunto, por ejemplo RAID-Z, pero con los mismos datos del sistema de archivos.
- **Utilidades de archivado**: guarde datos de ZFS con utilidades de archivado como `tar`, `cpio` y `pax`, o productos de copia de seguridad de otros proveedores. Actualmente, tanto `tar` como `cpio` traducen correctamente las listas ACL, pero no ocurre lo mismo con `pax`.

Cómo guardar datos de ZFS con otros productos de copia de seguridad

Aparte de los comandos `zfs send` y `zfs receive`, para guardar archivos ZFS también son aptas utilidades de archivado como los comandos `tar` y `cpio`. Estas utilidades permiten guardar y restaurar atributos de archivos ZFS y ACL. Seleccione las opciones correspondientes para los comandos `tar` y `cpio`.

Para obtener información actualizada sobre problemas con ZFS y productos de copia de seguridad de otros proveedores, consulte las Notas de la versión Oracle Solaris 11.

Identificación de flujos de instantáneas de ZFS

Una instantánea de un volumen o sistema de archivos ZFS se convierte en un flujo de instantáneas mediante el comando `zfs send`. Luego, puede utilizar el flujo de instantáneas para volver a crear un volumen o sistema de archivos ZFS mediante el comando `zfs receive`.

Según las opciones de `zfs send` que se han utilizado para crear el flujo de instantáneas, se generan distintos tipos de formatos de flujo.

- Flujo completo: consta de todos los contenidos de conjuntos de datos desde el momento en que se creó el conjunto de datos hasta la instantánea especificada.

  El flujo predeterminado generado por el comando `zfs send` es un flujo completo. Contiene un volumen o sistema de archivos, hasta la instantánea especificada, y la incluye. El flujo no contiene instantáneas distintas de la instantánea especificada en la línea de comandos.
- Flujo incremental: consta de las diferencias entre una instantánea y otra instantánea.

Un paquete de flujos es un tipo de flujo que contiene uno o varios flujos completos o incrementales. Existen tres tipos de paquetes de flujos:

- Paquete de flujos de replicación: consta del conjunto de datos especificado y sus descendientes. En él, se incluyen todas las instantáneas intermedias. Si el origen de un conjunto de datos clonado no es un descendiente de la instantánea especificada en la línea de comandos, ese conjunto de datos de origen no se incluye en el paquete de flujos. Para recibir el flujo, el conjunto de datos de origen debe existir en la agrupación de almacenamiento de destino.

  Considere la siguiente lista de conjuntos de datos y sus orígenes. Supongamos que se crearon en el orden en que aparecen a continuación.

```
NAME                    ORIGIN
pool/a                  -
pool/a/1                -
pool/a/1@clone          -
pool/b                  -
pool/b/1                pool/a/1@clone
```

```
pool/b/1@clone2         -
pool/b/2                pool/b/1@clone2
pool/b@pre-send         -
pool/b/1@pre-send       -
pool/b/2@pre-send       -
pool/b@send             -
pool/b/1@send           -
pool/b/2@send           -
```

Un paquete de flujos de replicación que se crea con la siguiente sintaxis:

```
# zfs send -R pool/b@send ....
```

Consta de los siguientes flujos completos e incrementales:

```
TYPE    SNAPSHOT                INCREMENTAL FROM
full    pool/b@pre-send         -
incr    pool/b@send             pool/b@pre-send
incr    pool/b/1@clone2         pool/a/1@clone
incr    pool/b/1@pre-send       pool/b/1@clone2
incr    pool/b/1@send           pool/b/1@send
incr    pool/b/2@pre-send       pool/b/1@clone2
incr    pool/b/2@send           pool/b/2@pre-send
```

En la salida anterior, la instantánea `pool/a/1@clone` no se incluye en el paquete de flujos de replicación. Como tal, este paquete de flujos de replicación sólo se puede recibir en una agrupación que ya tiene la instantánea `pool/a/1@clone`.

- Paquete de flujos recursivos: consta del conjunto de datos especificado y sus descendientes. A diferencia de los paquetes de flujos de replicación, las instantáneas intermedias no se incluyen, a menos que sean el origen de un conjunto de datos clonado que se incluye en el flujo. De manera predeterminada, si el origen de un conjunto de datos no es un descendiente de la instantánea especificada en la línea de comandos, el comportamiento es similar a los flujos de replicación. Sin embargo, un flujo recursivo autocontenido (descrito a continuación) se crea de tal manera que no hay dependencias externas.

  Un paquete de flujos recursivos que se crea con la siguiente sintaxis:

  ```
  # zfs send -r pool/b@send ...
  ```

  Consta de los siguientes flujos completos e incrementales:

  ```
  TYPE    SNAPSHOT                INCREMENTAL FROM
  full    pool/b@send             -
  incr    pool/b/1@clone2         pool/a/1@clone
  incr    pool/b/1@send           pool/b/1@clone2
  incr    pool/b/2@send           pool/b/1@clone2
  ```

  En la salida anterior, la instantánea `pool/a/1@clone` no se incluye en el paquete de flujos recursivos. Como tal, este paquete de flujos recursivos sólo se puede recibir en una agrupación que ya tiene la instantánea `pool/a/1@clone`. Este comportamiento es similar al escenario de paquetes de flujos de replicación descritos anteriormente.

- Paquete de flujos recursivos autocontenido: no depende de ningún conjunto de datos que no esté incluido en el paquete de flujos. Este paquete de flujos recursivos se crea con la siguiente sintaxis:

```
# zfs send -rc pool/b@send ...
```

Consta de los siguientes flujos completos e incrementales:

```
TYPE    SNAPSHOT                INCREMENTAL FROM
full    pool/b@send             -
full    pool/b/1@clone2
incr    pool/b/1@send           pool/b/1@clone2
incr    pool/b/2@send           pool/b/1@clone2
```

Tenga en cuenta que el flujo recursivo autocontenido tiene un flujo completo de la instantánea `pool/b/1@clone2`, lo que hace posible recibir la instantánea `pool/b/1` sin dependencias externas.

Envío de una instantánea ZFS

Puede utilizar el comando `zfs send` para enviar una copia de un flujo de instantáneas y recibirlo en otra agrupación del mismo sistema o en otra agrupación de un sistema diferente que se utiliza para almacenar datos de copia de seguridad. Por ejemplo, para enviar el flujo de instantáneas de otra agrupación al mismo sistema, utilice una sintaxis similar a la siguiente:

```
# zfs send tank/dana@snap1 | zfs recv spool/ds01
```

Puede utilizar `zfs recv` como alias para el comando `zfs receive`.

Si envía el flujo de instantáneas a otro sistema, utilice el comando `ssh` para enviar la salida `zfs send`. Por ejemplo:

```
sys1# zfs send tank/dana@snap1 | ssh sys2 zfs recv newtank/dana
```

Si se envía un flujo de datos completo, no debe existir el sistema de archivos de destino.

Los datos incrementales se pueden guardar con la opción `zfs send -i`. Por ejemplo:

```
sys1# zfs send -i tank/dana@snap1 tank/dana@snap2 | ssh sys2 zfs recv newtank/dana
```

El primer argumento (`snap1`) es la instantánea anterior, y el segundo (`snap2`), la instantánea posterior. En este caso, para que la recepción incremental sea posible, debe existir el sistema de archivos `newtank/dana`.

Nota – Si se accede a la información de archivos en el sistema de archivos original recibido, es posible que la operación de recepción de instantáneas incrementales falle y se muestre un mensaje similar al siguiente:

```
cannot receive incremental stream of tank/dana@snap2 into newtank/dana:
most recent snapshot of tank/dana@snap2 does not match incremental source
```

Considere la posibilidad de establecer la propiedad `atime` en off si necesita acceder a la información de archivos en el sistema de archivos original recibido o si también necesita recibir instantáneas incrementales en el sistema de archivos recibido.

El origen de *instantánea1* incremental se puede especificar como último componente del nombre de la instantánea. Este método abreviado significa que sólo se debe indicar el nombre después del signo de arroba `@` para *instantánea1*, que se supone que procede del mismo sistema de archivos que *instantánea2*. Por ejemplo:

```
sys1# zfs send -i snap1 tank/dana@snap2 | ssh sys2 zfs recv newtank/dana
```

Esta sintaxis de acceso directo es equivalente a la sintaxis incremental en el ejemplo anterior.

Si se intenta generar un flujo de datos incremental a partir de una `instantánea1` de otro sistema de archivos, aparece en pantalla el mensaje siguiente:

```
cannot send 'pool/fs@name': not an earlier snapshot from the same fs
```

Si tiene que almacenar muchas copias, puede ser conveniente comprimir una representación de flujos de datos de instantáneas de ZFS mediante el comando `gzip`. Por ejemplo:

```
# zfs send pool/fs@snap | gzip > backupfile.gz
```

Recepción de una instantánea ZFS

Al recibir una instantánea de sistema de archivos, debe tener en cuenta los aspectos siguientes:

- Se recibe tanto la instantánea como el sistema de archivos.
- Se desmontan el sistema de archivos y todos los sistemas de archivos subordinados.
- Mientras se efectúa la recepción, no es posible acceder a los sistemas de archivos.
- El sistema de archivos original que se va a recibir no debe existir mientras se transfiere.
- Si el nombre del sistema de archivos ya existe, puede utilizar el comando `zfs rename` para cambiar el nombre del sistema de archivos.

Por ejemplo:

```
# zfs send tank/gozer@0830 > /bkups/gozer.083006
# zfs receive tank/gozer2@today < /bkups/gozer.083006
# zfs rename tank/gozer tank/gozer.old
# zfs rename tank/gozer2 tank/gozer
```

Si realiza un cambio en el sistema de archivos de destino y quiere efectuar otro envío incremental de una instantánea, antes debe restaurar el sistema de archivos receptor.

Considere el siguiente ejemplo. En primer lugar, efectúe un cambio como éste en el sistema de archivos:

```
sys2# rm newtank/dana/file.1
```

A continuación, realice un envío incremental de `tank/dana@snap3`. Pero antes debe restaurar la versión previa del sistema de archivos receptor para recibir la nueva instantánea incremental. O puede eliminar el paso de restauración usando la opción `-F`. Por ejemplo:

```
sys1# zfs send -i tank/dana@snap2 tank/dana@snap3 | ssh sys2 zfs recv -F newtank/dana
```

Al recibir una instantánea incremental, ya debe existir el sistema de archivos de destino.

Si efectúa cambios en el sistema de archivos y no restaura el sistema de archivos receptor para recibir la nueva instantánea incremental, o no utiliza la opción `-F`, verá una mensaje similar a éste:

```
sys1# zfs send -i tank/dana@snap4 tank/dana@snap5 | ssh sys2 zfs recv newtank/dana
cannot receive: destination has been modified since most recent snapshot
```

Para que la opción `-F` funcione debidamente, primero hay que efectuar estas comprobaciones:

- Si la instantánea más reciente no coincide con el origen incremental, no se completan la restauración ni la recepción, y se genera un mensaje de error.
- Si inadvertidamente se indica un nombre de sistema de archivos que no coincide con el origen incremental especificado en el comando `zfs receive`, no se completan la restauración ni la recepción, y se genera el siguiente mensaje de error:

  ```
  cannot send 'pool/fs@name': not an earlier snapshot from the same fs
  ```

Aplicación de valores de propiedad diferentes a un flujo de instantáneas de ZFS

Puede enviar un flujo de instantáneas de ZFS con un valor de propiedad de sistema de archivos determinado, pero puede especificar un valor de propiedad local diferente cuando recibe el flujo de instantáneas. También puede especificar que se utilice el valor de propiedad original al recibir el flujo de instantáneas para volver a crear el sistema de archivos original. Además, puede desactivar una propiedad del sistema de archivos al recibir el flujo de instantáneas.

- Utilice `zfs inherit -S` para restablecer un valor de propiedad local al valor recibido, si lo hubiera. Si una propiedad no tiene un valor recibido, el comportamiento del comando `zfs inherit - S` es el mismo que el comando `zfs inherit` sin la opción `-S`. Si la propiedad no tiene un valor recibido, el comando `zfs inherit` enmascara el valor recibido con el valor heredado hasta que la emisión de un comando `zfs inherit -S` lo restablece al valor recibido.
- Puede utilizar `zfs get -o` para incluir la nueva columna `RECEIVED` no predeterminada. O bien, utilice el comando `zfs get -o all` para incluir todas las columnas, incluida `RECEIVED`.
- Puede utilizar la opción `zfs send -p` para incluir las propiedades en el flujo de envío sin la opción `-R`.
- Puede utilizar la opción `zfs receive -e` para utilizar el último elemento del nombre de instantánea enviado para determinar el nuevo nombre de instantánea. El ejemplo siguiente envía la instantánea `poola/bee/cee@1` al sistema `poold/eee` y sólo utiliza el último elemento (`cee@1`) del nombre de la instantánea para crear el sistema y la instantánea del archivo recibido.

```
# zfs list -rt all poola
NAME              USED  AVAIL  REFER  MOUNTPOINT
poola             134K   134G    23K  /poola
poola/bee          44K   134G    23K  /poola/bee
poola/bee/cee      21K   134G    21K  /poola/bee/cee
poola/bee/cee@1      0      -    21K  -
# zfs send -R poola/bee/cee@1 | zfs receive -e poold/eee
# zfs list -rt all poold
NAME              USED  AVAIL  REFER  MOUNTPOINT
poold             134K   134G    23K  /poold
poold/eee          44K   134G    23K  /poold/eee
poold/eee/cee      21K   134G    21K  /poold/eee/cee
poold/eee/cee@1      0      -    21K  -
```

En algunos casos, es posible que las propiedades del sistema de archivos de un flujo de envío no se apliquen al sistema de archivos receptor o que las propiedades del sistema de archivos local, como el valor de propiedad `mountpoint`, interfieran con una restauración.

Por ejemplo, el sistema de archivos `tank/data` tiene la propiedad `compression` desactivada. Una instantánea del sistema de archivos `tank/data` se envía con propiedades (opción `-p`) a una agrupación de seguridad y es recibida con la propiedad `compression` activada.

```
# zfs get compression tank/data
NAME       PROPERTY     VALUE     SOURCE
tank/data  compression  off       default
# zfs snapshot tank/data@snap1
# zfs send -p tank/data@snap1 | zfs recv -o compression=on -d bpool
# zfs get -o all compression bpool/data
NAME        PROPERTY     VALUE     RECEIVED  SOURCE
bpool/data  compression  on        off       local
```

En el ejemplo, la propiedad `compression` está activada cuando se recibe la instantánea en `bpool`. Por lo tanto, para `bpool/data`, el valor `compression` está activado.

Si este flujo de instantáneas se envía a una nueva agrupación, `restorepool`, para fines de recuperación, es posible que desee mantener todas las propiedades originales de las instantáneas. En este caso, debe utilizar el comando `zfs send -b` para restaurar las propiedades originales de las instantáneas. Por ejemplo:

```
# zfs send -b bpool/data@snap1 | zfs recv -d restorepool
# zfs get -o all compression restorepool/data
NAME              PROPERTY     VALUE     RECEIVED  SOURCE
restorepool/data  compression  off       off       received
```

En el ejemplo, el valor de compresión es `off`, que representa el valor de compresión de la instantánea del sistema de archivos original `tank/data`.

Si tiene un valor de propiedad de sistema de archivos local en un flujo de instantáneas y desea desactivar la propiedad cuando lo reciba, utilice el comando `zfs receive -x`. Por ejemplo, el siguiente comando envía un flujo de instantáneas recursivas de los sistemas de archivos de directorios principales con todas las propiedades del sistema de archivos reservadas para una agrupación de seguridad, pero sin valores de propiedad de cuota:

```
# zfs send -R tank/home@snap1 | zfs recv -x quota bpool/home
# zfs get -r quota bpool/home
NAME                   PROPERTY  VALUE  SOURCE
bpool/home             quota     none   local
bpool/home@snap1       quota     -      -
bpool/home/lori        quota     none   default
bpool/home/lori@snap1  quota     -      -
bpool/home/mark        quota     none   default
bpool/home/mark@snap1  quota     -      -
```

Si la instantánea recursiva no se recibe con la opción `-x`, la propiedad de cuota se establecerá en los sistemas de archivos recibidos.

```
# zfs send -R tank/home@snap1 | zfs recv bpool/home
# zfs get -r quota bpool/home
NAME                   PROPERTY  VALUE  SOURCE
bpool/home             quota     none   received
bpool/home@snap1       quota     -      -
bpool/home/lori        quota     10G    received
bpool/home/lori@snap1  quota     -      -
bpool/home/mark        quota     10G    received
bpool/home/mark@snap1  quota     -      -
```

Envío y recepción de flujos de instantáneas ZFS complejos

En esta sección se describe cómo utilizar las opciones `zfs send -I` y `-R` para enviar y recibir flujos de instantáneas más complejos.

Al enviar y recibir flujos de instantáneas ZFS complejos, tenga en cuenta los puntos siguientes:

- Utilice la opción `zfs send -I` para enviar todos los flujos incrementales de una instantánea a una instantánea acumulativa. Utilice también esta opción para enviar un flujo incremental de la instantánea original para crear un clon. Para que se acepte el flujo incremental, la instantánea original ya debe estar en la parte receptora.
- Utilice la opción `zfs send -R` para enviar un flujo de replicación de todos los sistemas de archivos descendentes. Cuando se recibe el flujo de repetición, se conservan todas las propiedades, las instantáneas, los sistemas de archivos descendientes y los duplicados.
- Cuando se utiliza la opción `zfs send -r` sin la opción `-c`, y cuando se utiliza la opción `zfs send -R`, los paquetes de flujos omiten el `origin` de los clones en algunas circunstancias. Para obtener más información, consulte "Identificación de flujos de instantáneas de ZFS" en la página 230.
- Utilice ambas opciones para enviar un flujo de repetición incremental.
  - Se mantienen los cambios de propiedades y también las operaciones `rename` y `destroy` de instantáneas y sistemas de archivos.
  - Si no se especifica `zfs recv -F` al recibir el flujo de repetición, se omiten las operaciones `destroy` de conjuntos de datos. La sintaxis `zfs recv -F` en este caso también mantiene su propiedad de aplicar *rollback (inversión) si es preciso*.
  - Al igual que en otros casos (que no sean `zfs send -R`) `-i` o `-I`, si se utiliza `-I`, se envían todas las instantáneas entre `snapA` y `snapD`. Si se utiliza `-i`, sólo se envía `snapD` (para todos los descendientes).
- Para recibir cualquiera de estos nuevos tipos de flujos `zfs send`, el sistema receptor debe ejecutar una versión del software capaz de enviarlos. La versión del flujo se incrementa.

  Sin embargo, puede acceder a los flujos desde versiones de agrupaciones más antiguas utilizando una versión del software más reciente. Por ejemplo, puede enviar y recibir flujos creados con las opciones más recientes a o desde una agrupación de la versión 3. Sin embargo, debe ejecutar software reciente para recibir un flujo enviado con las opciones más recientes.

EJEMPLO 6–1 Envío y recepción de flujos de instantáneas ZFS complejos

Un grupo de instantáneas incrementales se puede combinar en una instantánea utilizando la opción `zfs send -I`. Por ejemplo:

```
# zfs send -I pool/fs@snapA pool/fs@snapD > /snaps/fs@all-I
```

Luego deberá eliminar `snapB`, `snapC` y `snapD`.

```
# zfs destroy pool/fs@snapB
# zfs destroy pool/fs@snapC
# zfs destroy pool/fs@snapD
```

Para recibir la instantánea combinada, use el siguiente comando.

EJEMPLO 6–1 Envío y recepción de flujos de instantáneas ZFS complejos *(Continuación)*

```
# zfs receive -d -F pool/fs < /snaps/fs@all-I
# zfs list
NAME                      USED  AVAIL  REFER  MOUNTPOINT
pool                      428K  16.5G    20K  /pool
pool/fs                    71K  16.5G    21K  /pool/fs
pool/fs@snapA              16K      -  18.5K  -
pool/fs@snapB              17K      -    20K  -
pool/fs@snapC              17K      -  20.5K  -
pool/fs@snapD                0      -    21K  -
```

También puede utilizar el comando `zfs send -I` para combinar una instantánea y una instantánea clónica para crear un conjunto de datos combinado. Por ejemplo:

```
# zfs create pool/fs
# zfs snapshot pool/fs@snap1
# zfs clone pool/fs@snap1 pool/clone
# zfs snapshot pool/clone@snapA
# zfs send -I pool/fs@snap1 pool/clone@snapA > /snaps/fsclonesnap-I
# zfs destroy pool/clone@snapA
# zfs destroy pool/clone
# zfs receive -F pool/clone < /snaps/fsclonesnap-I
```

Puede utilizar el comando `zfs send -R` para repetir un sistema de archivos ZFS y todos los sistemas de archivos descendientes, hasta la instantánea en cuestión. Cuando se recibe este flujo, se conservan todas las propiedades, las instantáneas, los sistemas de archivos descendientes y los duplicados.

En el ejemplo siguiente, se crean instantáneas de los sistemas de archivos de usuario. Se crea un flujo de repetición de todas las instantáneas de usuario. A continuación, se destruyen y se recuperan las instantáneas y los sistemas de archivos originales.

```
# zfs snapshot -r users@today
# zfs list
NAME                USED  AVAIL  REFER  MOUNTPOINT
users               187K  33.2G    22K  /users
users@today            0      -    22K  -
users/user1         18K  33.2G    18K  /users/user1
users/user1@today      0      -    18K  -
users/user2         18K  33.2G    18K  /users/user2
users/user2@today      0      -    18K  -
users/user3         18K  33.2G    18K  /users/user3
users/user3@today      0      -    18K  -
# zfs send -R users@today > /snaps/users-R
# zfs destroy -r users
# zfs receive -F -d users < /snaps/users-R
# zfs list
NAME                USED  AVAIL  REFER  MOUNTPOINT
users               196K  33.2G    22K  /users
users@today            0      -    22K  -
users/user1         18K  33.2G    18K  /users/user1
users/user1@today      0      -    18K  -
users/user2         18K  33.2G    18K  /users/user2
```

EJEMPLO 6–1 Envío y recepción de flujos de instantáneas ZFS complejos *(Continuación)*

```
users/user2@today      0      -    18K  -
users/user3          18K  33.2G    18K  /users/user3
users/user3@today      0      -    18K  -
```

En el ejemplo siguiente, el comando `zfs send -R` se ha usado para replicar el sistema de archivos `users` y sus descendientes, y para enviar el flujo replicado a otra agrupación, `users2`.

```
# zfs create users2 mirror c0t1d0 c1t1d0
# zfs receive -F -d users2 < /snaps/users-R
# zfs list
NAME                  USED  AVAIL  REFER  MOUNTPOINT
users                 224K  33.2G    22K  /users
users@today              0      -    22K  -
users/user1           33K  33.2G    18K  /users/user1
users/user1@today     15K      -    18K  -
users/user2           18K  33.2G    18K  /users/user2
users/user2@today        0      -    18K  -
users/user3           18K  33.2G    18K  /users/user3
users/user3@today        0      -    18K  -
users2                188K  16.5G    22K  /users2
users2@today             0      -    22K  -
users2/user1          18K  16.5G    18K  /users2/user1
users2/user1@today       0      -    18K  -
users2/user2          18K  16.5G    18K  /users2/user2
users2/user2@today       0      -    18K  -
users2/user3          18K  16.5G    18K  /users2/user3
users2/user3@today       0      -    18K  -
```

Duplicación remota de datos de ZFS

Los comandos `zfs send` y `zfs recv` se utilizan para copiar de forma remota una representación de flujos de datos de instantánea de un sistema a otro. Por ejemplo:

```
# zfs send tank/cindy@today | ssh newsys zfs recv sandbox/restfs@today
```

Este comando envía los datos de instantánea `tank/cindy@today` y los recibe en el sistema de archivos `sandbox/restfs`. El comando también crea una instantánea `restfs@today` en el sistema `newsys`. En este ejemplo, se ha configurado al usuario para que utilice el comando `ssh` en el sistema remoto.

◆ ◆ ◆ CAPÍTULO 7

Uso de listas de control de acceso y atributos para proteger archivos Oracle Solaris ZFS

En este capítulo se proporciona información sobre el uso de las listas de control de acceso (ACL) para proteger los archivos ZFS ofreciendo más permisos granulares que los UNIX estándar.

Este capítulo se divide en las secciones siguientes:

Modelo de ACL de Solaris

Las versiones anteriores de Solaris admitían una implementación de listas de control de acceso (ACL) que se basaba principalmente en la especificación ACL de borrador POSIX. Estas clases de ACL se utilizan para proteger archivos UFS y se traducen de versiones de NFS anteriores a NFSv4.

NFSv4 es un nuevo modelo de ACL totalmente compatible que permite la interoperabilidad entre clientes UNIX y que no son UNIX. La nueva implementación de ACL, tal como se indica en las especificaciones de NFSv4, aporta una semántica mucho más rica que las ACL del tipo NT.

A continuación se exponen las diferencias principales del nuevo modelo de ACL:

- Se basa en la especificación de NFSv4 y se parece a las ACL del tipo NT.
- Proporciona un conjunto mucho más granular de privilegios de acceso. Para obtener más información, consulte la Tabla 7–2.

- Se define y visualiza con los comandos `chmod` y `ls`, en lugar de los comandos `setfacl` y `getfacl`.
- Aporta una semántica heredada mucho más rica para establecer la forma en que se aplican privilegios de acceso del directorio a los directorios, y así sucesivamente. Para obtener más información, consulte "Herencia de ACL" en la página 247.

Los modelos de ACL proporcionan un control de acceso mucho más granular que los permisos de archivos estándar. De forma muy parecida a las ACL de borrador POSIX, las nuevas ACL disponen de varias entradas de control de acceso.

Las ACL de borrador POSIX emplean una sola entrada para definir los permisos que se conceden y los que se deniegan. El nuevo modelo de ACL presenta dos clases de entradas de control de acceso que afectan a la comprobación de acceso: `ALLOW` y `DENY`. Así, a partir de una entrada de control de acceso que defina un conjunto de permisos no puede deducirse si se conceden o deniegan los permisos que hay definidos en dicha entrada.

La traducción entre las ACL del tipo NFSv4 y las de borrador POSIX se efectúa de la manera siguiente:

- Si emplea una utilidad que tiene en cuenta las ACL, por ejemplo los comandos `cp`, `mv`, `tar`, `cpio` o `rcp`, para transferir archivos UFS con ACL a un sistema de archivos ZFS, las ACL de borrador POSIX se traducen a sus equivalentes del tipo NFSv4.
- Algunas ACL de tipo NFSv4 se traducen a ACL de borrador POSIX. Si una ACL de tipo NFSv4 no se traduce a una ACL de borrador POSIX, en pantalla aparece un mensaje parecido al siguiente:

  ```
  # cp -p filea /var/tmp
  cp: failed to set acl entries on /var/tmp/filea
  ```
- Si crea un contenedor UFS `tar` o `cpio` con la opción de mantener las ACL (`tar -p` o `cpio -P`) en un sistema que ejecuta una versión actual de Solaris, las ACL se pierden si el contenedor se extrae a un sistema que ejecuta una versión inferior de Solaris.

  Se extraen todos los archivos con los modelos de archivos correctos, pero se omiten las entradas de ACL.
- El comando `ufsrestore` es apto para restaurar datos en un sistema de archivos ZFS. Si los datos originales incluyen ACL de tipo POSIX, se convierten a ACL de tipo NFSv4.
- Si intenta definir una ACL del tipo NFSv4 en un archivo UFS, en pantalla aparece un mensaje similar al siguiente:

  ```
  chmod: ERROR: ACL type's are different
  ```
- Si intenta definir una ACL de borrador POSIX en un archivo ZFS, en pantalla se muestran mensajes parecidos al siguiente:

  ```
  # getfacl filea
  File system doesn't support aclent_t style ACL's.
  See acl(5) for more information on Solaris ACL support.
  ```

Para obtener información sobre otras limitaciones con las ACL y demás productos para copias de seguridad, consulte "Cómo guardar datos de ZFS con otros productos de copia de seguridad" en la página 230.

Descripciones de la sintaxis para definir las ACL

Se proporcionan dos formatos básicos de ACL:

- **ACL trivial**: contiene únicamente entradas las entradas tradicionales de UNIX `user`, `group` y `owner`.
- **ACL no trivial**: contiene otras entradas además de owner, group y everyone, incluye conjuntos de indicadores heredados o las entradas están ordenadas de una manera no tradicional.

Sintaxis para definir ACL triviales

`chmod [options] A[index]{+|=}owner@ |group@ |everyone@:`
access-permissions`/...[:`*inheritance-flags*`]: deny | allow` *archivo*

`chmod [options] A-owner@, group@, everyone@:`*access-permissions*
`/...[:`*inheritance-flags*`]:deny | allow` *archivo* ...

`chmod [options] A[index]-` *archivo*

Sintaxis para definir ACL no triviales

`chmod [options] A[index]{+|=}user|group:name:`*access-permissions*
`/...[:`*inheritance-flags*`]:deny | allow` *archivo*

`chmod [options] A-user|group:name:`*access-permissions* `/...[:`*inheritance-flags*`]:deny | allow` *archivo* ...

`chmod [options] A[index]-` *archivo*

owner@, group@, everyone@
: Identifica el *tipo de entrada de ACL* de la sintaxis de ACL triviales. Para obtener una descripción de *tipos de entrada de ACL*, consulte la Tabla 7–1.

user or group:*ACL-entry-ID=username or groupname*
: Identifica el *tipo de entrada de ACL* de la sintaxis de ACL explícitas. El usuario y el grupo de *ACL-entry-type* deben contener también el *ACL-entry-ID*, *username* o *groupname*. Para obtener una descripción de *tipos de entrada de ACL*, consulte la Tabla 7–1.

access-permissions/.../
: Identifica los permisos de acceso que se conceden o deniegan. Para obtener una descripción de los permisos de acceso de ACL, consulte la Tabla 7–2.

inheritance-flags
Identifica una lista opcional de indicadores de herencia de ACL. Para obtener una descripción de los indicadores de herencia, consulte la Tabla 7–4.

`deny | allow`
Identifica si se conceden o deniegan los permisos de acceso.

En el siguiente ejemplo, no existe ningún valor de *ID de entrada de ACL* para owner@, group@ o everyone@.

```
group@:write_data/append_data/execute:deny
```

El ejemplo siguiente incluye un *ID de entrada LCA* porque en la LCA se incluye un usuario específico (*tipo de entrada LCA*).

```
0:user:gozer:list_directory/read_data/execute:allow
```

Cuando en pantalla se muestra una entrada de ACL, se parece al ejemplo siguiente:

```
2:group@:write_data/append_data/execute:deny
```

El **2** o el *ID de índice* de este ejemplo identifica la entrada de ACL de la ACL más grande, que podría tener varias entradas para owner (propietario), UID específicos, group (grupo) y everyone (cualquiera). Se puede especificar el *ID de índice* con el comando `chmod` para identificar la parte de la ACL que desea modificar. Por ejemplo, el ID de índice 3 puede identificarse como `A3` en el comando `chmod` de una forma similar a la siguiente:

```
chmod A3=user:venkman:read_acl:allow filename
```

Los tipos de entrada de ACL, que son las representaciones de ACL de los propietarios, grupos, etc., se describen en la tabla siguiente.

TABLA 7–1 Tipos de entrada de ACL

Tipo de entrada de ACL	Descripción
`owner@`	Especifica el acceso que se concede al propietario del objeto.
`group@`	Especifica el acceso que se concede al grupo propietario del objeto.
`everyone@`	Especifica el acceso que se concede a cualquier usuario o grupo que no coincida con ninguna otra entrada de ACL.
`user`	Con un nombre de usuario, especifica el acceso que se concede a un usuario adicional del objeto. Debe incluir el *ID de entrada de ACL*, que contiene un *username* o *userID*. Si el valor no es un ID de usuario numérico o *username* válido, el tipo de entrada de ACL tampoco es válido.

TABLA 7–1 Tipos de entrada de ACL *(Continuación)*

Tipo de entrada de ACL	Descripción
`group`	Con un nombre de grupo, especifica el acceso que se concede a un grupo adicional del objeto. Debe incluir el *ID de entrada de ACL*, que contiene un *groupname* o *groupID*. Si el valor no es un ID de grupo numérico o *groupname* válido, el tipo de entrada de ACL tampoco es válido.

En la tabla siguiente se describen los privilegios de acceso de LCA.

TABLA 7–2 Privilegios de acceso de ACL

Privilegio de acceso	Privilegio de acceso compacto	Descripción
`add_file`	`w`	Permiso para agregar un archivo nuevo a un directorio.
`add_subdirectory`	`p`	En un directorio, permiso para crear un subdirectorio.
`append_data`	`p`	Actualmente no se ha implementado.
`delete`	`d`	Permiso para suprimir un archivo. Para obtener más información sobre el comportamiento de permiso `delete` específico, consulte la Tabla 7–3.
`delete_child`	`D`	Permiso para suprimir un archivo o un directorio dentro de un directorio. Para obtener más información sobre el comportamiento de permiso `delete_child` específico, consulte la Tabla 7–3.
`execute`	`x`	Permiso para ejecutar un archivo o buscar en el contenido de un directorio.
`list_directory`	`r`	Permiso para resumir el contenido de un directorio.
`read_acl`	`c`	Permiso para leer la ACL (`ls`).
`read_attributes`	`a`	Permiso para leer los atributos básicos (no ACL) de un archivo. Los atributos de tipo stat pueden considerarse atributos básicos. Permitir este bit de la máscara de acceso significa que la entidad puede ejecutar `ls`(1) y `stat`(2).
`read_data`	`r`	Permiso para leer el contenido del archivo.
`read_xattr`	`R`	Permiso para leer los atributos extendidos de un archivo o al buscar en el directorio de atributos extendidos del archivo.
`synchronize`	`s`	Actualmente no se ha implementado.

TABLA 7–2 Privilegios de acceso de ACL *(Continuación)*

Privilegio de acceso	Privilegio de acceso compacto	Descripción
`write_xattr`	`W`	Permiso para crear atributos extendidos o escribir en el directorio de atributos extendidos. Si se concede este permiso a un usuario, el usuario puede crear un directorio de atributos extendidos para un archivo. Los permisos de atributo del archivo controlan el acceso al atributo por parte del usuario.
`write_data`	`w`	Permiso para modificar o reemplazar el contenido de un archivo.
`write_attributes`	`A`	Permiso para cambiar las horas asociadas con un archivo o directorio a un valor arbitrario.
`write_acl`	`C`	Permiso para escribir en la ACL o posibilidad de modificarla mediante el comando `chmod`.
`write_owner`	`o`	Permiso para cambiar el grupo o propietario del archivo. O posibilidad de ejecutar los comandos `chown` o `chgrp` en el archivo. Permiso para adquirir la propiedad de un archivo o para cambiar la propiedad del grupo del archivo a un grupo al que pertenezca el usuario. Si desea cambiar la propiedad de grupo o archivo a un usuario o grupo arbitrario, se necesita el privilegio `PRIV_FILE_CHOWN`.

La siguiente tabla proporciona información adicional sobre los comportamientos de ACL `delete` y `delete_child` .

TABLA 7–3 Comportamientos de permiso de ACL delete y delete_child

Permisos de directorio principal	Permisos de objeto de destino		
	ACL permite suprimir	ACL deniega suprimir	Permiso de supresión no especificado
ACL permite `delete_child`	Permitir	Permitir	Permitir
ACL deniega `delete_child`	Permitir	Denegar	Denegar
ACL permite solamente `write` y `execute`	Permitir	Permitir	Permitir
ACL deniega `write` y `execute`	Permitir	Denegar	Denegar

Conjuntos de LCA de ZFS

Las siguientes combinaciones de ACL se pueden aplicar en un *conjunto de ACL* en lugar de establecer permisos individuales por separado. Están disponibles los siguientes conjuntos de ACL.

Nombre de conjunto de ACL	Permisos de ACL incluidos
`full_set`	Todos los permisos
`modify_set`	Todos los permisos salvo `write_acl` y `write_owner`
`read_set`	`read_data`, `read_attributes`, `read_xattr` y `read_acl`
`write_set`	`write_data`, `append_data`, `write_attributes` y `write_xattr`

Estos conjuntos de ACL vienen predefinidos y no se pueden modificar.

Herencia de ACL

La finalidad de utilizar la herencia de LCA es que los archivos o directorios que se creen puedan heredar las LCA que en principio deben heredar, pero sin prescindir de los bits de permiso en el directorio superior.

De forma predeterminada, las LCA no se propagan. Si establece una ACL no trivial en un directorio, ésta no se heredará en ningún directorio posterior. Debe especificar la herencia de una LCA en un archivo o directorio.

En la tabla siguiente se describen los indicadores de herencia opcionales.

TABLA 7–4 Indicadores de herencia de ACL

Indicador de herencia	Indicador de herencia compacto	Descripción
`file_inherit`	`f`	La ACL sólo se hereda del directorio superior a los archivos del directorio.
`dir_inherit`	`d`	La ACL sólo se hereda del directorio superior a los subdirectorios del directorio.
`inherit_only`	`i`	La ACL se hereda del directorio superior, pero únicamente se aplica a los archivos y subdirectorios que se creen, no al directorio en sí. Para especificar lo que se hereda, se necesita el indicador `file_inherit`, `dir_inherit` o ambos.
`no_propagate`	`n`	La ACL se hereda sólo del directorio superior al contenido del primer nivel del directorio. Se excluye el contenido del segundo nivel o inferiores. Para especificar lo que se hereda, se necesita el indicador `file_inherit`, `dir_inherit` o ambos.
-	N/A	Ningún permiso concedido.
Actualmente, los siguientes indicadores se aplican solamente a un servidor o cliente de SMB.		

TABLA 7–4 Indicadores de herencia de ACL *(Continuación)*

Indicador de herencia	Indicador de herencia compacto	Descripción
`successful_access`	`S`	Indica si se debe iniciar una alarma o un registro de auditoría cuando un acceso es correcto. Este indicador se utiliza con tipos de ACE de auditoría o de alarma.
`failed_access`	`F`	Indica si se debe iniciar una alarma o un registro de auditoría cuando un acceso falla. Este indicador se utiliza con tipos de ACE de auditoría o de alarma.
`inherited`	`I`	Indica que se ha heredado una ACE.

Además, se puede establecer una directriz de herencia de ACL predeterminada del sistema de archivos más o menos estricta mediante la propiedad del sistema de archivos `aclinherit`. Para obtener más información, consulte la siguiente sección.

Propiedades de ACL

El sistema de archivos ZFS incluye las siguientes propiedades de ACL para determinar el comportamiento específico de herencia de ACL e interacción de ACL con operaciones `chmod`.

- `aclinherit`: determina el comportamiento de herencia de ACL. Entre los valores se incluyen los siguientes:
  - `discard`: en los objetos nuevos, si se crea un archivo o directorio, no se heredan entradas de LCA. La ACL del archivo o directorio es igual al modo de permiso del archivo o directorio.
  - `noallow`: en los objetos nuevos, sólo se heredan las entradas de LCA cuyo tipo de acceso sea `deny`.
  - `restricted`: en los objetos nuevos, al heredarse una entrada de ACL se eliminan los permisos `write_owner` y `write_acl`.
  - `passthrough`: si el valor de propiedad se configura como `passthrough`, los archivos se crean con un modo que determinan las entradas de control de acceso que se pueden heredar. Si no existen entradas de control de acceso que se puedan heredar y que afecten al modo, el modo se configurará de acuerdo con el modo solicitado desde la aplicación.
  - `Passthrough-x` : tiene la misma semántica que `passthrough`, excepto que cuando `passthrough-x` está activada, los archivos se crean con el permiso de ejecución (`x`), pero sólo si el permiso de ejecución se ha establecido en el modo de creación de archivos y en un ACE heredable que afecta al modo.

  El modo predeterminado para `aclinherit` es `restricted`.
- `aclmode`: modifica el comportamiento de ACL cuando un archivo se crea por primera vez o controla cómo se modifica una ACL durante una operación `chmod`. Puede tener los valores siguientes:

- `discard`: un sistema de archivos con una propiedad `aclmode` de `discard` suprime todas las entradas de ACL que no representan el modo del archivo. Éste es el valor predeterminado.
- `mask`: un sistema de archivos con una propiedad `aclmode` de `mask` reduce los permisos de usuario o de grupo. Se reducen los permisos para que no superen los bits de permisos de grupo, a menos que se trate de una entrada de usuario cuyo UID sea igual al del propietario del archivo o directorio. Así, los permisos de ACL se reducen para que no superen los bits de permisos del propietario. El valor de máscara también conserva la ACL cuando cambian los modos, siempre que no se haya realizado una operación de conjunto de ACL explícita.
- `passthrough`: un sistema de archivos con una propiedad `aclmode` de `passthrough` indica que no se realizaron más cambios en la ACL aparte de generar las entradas necesarias de ACL para representar el nuevo modo del archivo o del directorio.

`discard` es el modo predeterminado de la propiedad `aclmode`.

Para obtener más información sobre el uso de la propiedad `aclmode`, consulte el Ejemplo 7–14.

Establecimiento de las LCA en archivos ZFS

Al implementarse con ZFS, las ACL se componen de una matriz de entradas de ACL. ZFS proporciona un modelo de ACL *pura* en el que todos los archivos disponen de una ACL. En general, la LCA es *trivial* en el sentido de que sólo representa las entradas de UNIX `owner/group/other` tradicionales.

Los archivos ZFS siguen teniendo bits de permisos y un modo; sin embargo, estos valores son más de una caché de lo que representa la ACL. Así, si cambia los permisos del archivo, su LCA se actualiza en consonancia. Además, si elimina una ACL no trivial que concedía a un usuario acceso a un archivo o directorio, ese usuario quizá siga disponiendo de acceso gracias a los bits de permisos del archivo o directorio que conceden acceso al grupo o a todos los usuarios. Todas las decisiones de control de acceso se supeditan a los permisos representados en una LCA de archivo o directorio.

A continuación se proporcionan las reglas principales de acceso de ACL de un archivo ZFS:

- ZFS procesa entradas de ACL en el orden que figuran en la ACL, de arriba abajo.
- Sólo se procesan las entradas de ACL que tengan a "alguien" que coincida con quien solicita acceso.
- Una vez que se concede un permiso, una entrada de denegación de ACL posterior no lo puede denegar en el mismo conjunto de permisos de ACL.
- El permiso `write_acl` se concede de forma incondicional al propietario del archivo aunque el permiso se deniegue explícitamente. De lo contrario, se deniega cualquier permiso que no quede especificado.

Cuando se deniegan permisos o falta un permiso de acceso, el subsistema de privilegios determina la solicitud de acceso que se concede al propietario del archivo o superusuario. Es un mecanismo para permitir que los propietarios de archivos siempre puedan acceder a sus archivos y que los superusuarios puedan modificar archivos en situaciones de recuperación.

Si en un directorio se establece una ACL no trivial, los directorios secundarios no heredan la ACL de manera automática. Si se establece una ACL no trivial y desea que la hereden los directorios subordinados, debe utilizar los indicadores de herencia de ACL. Para obtener más información, consulte la Tabla 7–4 y "Establecimiento de herencia de LCA en archivos ZFS en formato detallado" en la página 257.

Al crear un archivo, y en función del valor `umask`, se aplica una ACL similar a la siguiente:

```
$ ls -v file.1
-rw-r--r--   1 root     root      206663 Jun 23 15:06 file.1
     0:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     1:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     2:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
         :allow
```

Cada categoría de usuario (`owner@`, `group@`, `everyone@`) tiene una entrada de ACL en este ejemplo.

A continuación se proporciona una descripción de esta LCA de archivo:

`0:owner@` El propietario puede leer y modificar el contenido del archivo (`read_data/write_data/append_data/read_xattr`). También puede modificar atributos del archivo, como indicaciones de hora, atributos extendidos y ACL (`write_xattr/read_attributes/write_attributes/read_acl/write_acl`). Además, puede modificar la propiedad del archivo (`write_owner:allow`).

El permiso de acceso `synchronize` no está implementado en la actualidad.

`1:group@` Se concede al grupo permisos de lectura del archivo y los atributos del archivo (`read_data/read_xattr/read_attributes/read_acl:allow`).

`2:everyone@` Se concede a quienes no sean usuario ni grupo permisos de lectura del archivo y los atributos del archivo (`read_data/read_xattr/read_attributes/read_acl/synchronize:allow`). El permiso de acceso `synchronize` no está implementado en la actualidad.

Si se crea un directorio, y según el valor de `umask`, una ACL de directorio predeterminada tendrá un aspecto similar al siguiente:

```
$ ls -dv dir.1
drwxr-xr-x   2 root     root           2 Jul 20 13:44 dir.1
     0:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     1:group@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
     2:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

A continuación se proporciona una descripción de esta ACL de directorio:

`0:owner@` El propietario puede leer y modificar el contenido del directorio (`list_directory/read_data/add_file/write_data/add_subdirectory /append_data`) y leer y modificar los atributos de un archivo, como los indicadores de horas, los atributos ampliados y las ACL (`/read_xattr/write_xattr/read_attributes/write_attributes/read_acl/ write_acl`). Además, el propietario puede buscar contenidos (`execute`), suprimir un archivo o un directorio (`delete_child`) y modificar la propiedad del directorio (`write_owner:allow`).

El permiso de acceso `synchronize` no está implementado en la actualidad.

`1:group@` El grupo puede mostrar y leer el contenido y los atributos del directorio. Además, tiene permisos de ejecución para buscar en el contenido del directorio (`list_directory/read_data/read_xattr/execute/read_attributes /read_acl/synchronize:allow`).

`2:everyone@` Se concede a quien no sea usuario ni grupo permisos de lectura y ejecución del contenido y los atributos del directorio (`list_directory/read_data/read_xattr/execute/read_ attributes/read_acl/synchronize:allow`). El permiso de acceso `synchronize` no está implementado en la actualidad.

Establecimiento y visualización de ACL en archivos ZFS en formato detallado

El comando `chmod` es válido para modificar las ACL de archivos ZFS. La sintaxis siguiente del comando `chmod` para modificar ACL utiliza *especificación ACL* para identificar el formato de la ACL. Para obtener una descripción de *especificación ACL*, consulte "Descripciones de la sintaxis para definir las ACL" en la página 243.

- Agregación de entradas de ACL
  - Agregación de una entrada de ACL para un usuario

```
% chmod A+acl-specification filename
```

- Agregación de una entrada de ACL mediante *index-ID*

  ```
  % chmod Aindex-ID+acl-specification filename
  ```

  Esta sintaxis inserta la nueva entrada de ACL en la ubicación de *index-ID* que se especifica.

- Sustitución de una entrada de ACL

  ```
  % chmod A=acl-specification filename
  % chmod Aindex-ID=acl-specification filename
  ```

- Eliminación de entradas de ACL
  - Eliminación de una entrada de ACL mediante *index-ID*

    ```
    % chmod Aindex-ID- filename
    ```

  - Eliminación de una entrada de ACL por usuario

    ```
    % chmod A-acl-specification filename
    ```

  - Eliminación de todas las entradas de control de acceso no triviales de un archivo

    ```
    % chmod A- filename
    ```

Para ver en pantalla información de ACL en modo detallado, se utiliza el comando `ls -v`. Por ejemplo:

```
# ls -v file.1
-rw-r--r--   1 root     root      206695 Jul 20 13:43 file.1
     0:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     1:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     2:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
         :allow
```

Para obtener información sobre el uso del formato de ACL compacto, consulte "Establecimiento y visualización de ACL en archivos ZFS en formato compacto" en la página 262.

EJEMPLO 7–1 Modificación de ACL triviales en archivos ZFS

En esta sección, se proporcionan ejemplos acerca de la configuración y la visualización de ACL triviales, lo que significa que únicamente las entradas UNIX tradicionales user, group y other se incluyen en la ACL.

En el ejemplo siguiente, en `file.1` hay una ACL trivial:

```
# ls -v file.1
-rw-r--r--   1 root     root      206695 Jul 20 13:43 file.1
     0:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
```

EJEMPLO 7–1 Modificación de ACL triviales en archivos ZFS *(Continuación)*

```
        /synchronize:allow
     1:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     2:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
        :allow
```

En el ejemplo siguiente, se conceden permisos de `write_data` para `group@`.

```
# chmod A1=group@:read_data/write_data:allow file.1
# ls -v file.1
-rw-rw-r--   1 root     root      206695 Jul 20 13:43 file.1
     0:owner@:read_data/write_data/append_data/read_xattr/write_xattr
        /read_attributes/write_attributes/read_acl/write_acl/write_owner
        /synchronize:allow
     1:group@:read_data/write_data:allow
     2:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
        :allow
```

En el ejemplo siguiente, los permisos de `file.1` se establecen en 644.

```
# chmod 644 file.1
# ls -v file.1
-rw-r--r--   1 root     root      206695 Jul 20 13:43 file.1
     0:owner@:read_data/write_data/append_data/read_xattr/write_xattr
        /read_attributes/write_attributes/read_acl/write_acl/write_owner
        /synchronize:allow
     1:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     2:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
        :allow
```

EJEMPLO 7–2 Establecimiento de ACL no triviales en archivos ZFS

En esta sección se proporcionan ejemplos de establecimiento y visualización de ACL no triviales.

En el ejemplo siguiente, se agregan permisos de `read_data/execute` para el usuario `gozer` en el directorio `test.dir`.

```
# chmod A+user:gozer:read_data/execute:allow test.dir
# ls -dv test.dir
drwxr-xr-x+  2 root     root           2 Jul 20 14:23 test.dir
     0:user:gozer:list_directory/read_data/execute:allow
     1:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
        /append_data/read_xattr/write_xattr/execute/delete_child
        /read_attributes/write_attributes/read_acl/write_acl/write_owner
        /synchronize:allow
     2:group@:list_directory/read_data/read_xattr/execute/read_attributes
        /read_acl/synchronize:allow
     3:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
        /read_acl/synchronize:allow
```

En el ejemplo siguiente, se retiran los permisos de `read_data/execute` para el usuario `gozer`.

EJEMPLO 7–2 Establecimiento de ACL no triviales en archivos ZFS *(Continuación)*

```
# chmod A0- test.dir
# ls -dv test.dir
drwxr-xr-x   2 root     root           2 Jul 20 14:23 test.dir
     0:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     1:group@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
     2:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

EJEMPLO 7–3 Interacción de ACL con permisos en archivos ZFS

Los siguientes ejemplos de ACL ilustran la interacción entre el establecimiento de las ACL y el cambio de los bits de permisos del archivo o el directorio.

En el ejemplo siguiente, en `file.2` hay una ACL trivial:

```
# ls -v file.2
-rw-r--r--   1 root     root        2693 Jul 20 14:26 file.2
     0:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     1:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     2:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
         :allow
```

En el siguiente ejemplo, los permisos `allow` de ACL se eliminan de `everyone@`.

```
# chmod A2- file.2
# ls -v file.2
-rw-r-----   1 root     root        2693 Jul 20 14:26 file.2
     0:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     1:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
```

En esta salida, los bits de permisos del archivo se restablecen de 644 a 640. Los permisos de lectura de `everyone@` se han suprimido de los bits de permisos del archivo cuando se retiran los permisos de LCA de `everyone@`.

En el ejemplo siguiente, la ACL se reemplaza con permisos de `read_data/write_data` para `everyone@`.

```
# chmod A=everyone@:read_data/write_data:allow file.3
# ls -v file.3
-rw-rw-rw-   1 root     root        2440 Jul 20 14:28 file.3
     0:everyone@:read_data/write_data:allow
```

EJEMPLO 7–3 Interacción de ACL con permisos en archivos ZFS *(Continuación)*

En esta salida, la sintaxis de `chmod` reemplaza la ACL con permisos de `read_data/write_data:allow` por permisos de lectura/escritura para owner (propietario), group (grupo) y `everyone@` (cualquiera). En este modelo, `everyone@` especifica acceso a cualquier grupo o usuario. Como no hay entrada de ACL de `owner@` o `group@` para anular los permisos de propietario y grupo, los bits de permisos se establecen en 666.

En el ejemplo siguiente, la ACL se reemplaza por permisos de lectura para el usuario `gozer`.

```
# chmod A=user:gozer:read_data:allow file.3
# ls -v file.3
----------+  1 root     root        2440 Jul 20 14:28 file.3
     0:user:gozer:read_data:allow
```

En esta salida, los permisos de archivo se calculan que sean 000 porque no hay entradas de ACL para `owner@`, `group@` ni `everyone@`, que representan los componentes de permisos habituales de un archivo. El propietario del archivo puede solventar esta situación restableciendo los permisos (y la ACL) de la forma siguiente:

```
# chmod 655 file.3
# ls -v file.3
-rw-r-xr-x   1 root     root        2440 Jul 20 14:28 file.3
     0:owner@:execute:deny
     1:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:read_data/read_xattr/execute/read_attributes/read_acl
         /synchronize:allow
     3:everyone@:read_data/read_xattr/execute/read_attributes/read_acl
         /synchronize:allow
```

EJEMPLO 7–4 Restauración de ACL triviales en archivos ZFS

Puede utilizar el comando `chmod` para eliminar todas las ACL no triviales de un archivo o directorio.

En el ejemplo siguiente, hay dos entradas de control de acceso no triviales en `test5.dir`.

```
# ls -dv test5.dir
drwxr-xr-x+  2 root     root           2 Jul 20 14:32 test5.dir
     0:user:lp:read_data:file_inherit:deny
     1:user:gozer:read_data:file_inherit:deny
     2:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     3:group@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
     4:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

EJEMPLO 7–4 Restauración de ACL triviales en archivos ZFS *(Continuación)*

En el ejemplo siguiente, se han eliminado las LCA no triviales de los usuarios `gozer` and `lp`. La ACL restante contiene los valores predeterminados de `owner@`, `group@` y `everyone@`.

```
# chmod A- test5.dir
# ls -dv test5.dir
drwxr-xr-x   2 root     root           2 Jul 20 14:32 test5.dir
     0:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     1:group@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
     2:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

EJEMPLO 7–5 Aplicación de un conjunto de ACL en los archivos de ZFS

Los conjuntos de ACL están disponibles, de modo que no es necesario aplicar permisos de ACL por separado. Para obtener una descripción de los conjuntos de ACL, consulte "Conjuntos de LCA de ZFS" en la página 246.

Por ejemplo, puede aplicar `read_set` como se indica a continuación:

```
# chmod A+user:otto:read_set:allow file.1
# ls -v file.1
-r--r--r--+  1 root     root      206695 Jul 20 13:43 file.1
     0:user:otto:read_data/read_xattr/read_attributes/read_acl:allow
     1:owner@:read_data/read_xattr/write_xattr/read_attributes
         /write_attributes/read_acl/write_acl/write_owner/synchronize:allow
     2:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     3:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
         :allow
```

Puede aplicar `write_set` y `read_set` como se indica a continuación:

```
# chmod A+user:otto:read_set/write_set:allow file.2
# ls -v file.2
-rw-r--r--+  1 root     root        2693 Jul 20 14:26 file.2
     0:user:otto:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl:allow
     1:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     3:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
         :allow
```

Establecimiento de herencia de LCA en archivos ZFS en formato detallado

Puede determinar la forma en que se heredan o no las ACL en archivos o directorios. De forma predeterminada, las LCA no se propagan. Si en un directorio se establece una ACL no trivial, no se heredará en ningún directorio posterior. Debe especificar la herencia de una LCA en un archivo o directorio.

La propiedad `aclinherit` se puede establecer de manera global en un sistema de archivos. De manera predeterminada, la propiedad `aclinherit` está establecida en `restricted`.

Para obtener más información, consulte "Herencia de ACL" en la página 247.

EJEMPLO 7–6 Concesión de herencia de ACL predeterminada

De forma predeterminada, las ACL no se propagan por una estructura de directorios.

En el ejemplo siguiente, se aplica una entrada de control de acceso no trivial de `read_data/write_data/execute` para el usuario `gozer` en `test.dir`.

```
# chmod A+user:gozer:read_data/write_data/execute:allow test.dir
# ls -dv test.dir
drwxr-xr-x+  2 root     root           2 Jul 20 14:53 test.dir
     0:user:gozer:list_directory/read_data/add_file/write_data/execute:allow
     1:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
     3:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

Si se crea un subdirectorio de `test.dir`, no se propaga la entrada de control de acceso del usuario `gozer`. El usuario `gozer` sólo dispondrá de acceso a `sub.dir` si se le conceden permisos de acceso de `sub.dir` como propietario del archivo, miembro del grupo o `everyone@`.

```
# mkdir test.dir/sub.dir
# ls -dv test.dir/sub.dir
drwxr-xr-x   2 root     root           2 Jul 20 14:54 test.dir/sub.dir
     0:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     1:group@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
     2:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

EJEMPLO 7–7 Concesión de herencia de ACL en archivos y directorios

En los ejemplos siguientes se identifican las entradas de control de acceso de archivos y directorios que se aplican al establecerse el indicador `file_inherit`.

En el ejemplo siguiente, se agregan los permisos `read_data/write_data` para los archivos del directorio `test2.dir` para el usuario `gozer`, de manera que éste disponga de acceso de lectura a cualquier archivo que se cree.

```
# chmod A+user:gozer:read_data/write_data:file_inherit:allow test2.dir
# ls -dv test2.dir
drwxr-xr-x+  2 root     root           2 Jul 20 14:55 test2.dir
     0:user:gozer:read_data/write_data:file_inherit:allow
     1:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
     3:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

En el ejemplo siguiente, los permisos del usuario de `gozer` se aplican en el archivo `test2.dir/file.2` recién creado. La herencia de LCA concedida, `read_data:file_inherit:allow`, significa que el usuario `gozer` puede leer el contenido de cualquier archivo que se cree.

```
# touch test2.dir/file.2
# ls -v test2.dir/file.2
-rw-r--r--+  1 root     root           0 Jul 20 14:56 test2.dir/file.2
     0:user:gozer:read_data:inherited:allow
     1:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     3:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
         :allow
```

Como la propiedad `aclinherit` para este sistema de archivos se establece en el modo predeterminado, `restricted`, el usuario `gozer` no dispone del permiso `write_data` en `file.2` porque el permiso de grupo del archivo no lo permite.

El permiso `inherit_only`, que se concede si se establecen los indicadores `file_inherit` o `dir_inherit`, se emplea para propagar la ACL por la estructura de directorios. Así, al usuario `gozer` sólo se le conceden o deniegan permisos de `everyone@`, a menos que sea propietario del archivo o miembro del grupo propietario del archivo. Por ejemplo:

```
# mkdir test2.dir/subdir.2
# ls -dv test2.dir/subdir.2
drwxr-xr-x+  2 root     root           2 Jul 20 14:57 test2.dir/subdir.2
     0:user:gozer:list_directory/read_data/add_file/write_data:file_inherit
         /inherit_only/inherited:allow
```

EJEMPLO 7–7 Concesión de herencia de ACL en archivos y directorios *(Continuación)*

```
     1:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
     3:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

En los ejemplos siguientes se identifican las ACL de archivo y directorio que se aplican si se establecen los indicadores `file_inherit` y `dir_inherit`.

En el ejemplo siguiente, al usuario `gozer` se le conceden permisos de lectura, escritura y ejecución que se heredan para archivos y directorios recientemente creados.

```
# chmod A+user:gozer:read_data/write_data/execute:file_inherit/dir_inherit:allow
test3.dir
# ls -dv test3.dir
drwxr-xr-x+  2 root     root           2 Jul 20 15:00 test3.dir
     0:user:gozer:list_directory/read_data/add_file/write_data/execute
         :file_inherit/dir_inherit:allow
     1:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
     3:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

El texto `inherited` en el resultado que se muestra a continuación es un mensaje informativo que indica que la ACE es heredada.

```
# touch test3.dir/file.3
# ls -v test3.dir/file.3
-rw-r--r--+  1 root     root           0 Jul 20 15:01 test3.dir/file.3
     0:user:gozer:read_data:inherited:allow
     1:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     3:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
         :allow
```

En los ejemplos anteriores, debido a que los bits de permisos del directorio principal para `group@` y `everyone@` deniegan permisos de escritura y ejecución, se le deniegan al usuario `gozer` permisos de escritura y ejecución. El valor predeterminado de la propiedad `aclinherit` es `restricted`, lo cual significa que no se heredan los permisos `write_data` y `execute`.

EJEMPLO 7–7 Concesión de herencia de ACL en archivos y directorios *(Continuación)*

En el siguiente ejemplo, al usuario `gozer` se le conceden derechos de lectura, escritura y ejecución que se heredan para archivos recientemente creados pero que no se propagan por el resto del directorio.

```
# chmod A+user:gozer:read_data/write_data/execute:file_inherit/no_propagate:allow
test4.dir
# ls -dv test4.dir
drwxr--r--+  2 root     root           2 Mar  1 12:11 test4.dir
     0:user:gozer:list_directory/read_data/add_file/write_data/execute
         :file_inherit/no_propagate:allow
     1:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:list_directory/read_data/read_xattr/read_attributes/read_acl
         /synchronize:allow
     3:everyone@:list_directory/read_data/read_xattr/read_attributes/read_acl
         /synchronize:allow
```

Como se ilustra en el siguiente ejemplo, los permisos `read_data/write_data/execute` de `gozer` se reducen en función de los permisos del grupo propietario.

```
# touch test4.dir/file.4
# ls -v test4.dir/file.4
-rw-r--r--+  1 root     root           0 Jul 20 15:09 test4.dir/file.4
     0:user:gozer:read_data:inherited:allow
     1:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     3:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
         :allow
```

EJEMPLO 7–8 Herencia de ACL con el modo ACL heredado establecido en passthrough

Si la propiedad `aclinherit` del sistema de archivos `tank/cindy` se establece en `passthrough`, el usuario `gozer` hereda la ACL que se aplica a `test5.dir` para el archivo recién creado `file.4` de la manera que se indica a continuación:

```
# zfs set aclinherit=passthrough tank/cindy
# touch test4.dir/file.5
# ls -v test4.dir/file.5
-rw-r--r--+  1 root     root           0 Jul 20 14:16 test4.dir/file.5
     0:user:gozer:read_data/write_data/execute:inherited:allow
     1:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     3:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
         :allow
```

EJEMPLO 7–9 Herencia de ACL con el modo de herencia de ACL establecido en Discard

Si la propiedad `aclinherit` de un sistema de archivos se establece en `discard`, las ACL se pueden descartar si cambian los bits de permisos en un directorio. Por ejemplo:

```
# zfs set aclinherit=discard tank/cindy
# chmod A+user:gozer:read_data/write_data/execute:dir_inherit:allow test5.dir
# ls -dv test5.dir
drwxr-xr-x+  2 root     root           2 Jul 20 14:18 test5.dir
     0:user:gozer:list_directory/read_data/add_file/write_data/execute
         :dir_inherit:allow
     1:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
     3:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

Si, posteriormente, decide restringir los bits de permisos de un directorio, se prescinde de la ACL no trivial. Por ejemplo:

```
# chmod 744 test5.dir
# ls -dv test5.dir
drwxr--r--   2 root     root           2 Jul 20 14:18 test5.dir
     0:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     1:group@:list_directory/read_data/read_xattr/read_attributes/read_acl
         /synchronize:allow
     2:everyone@:list_directory/read_data/read_xattr/read_attributes/read_acl
         /synchronize:allow
```

EJEMPLO 7–10 Herencia de ACL con el modo de herencia de ACL establecido en nonallow

En este ejemplo se establecen dos ACL no triviales con herencia de archivos. Una ACL concede el permiso `read_data` y una ACL deniega el permiso `read_data`. Asimismo, el ejemplo muestra la manera de especificar dos entradas de control de acceso en el mismo comando `chmod`.

```
# zfs set aclinherit=noallow tank/cindy
# chmod A+user:gozer:read_data:file_inherit:deny,user:lp:read_data:file_inherit:allow
test6.dir
# ls -dv test6.dir
drwxr-xr-x+  2 root     root           2 Jul 20 14:22 test6.dir
     0:user:gozer:read_data:file_inherit:deny
     1:user:lp:read_data:file_inherit:allow
     2:owner@:list_directory/read_data/add_file/write_data/add_subdirectory
         /append_data/read_xattr/write_xattr/execute/delete_child
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     3:group@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

EJEMPLO 7–10 Herencia de ACL con el modo de herencia de ACL establecido en nonallow *(Continuación)*

```
     4:everyone@:list_directory/read_data/read_xattr/execute/read_attributes
         /read_acl/synchronize:allow
```

Como se indica en el ejemplo siguiente, al crear un archivo, se prescinde de la ACL que concede el permiso `read_data`.

```
# touch test6.dir/file.6
# ls -v test6.dir/file.6
-rw-r--r--+  1 root     root           0 Jul 20 14:23 test6.dir/file.6
     0:user:gozer:read_data:inherited:deny
     1:owner@:read_data/write_data/append_data/read_xattr/write_xattr
         /read_attributes/write_attributes/read_acl/write_acl/write_owner
         /synchronize:allow
     2:group@:read_data/read_xattr/read_attributes/read_acl/synchronize:allow
     3:everyone@:read_data/read_xattr/read_attributes/read_acl/synchronize
         :allow
```

Establecimiento y visualización de ACL en archivos ZFS en formato compacto

En archivos ZFS puede establecer y visualizar permisos en un formato compacto que utiliza 14 caracteres exclusivos para representar los permisos. Las letras que representan los permisos compactos se enumeran en la Tabla 7–2 and Tabla 7–4.

Puede visualizar listas de ACL en formato compacto para archivos y directorios mediante el comando `ls -V`. Por ejemplo:

```
# ls -V file.1
-rw-r--r--   1 root     root      206695 Jul 20 14:27 file.1
                owner@:rw-p--aARWcCos:-------:allow
                group@:r-----a-R-c--s:-------:allow
             everyone@:r-----a-R-c--s:-------:allow
```

La salida de ACL compacta se interpreta de la forma siguiente:

`owner@` El propietario puede leer y modificar el contenido del archivo (`rw=read_data/write_data`), (`p= append_data`). El propietario también puede modificar los atributos del archivo, como las indicaciones de fecha y hora, los atributos ampliados y las ACL (`a=read_attributes` , `W=write_xattr`, `R= read_xattr`, `A=write_attributes` , `c=read_acl`, `C= write_acl`). Además, el propietario puede modificar la propiedad del archivo (`o=write_owner`).

El permiso de acceso `synchronize` (`s`) no está implementado en la actualidad.

`group@` Se otorga al grupo permisos de lectura para el archivo (`r= read_data`) y los atributos del archivo (`a=read_attributes` , `R=read_xattr`, `c= read_acl`).

El permiso de acceso `synchronize` (`s`) no está implementado en la actualidad.

`everyone@` Se concede a quien no sea usuario ni grupo los permisos de lectura del archivo y los atributos del archivo (`r=read_data`, `a=append_data`, `R=read_xattr` , `c=read_acl` y `s= synchronize`).

El permiso de acceso `synchronize` (`s`) no está implementado en la actualidad.

El formato compacto de las ACL presenta las ventajas siguientes respecto al formato detallado:

- Los permisos se pueden especificar como argumentos posicionales en el comando `chmod`.
- Los caracteres de guión (-), que no identifican permisos, se pueden eliminar. Sólo hace falta especificar los caracteres necesarios.
- Los indicadores de permisos y de herencia se establecen de la misma manera.

Para obtener información sobre el uso del formato de ACL detallado, consulte "Establecimiento y visualización de ACL en archivos ZFS en formato detallado" en la página 251.

EJEMPLO 7–11 Establecimiento y visualización de las ACL en formato compacto

En el ejemplo siguiente, en `file.1` hay una LCA trivial:

```
# ls -V file.1
-rw-r--r--   1 root     root      206695 Jul 20 14:27 file.1
                owner@:rw-p--aARWcCos:-------:allow
                group@:r-----a-R-c--s:-------:allow
             everyone@:r-----a-R-c--s:-------:allow
```

En este ejemplo, se agregan los permisos `read_data/execute` para el usuario `gozer` en `file.1`.

```
# chmod A+user:gozer:rx:allow file.1
# ls -V file.1
-rw-r--r--+  1 root     root      206695 Jul 20 14:27 file.1
             user:gozer:r-x-----------:-------:allow
                owner@:rw-p--aARWcCos:-------:allow
                group@:r-----a-R-c--s:-------:allow
             everyone@:r-----a-R-c--s:-------:allow
```

En el ejemplo siguiente, al usuario `gozer` se le conceden permisos de lectura, escritura y ejecución que se heredan para archivos y directorios recientemente creados mediante el formato de ACL comprimido.

```
# chmod A+user:gozer:rwx:fd:allow dir.2
# ls -dV dir.2
drwxr-xr-x+  2 root     root           2 Jul 20 14:33 dir.2
             user:gozer:rwx-----------:fd-----:allow
                owner@:rwxp-DaARWcCos:-------:allow
```

EJEMPLO 7–11 Establecimiento y visualización de las ACL en formato compacto *(Continuación)*

```
                group@:r-x---a-R-c--s:-------:allow
             everyone@:r-x---a-R-c--s:-------:allow
```

También puede cortar y pegar indicadores de herencia y permisos de la salida de `ls -V` en el formato compacto de `chmod`. Por ejemplo, para duplicar los permisos e indicadores de herencia de `dir.2` para el usuario `gozer` en el usuario `cindy` en `dir.2`, copie y pegue los permisos y los indicadores de herencia (`rwx-----------:fd-----:allow`) en el comando `chmod`. Por ejemplo:

```
# chmod A+user:cindy:rwx-----------:fd-----:allow dir.2
# ls -dV dir.2
drwxr-xr-x+  2 root     root           2 Jul 20 14:33 dir.2
             user:cindy:rwx-----------:fd-----:allow
             user:gozer:rwx-----------:fd-----:allow
                 owner@:rwxp-DaARWcCos:-------:allow
                 group@:r-x---a-R-c--s:-------:allow
              everyone@:r-x---a-R-c--s:-------:allow
```

EJEMPLO 7–12 Herencia de ACL con el modo ACL heredado establecido en passthrough

Un sistema de archivos que tiene la propiedad `aclinherit` establecida en `passthrough` hereda todas las entradas LCA que se pueden heredar sin modificaciones en las entradas LCA cuando se heredan. Si la propiedad se configura como `passthrough`, los archivos se crean con un modo de permiso que determinan las entradas de control de acceso que se pueden heredar. Si no existen entradas de control de acceso que se puedan heredar y que afecten al modo de permiso, el modo de permiso se configurará de acuerdo con el modo solicitado desde la aplicación.

Los ejemplos siguientes utilizan sintaxis de ACL compacta para mostrar cómo heredar bits de permisos estableciendo el modo `aclinherit` en `passthrough`.

En este ejemplo, se establece una ACL en `test1.dir` para forzar la herencia. La sintaxis crea una entrada LCA `owner@`, `group@` y `everyone@` para cada archivo que cree. Los directorios que cree heredan una entrada de ACL `@owner`, `group@` y `everyone@`.

```
# zfs set aclinherit=passthrough tank/cindy
# pwd
/tank/cindy
# mkdir test1.dir

# chmod A=owner@:rwxpcCosRrWaAdD:fd:allow,group@:rwxp:fd:allow,
everyone@::fd:allow test1.dir
# ls -Vd test1.dir
drwxrwx---+  2 root     root           2 Jul 20 14:42 test1.dir
                 owner@:rwxpdDaARWcCos:fd-----:allow
                 group@:rwxp----------:fd-----:allow
              everyone@:--------------:fd-----:allow
```

En este ejemplo, un archivo recién creado hereda la LCA especificada para heredarse en los archivos recién creados.

EJEMPLO 7–12 Herencia de ACL con el modo ACL heredado establecido en passthrough *(Continuación)*

```
# cd test1.dir
# touch file.1
# ls -V file.1
-rwxrwx---+  1 root     root           0 Jul 20 14:44 file.1
                  owner@:rwxpdDaARWcCos:------I:allow
                  group@:rwxp----------:------I:allow
               everyone@:--------------:------I:allow
```

En este ejemplo, un directorio que se cree hereda tanto las entradas que controlan el acceso a este directorio como las entradas de control de acceso para la futura propagación a los elementos secundarios del directorio que se cree.

```
# mkdir subdir.1
# ls -dV subdir.1
drwxrwx---+  2 root     root           2 Jul 20 14:45 subdir.1
                  owner@:rwxpdDaARWcCos:fd----I:allow
                  group@:rwxp----------:fd----I:allow
               everyone@:--------------:fd----I:allow
```

Las entradas `fd----I` son para propagar la herencia y no se tienen en cuenta durante el control de acceso.

En el siguiente ejemplo, se crea un archivo con una ACL trivial en otro directorio en el que no hay entradas de control de acceso heredadas.

```
# cd /tank/cindy
# mkdir test2.dir
# cd test2.dir
# touch file.2
# ls -V file.2
-rw-r--r--   1 root     root           0 Jul 20 14:48 file.2
                  owner@:rw-p--aARWcCos:-------:allow
                  group@:r-----a-R-c--s:-------:allow
               everyone@:r-----a-R-c--s:-------:allow
```

EJEMPLO 7–13 Herencia de ACL con el modo ACL heredado establecido en passthrough-X

Cuando `aclinherit=passthrough-x` está activada, los archivos se crean con el permiso de ejecución (`x`) para `owner@`, `group@` o `everyone@`, pero sólo si el permiso de ejecución se define en el modo de creación de archivos y en un ACE heredable que afecta al modo.

El siguiente ejemplo muestra cómo se heredan los permisos de ejecución al establecer el modo `aclinherit` en `passthrough-x`.

```
# zfs set aclinherit=passthrough-x tank/cindy
```

La siguiente ACL se establece en `/tank/cindy/test1.dir` para proporcionar herencia de ACL ejecutable para los archivos de `owner@`.

EJEMPLO 7–13 Herencia de ACL con el modo ACL heredado establecido en passthrough-X *(Continuación)*

```
# chmod A=owner@:rwxpcCosRrWaAdD:fd:allow,group@:rwxp:fd:allow,
everyone@::fd:allow test1.dir
# ls -Vd test1.dir
drwxrwx---+  2 root     root           2 Jul 20 14:50 test1.dir
                owner@:rwxpdDaARWcCos:fd-----:allow
                group@:rwxp----------:fd-----:allow
             everyone@:--------------:fd-----:allow
```

Un archivo (`file1`) se crea con permisos solicitados `0666`. Los permisos resultantes son `0660`. El permiso de ejecución no se ha heredado porque el modo de creación no lo solicitó.

```
# touch test1.dir/file1
# ls -V test1.dir/file1
-rw-rw----+  1 root     root           0 Jul 20 14:52 test1.dir/file1
                owner@:rw-pdDaARWcCos:------I:allow
                group@:rw-p----------:------I:allow
             everyone@:--------------:------I:allow
```

A continuación, se generará un ejecutable llamado `t` mediante el compilador `cc` en el directorio `testdir`.

```
# cc -o t t.c
# ls -V t
-rwxrwx---+  1 root     root        7396 Dec  3 15:19 t
                owner@:rwxpdDaARWcCos:------I:allow
                group@:rwxp----------:------I:allow
             everyone@:--------------:------I:allow
```

Los permisos resultantes son `0770` porque `cc` solicitó permisos `0777`, que provocaron que el permiso de ejecución se heredara de las entradas `owner@`, `group@` y `everyone@`.

EJEMPLO 7–14 Interacción de ACL con las operaciones `chmod` en archivos ZFS

Los siguientes ejemplos muestran de qué manera los valores específicos de propiedad `aclmode` y `aclinherit` influyen en la interacción de las ACL existentes con una operación `chmod` que cambia los permisos de archivo o de directorio para reducir o ampliar los permisos de ACL existentes a fin de que sean coherentes con el grupo propietario.

En este ejemplo, la propiedad `aclmode` se establece como `mask` y la propiedad `aclinherit` se establece como `restricted`. Los permisos de ACL de este ejemplo se muestran en modo compacto, que permite ilustrar el cambio de los permisos con más facilidad.

El archivo original y la propiedad de grupo y los permisos de ACL son los siguientes:

```
# zfs set aclmode=mask pond/whoville
# zfs set aclinherit=restricted pond/whoville

# ls -lV file.1
```

EJEMPLO 7–14 Interacción de ACL con las operaciones chmod en archivos ZFS *(Continuación)*

```
-rwxrwx---+  1 root     root      206695 Aug 30 16:03 file.1
               user:amy:r-----a-R-c---:-------:allow
              user:rory:r-----a-R-c---:-------:allow
         group:sysadmin:rw-p--aARWc---:-------:allow
            group:staff:rw-p--aARWc---:-------:allow
                 owner@:rwxp--aARWcCos:-------:allow
                 group@:rwxp--aARWc--s:-------:allow
              everyone@:------a-R-c--s:-------:allow
```

Una operación chown cambia la propiedad de archivo de `file.1`, y el usuario propietario, `amy`, empieza a ver la salida. Por ejemplo:

```
# chown amy:staff file.1
# su - amy
$ ls -lV file.1
-rwxrwx---+  1 amy      staff     206695 Aug 30 16:03 file.1
               user:amy:r-----a-R-c---:-------:allow
              user:rory:r-----a-R-c---:-------:allow
         group:sysadmin:rw-p--aARWc---:-------:allow
            group:staff:rw-p--aARWc---:-------:allow
                 owner@:rwxp--aARWcCos:-------:allow
                 group@:rwxp--aARWc--s:-------:allow
              everyone@:------a-R-c--s:-------:allow
```

La siguiente operación chmod cambia el modo de los permisos a uno más restrictivo. En este ejemplo, los permisos de ACL modificados de los grupos `sysadmin` y `staff` no exceden los permisos del grupo propietario.

```
$ chmod 640 file.1
$ ls -lV file.1
-rw-r-----+  1 amy      staff     206695 Aug 30 16:03 file.1
               user:amy:r-----a-R-c---:-------:allow
              user:rory:r-----a-R-c---:-------:allow
         group:sysadmin:r-----a-R-c---:-------:allow
            group:staff:r-----a-R-c---:-------:allow
                 owner@:rw-p--aARWcCos:-------:allow
                 group@:r-----a-R-c--s:-------:allow
              everyone@:------a-R-c--s:-------:allow
```

La siguiente operación chmod cambia el modo de los permisos a uno menos restrictivo. En este ejemplo, los permisos de ACL modificados de los grupos `sysadmin` y `staff` se restauran para permitir los mismos permisos que el grupo propietario.

```
$ chmod 770 file.1
$ ls -lV file.1
-rwxrwx---+  1 amy      staff     206695 Aug 30 16:03 file.1
               user:amy:r-----a-R-c---:-------:allow
              user:rory:r-----a-R-c---:-------:allow
         group:sysadmin:rw-p--aARWc---:-------:allow
            group:staff:rw-p--aARWc---:-------:allow
                 owner@:rwxp--aARWcCos:-------:allow
                 group@:rwxp--aARWc--s:-------:allow
```

EJEMPLO 7–14 Interacción de ACL con las operaciones chmod en archivos ZFS *(Continuación)*

```
everyone@:------a-R-c--s:-------:allow
```

Aplicación de atributos especiales a los archivos de ZFS

Los siguientes ejemplos muestran cómo aplicar y visualizar atributos especiales, como inmutabilidad o acceso de sólo lectura, a los archivos de ZFS.

Para obtener más información sobre la visualización y la aplicación de atributos especiales, consulte `ls(1)` y `chmod(1)`.

EJEMPLO 7–15 Aplicar inmutabilidad a un archivo ZFS

Utilice la siguiente sintaxis para convertir a un archivo en inmutable:

```
# chmod S+ci file.1
# echo this >>file.1
-bash: file.1: Not owner
# rm file.1
rm: cannot remove ‘file.1’: Not owner
```

Puede mostrar atributos especiales en archivos ZFS mediante la siguiente sintaxis:

```
# ls -l/c file.1
-rw-r--r--+  1 root     root      206695 Jul 20 14:27 file.1
                {A-----im----}
```

Utilice la siguiente sintaxis para eliminar la inmutabilidad del archivo:

```
# chmod S-ci file.1
# ls -l/c file.1
-rw-r--r--+  1 root     root      206695 Jul 20 14:27 file.1
                {A------m----}
# rm file.1
```

EJEMPLO 7–16 Aplicar acceso de sólo lectura a un archivo ZFS

El siguiente ejemplo muestra cómo aplicar acceso de sólo lectura a un archivo ZFS.

```
# chmod S+cR file.2
# echo this >>file.2
-bash: file.2: Not owner
```

EJEMPLO 7–17 Mostrar y cambiar atributos de archivos ZFS

Puede mostrar y configurar atributos especiales con la siguiente sintaxis:

EJEMPLO 7–17 Mostrar y cambiar atributos de archivos ZFS *(Continuación)*

```
# ls -l/v file.3
-r--r--r--   1 root     root      206695 Jul 20 14:59 file.3
                {archive,nohidden,noreadonly,nosystem,noappendonly,nonodump,
noimmutable,av modified,noav_quarantined,nonounlink,nooffline,nosparse}
# chmod S+cR file.3
# ls -l/v file.3
-r--r--r--   1 root     root      206695 Jul 20 14:59 file.3
                {archive,nohidden,readonly,nosystem,noappendonly,nonodump,noimmutable,
av_modified,noav_quarantined,nonounlink,nooffline,nosparse}
```

Algunos de estos atributos solamente se aplican en un entorno Oracle Solaris SMB.

Puede borrar todos los atributos de un archivo. Por ejemplo:

```
# chmod S-a file.3
# ls -l/v file.3
-r--r--r--   1 root     root      206695 Jul 20 14:59 file.3
                {noarchive,nohidden,noreadonly,nosystem,noappendonly,nonodump,
noimmutable,noav_modified,noav_quarantined,nonounlink,nooffline,nosparse}
```

◆ ◆ ◆ CAPÍTULO 8

Administración delegada de ZFS Oracle Solaris

Este capítulo describe la forma de utilizar la administración delegada para permitir que los usuarios sin privilegios puedan efectuar tareas de administración de ZFS.

Este capítulo se divide en las secciones siguientes:

Descripción general de la administración delegada de ZFS

La administración delegada de ZFS permite distribuir permisos más concretos a determinados usuarios, grupos o a todo el mundo. Se permiten dos tipos de permisos delegados:

- Se pueden delegar permisos concretos a determinadas personas, por ejemplo `create`, `destroy`, `mount`, `snapshot`, etcétera.
- Se pueden definir grupos de permisos denominados *conjuntos de permisos*. Un conjunto de permisos se puede actualizar posteriormente y todos los usuarios de dicho conjunto adquieren ese cambio de forma automática. Los conjuntos de permisos comienzan con el carácter `@` y tienen un límite de 64 caracteres. Después del símbolo `@`, los demás caracteres del nombre del conjunto están sujetos a las mismas restricciones que los nombres de sistemas de archivos ZFS normales.

La administración delegada de ZFS proporciona funciones parecidas al modelo de seguridad RBAC. La delegación de ZFS aporta las ventajas siguientes en la administración de agrupaciones de almacenamiento y sistemas de archivos ZFS:

- Si se migra la agrupación, se mantienen los permisos en la agrupación de almacenamiento ZFS.

- Proporciona herencia dinámica para poder controlar cómo se propagan los permisos por los sistemas de archivos.
- Se puede configurar para que sólo el creador de un sistema de archivos pueda destruir el sistema de archivos.
- Se pueden delegar permisos en determinados sistemas de archivos. Los sistemas de archivos creados pueden designar permisos automáticamente.
- Proporciona administración NFS simple. Por ejemplo, un usuario que cuenta con permisos explícitos puede crear una instantánea por NFS en el correspondiente directorio `.zfs/snapshot`.

A la hora de distribuir tareas de ZFS, plantéese la posibilidad de utilizar la administración delegada. Para obtener información sobre el uso de RBAC para gestionar tareas de administración de Oracle Solaris, consulte la Parte III, "Roles, perfiles de derechos y privilegios" de *Administración de Oracle Solaris 11.1: servicios de seguridad*.

Desactivación de permisos delegados de ZFS

Puede controlar las funciones de administración delegada mediante la propiedad `delegation` de la agrupación. Por ejemplo:

```
# zpool get delegation users
NAME  PROPERTY    VALUE       SOURCE
users  delegation  on          default
# zpool set delegation=off users
# zpool get delegation users
NAME  PROPERTY    VALUE       SOURCE
users  delegation  off         local
```

De forma predeterminada se activa la propiedad `delegation`.

Delegación de permisos de ZFS

Utilice el comando `zfs allow` para delegar permisos en sistemas de archivos ZFS a usuarios que no sean root de las maneras siguientes:

- Los permisos individuales se pueden delegar a un grupo, un usuario o a todo el mundo.
- Los grupos de permisos individuales se pueden delegar como *conjunto de permisos* a un grupo, un usuario o a todo el mundo.
- Los permisos se pueden delegar localmente sólo al sistema de archivos actual o a todos los descendientes de dicho sistema de archivos.

En la tabla siguiente figuran las operaciones que se pueden delegar y los permisos dependientes que se necesitan para efectuar las operaciones delegadas.

Permiso (subcomando)	Descripción	Dependencias
`allow`	Permiso para delegar a otro usuario los permisos que tiene uno mismo.	También debe tener el permiso que se está concediendo.
`clónico`	Permiso para clonar cualquier instantánea del conjunto de datos.	También se debe disponer de los permisos `create` y `mount` en el sistema de archivos original.
`create`	Permiso para crear conjuntos de archivos descendientes.	También se debe disponer del permiso `mount`.
`destroy`	Permiso para destruir un conjunto de datos.	También se debe disponer del permiso `mount`.
`diff`	Permiso para identificar rutas en un conjunto de datos.	Los usuarios que no son root necesitan este permiso para utilizar el comando `zfs diff`.
`hold`	Permiso para retener una instantánea.	
`mount`	Permiso para montar y desmontar un sistema de archivos, así como para crear y destruir vínculos de dispositivos de volumen.	
`promote`	Permiso para promover un clon a un conjunto de datos.	También se debe disponer de los permisos `mount` y `promote` en el sistema de archivos original.
`receive`	Permiso para crear sistemas de archivos descendientes mediante el comando `zfs receive`.	También se debe disponer de los permisos `mount` y `create`.
`release`	Permiso para liberar la retención de una instantánea, lo que puede destruir la instantánea.	
`rename`	Permiso para cambiar el nombre a un conjunto de datos.	También se debe disponer de los permisos `create` y `mount` en el nuevo elemento principal.
`rollback`	Permiso para revertir una instantánea.	
`send`	Permiso para enviar un flujo de instantáneas.	
`share`	Permiso para compartir y anular la compartición de un sistema de archivos.	Se debe disponer de `share` y `share.nfs` para crear un recurso compartido NFS. Se debe disponer de `share` y `share.smb` para crear un recurso compartido SMB.

Permiso (subcomando)	Descripción	Dependencias
instantánea	Permiso para crear una instantánea de un conjunto de datos.	

Puede delegar el siguiente conjunto de permisos, pero un permiso puede estar limitado a acceso, lectura o cambio:

- groupquota
- groupused
- clave
- keychange
- userprop
- userquota
- userused

Además, puede delegar la administración de las siguientes propiedades de ZFS a usuarios que no sean root:

- aclinherit
- aclmode
- atime
- canmount
- casesensitivity
- suma de comprobación
- compression
- copies
- dedup
- devices
- cifrado
- exec
- keysource
- logbias
- mountpoint
- nbmand
- normalization
- primarycache
- quota
- readonly
- recordsize
- refquota
- refreservation
- reservation
- rstchown
- secondarycache
- setuid

- shadow
- share.nfs
- share.smb
- snapdir
- sync
- utf8only
- version
- volblocksize
- volsize
- vscan
- xattr
- zoned

Algunas de estas propiedades sólo se pueden establecer al crear el conjunto de datos. Para obtener una descripción de estas propiedades, consulte "Introducción a las propiedades de ZFS" en la página 151.

Delegación de permisos de ZFS (zfs allow)

La sintaxis zfs allow es la siguiente:

```
zfs allow -[ldugecs] everyone|user|group[,...] perm|@setname,...] filesystem| volume
```

La siguiente sintaxis de zfs allow (en negrita) identifica a quién se delegan los permisos:

zfs allow [-uge]|user|group|everyone [,...] *filesystem* | *volume*

Se pueden especificar varias entidades en una lista separada por comas. Si no se especifican opciones de -uge, el argumento se interpreta de forma preferente como la palabra clave everyone, después como nombre de usuario y, en último lugar, como grupo de nombre. Para especificar un usuario o grupo denominado "everyone", utilice la opción -u o -g. Para especificar un grupo con el mismo nombre que un usuario, utilice la opción -g. La opción -c delega permisos de create-time.

La siguiente sintaxis de zfs allow (en negrita) identifica cómo se especifican los permisos y conjuntos de permisos:

zfs allow [-s] ... perm|@setname [,...] *filesystem* | *volume*

Se pueden especificar varios permisos en una lista separada por comas. Los nombres de permisos son los mismos que las propiedades y los subcomandos de ZFS. Para obtener más información, consulte la sección anterior.

Los permisos se pueden agregar a *conjuntos de permisos* y los identifica la opción -s. Otros comandos de zfs allow pueden utilizar conjuntos de permisos para el sistema de archivos especificado y sus elementos descendientes. Los conjuntos de permisos se evalúan

dinámicamente, por lo tanto los cambios que haya en un conjunto se actualizan de manera inmediata. Los conjuntos de permisos siguen los mismos requisitos de denominación que los sistemas de archivos ZFS; sin embargo, el nombre debe comenzar con el signo de arroba (@) y tener un máximo de 64 caracteres.

La siguiente sintaxis de `zfs allow` (en negrita) identifica cómo se delegan los permisos:

`zfs allow [-ld] ... ...` *filesystem* | *volume*

La opción `-l` indica que los permisos se conceden al sistema de archivos especificado, pero no a los descendientes, a menos que también se especifique la opción `-d`. La opción `-d` indica que los permisos se conceden a los sistemas de archivos descendientes, pero no a este sistema de archivos, a menos que también se especifique la opción `-l`. Si no se especifica ninguna de las opciones, los permisos se conceden para el volumen o sistema de archivos y todos sus elementos descendientes.

Eliminación de permisos delegados de ZFS (`zfs unallow`)

Mediante el comando `zfs unallow` se pueden eliminar los permisos que se han delegado.

Por ejemplo, suponga que ha delegado los permisos `create`, `destroy`, `mount` y `snapshot` de la forma siguiente:

```
# zfs allow cindy create,destroy,mount,snapshot tank/home/cindy
# zfs allow tank/home/cindy
---- Permissions on tank/home/cindy --------------------------------
Local+Descendent permissions:
        user cindy create,destroy,mount,snapshot
```

Para eliminar estos permisos, deberá utilizar la siguiente sintaxis:

```
# zfs unallow cindy tank/home/cindy
# zfs allow tank/home/cindy
```

Delegación de permisos ZFS (ejemplos)

EJEMPLO 8–1 Delegación de permisos a un determinado usuario

Si delega los permisos `create` y `mount` a un determinado usuario, compruebe que dicho usuario disponga de permisos en el punto de montaje subyacente.

Por ejemplo, para delegar al usuario `mark` los permisos `create` y `mount` en el sistema de archivos `tank`, primero defina los permisos:

```
# chmod A+user:mark:add_subdirectory:fd:allow /tank/home
```

EJEMPLO 8–1 Delegación de permisos a un determinado usuario *(Continuación)*

A continuación, utilice el comando `zfs allow` para delegar los permisos `create` , `destroy` y `mount`. Por ejemplo:

```
# zfs allow mark create,destroy,mount tank/home
```

El usuario `mark` ahora puede crear sus propios sistemas de archivos en el sistema de archivos `tank/home`. Por ejemplo:

```
# su mark
mark$ zfs create tank/home/mark
mark$ ^D
# su lp
$ zfs create tank/home/lp
cannot create 'tank/home/lp': permission denied
```

EJEMPLO 8–2 Delegación de los permisos create y destroy en un grupo.

El siguiente ejemplo muestra cómo configurar un sistema de archivos de forma que cualquier integrante del grupo `staff` pueda crear y montar sistemas de archivos en el sistema de archivos `tank/home`, así como destruir sus propios sistemas de archivos. Ahora bien, los miembros del grupo `staff` no pueden destruir los sistemas de archivos de nadie más.

```
# zfs allow staff create,mount tank/home
# zfs allow -c create,destroy tank/home
# zfs allow tank/home
---- Permissions on tank/home --------------------------------------
Create time permissions:
        create,destroy
Local+Descendent permissions:
        group staff create,mount
# su cindy
cindy% zfs create tank/home/cindy/files
cindy% exit
# su mark
mark% zfs create tank/home/mark/data
mark% exit
cindy% zfs destroy tank/home/mark/data
cannot destroy 'tank/home/mark/data': permission denied
```

EJEMPLO 8–3 Delegación de permisos en el nivel correcto del sistema de archivos

Compruebe que conceda permisos a los usuarios en el nivel correcto del sistema de archivos. Por ejemplo, se delegan al usuario `mark` los permisos `create`, `destroy` y `mount` para los sistemas de archivos local y descendiente. Se delega al usuario `mark` permiso local para crear una instantánea del sistema de archivos `tank/home`, pero no puede crear una instantánea de su propio sistema de archivos. Así pues, no se le ha delegado el permiso `snapshot` en el nivel correcto del sistema de archivos.

```
# zfs allow -l mark snapshot tank/home
# zfs allow tank/home
```

EJEMPLO 8–3 Delegación de permisos en el nivel correcto del sistema de archivos *(Continuación)*

```
---- Permissions on tank/home --------------------------------------
Create time permissions:
        create,destroy
Local permissions:
        user mark snapshot
Local+Descendent permissions:
        group staff create,mount
# su mark
mark$ zfs snapshot tank/home@snap1
mark$ zfs snapshot tank/home/mark@snap1
cannot create snapshot 'tank/home/mark@snap1': permission denied
```

Para delegar al usuario `mark` permiso en el nivel de sistema de archivos descendiente, utilice la opción `zfs allow -d`. Por ejemplo:

```
# zfs unallow -l mark snapshot tank/home
# zfs allow -d mark snapshot tank/home
# zfs allow tank/home
---- Permissions on tank/home --------------------------------------
Create time permissions:
        create,destroy
Descendent permissions:
        user mark snapshot
Local+Descendent permissions:
        group staff create,mount
# su mark
$ zfs snapshot tank/home@snap2
cannot create snapshot 'tank/home@snap2': permission denied
$ zfs snapshot tank/home/mark@snappy
```

El usuario `mark` ahora sólo puede crear una instantánea por debajo del nivel de sistema de archivos `tank/home`.

EJEMPLO 8–4 Definición y uso de permisos delegados complejos

Puede delegar permisos a usuarios o grupos. Por ejemplo, el siguiente comando `zfs allow` delega determinados permisos al grupo `staff`. Asimismo, se delegan los permisos `destroy` y `snapshot` una vez creados los sistemas de archivos en `tank/home`.

```
# zfs allow staff create,mount tank/home
# zfs allow -c destroy,snapshot tank/home
# zfs allow tank/home
---- Permissions on tank/home --------------------------------------
Create time permissions:
        create,destroy,snapshot
Local+Descendent permissions:
        group staff create,mount
```

Debido a que el usuario `mark` es miembro del grupo `staff`, puede crear sistemas de archivos en `tank/home`. Además, el usuario `mark` puede crear una instantánea de `tank/home/mark2` porque dispone de los permisos correspondientes para hacerlo. Por ejemplo:

EJEMPLO 8–4 Definición y uso de permisos delegados complejos *(Continuación)*

```
# su mark
$ zfs create tank/home/mark2
$ zfs allow tank/home/mark2
---- Permissions on tank/home/mark2 --------------------------------
Local permissions:
        user mark create,destroy,snapshot
---- Permissions on tank/home --------------------------------------
Create time permissions:
        create,destroy,snapshot
Local+Descendent permissions:
        group staff create,mount
```

Sin embargo, el usuario `mark` no puede crear una instantánea de `tank/home/mark` porque carece de los permisos correspondientes para hacerlo. Por ejemplo:

```
$ zfs snapshot tank/home/mark@snap1
cannot create snapshot 'tank/home/mark@snap1': permission denied
```

En este ejemplo, el usuario `mark` tiene el permiso `create` en el directorio principal, lo que significa que puede crear instantáneas. Esta situación hipotética es útil si el sistema de archivos está montado por NFS.

```
$ cd /tank/home/mark2
$ ls
$ cd .zfs
$ ls
shares snapshot
$ cd snapshot
$ ls -l
total 3
drwxr-xr-x   2 mark    staff          2 Sep 27 15:55 snap1
$ pwd
/tank/home/mark2/.zfs/snapshot
$ mkdir snap2
$ zfs list
# zfs list -r tank/home
NAME                   USED  AVAIL  REFER  MOUNTPOINT
tank/home/mark          63K  62.3G    32K  /tank/home/mark
tank/home/mark2         49K  62.3G    31K  /tank/home/mark2
tank/home/mark2@snap1   18K      -    31K  -
tank/home/mark2@snap2     0      -    31K  -
$ ls
snap1  snap2
$ rmdir snap2
$ ls
snap1
```

EJEMPLO 8–5 Definición y uso de un conjunto de permisos delegados de ZFS

En el ejemplo siguiente se muestra cómo crear un conjunto de permisos `@myset` y se delega el grupo de permisos `rename` al grupo `staff` para el sistema de archivos `tank`. El usuario `cindy`, miembro del grupo `staff`, tiene permiso para crear un sistema de archivos en `tank`. Sin embargo, el usuario `lp` no tiene permiso para crear un sistema de archivos en `tank`.

EJEMPLO 8–5 Definición y uso de un conjunto de permisos delegados de ZFS *(Continuación)*

```
# zfs allow -s @myset create,destroy,mount,snapshot,promote,clone,readonly tank
# zfs allow tank
---- Permissions on tank --------------------------------------------
Permission sets:
        @myset clone,create,destroy,mount,promote,readonly,snapshot
# zfs allow staff @myset,rename tank
# zfs allow tank
---- Permissions on tank --------------------------------------------
Permission sets:
        @myset clone,create,destroy,mount,promote,readonly,snapshot
Local+Descendent permissions:
        group staff @myset,rename
# chmod A+group:staff:add_subdirectory:fd:allow tank
# su cindy
cindy% zfs create tank/data
cindy% zfs allow tank
---- Permissions on tank --------------------------------------------
Permission sets:
        @myset clone,create,destroy,mount,promote,readonly,snapshot
Local+Descendent permissions:
        group staff @myset,rename
cindy% ls -l /tank
total 15
drwxr-xr-x   2 cindy   staff          2 Jun 24 10:55 data
cindy% exit
# su lp
$ zfs create tank/lp
cannot create 'tank/lp': permission denied
```

Visualización de permisos delegados de ZFS

Para visualizar permisos puede utilizar el comando siguiente:

```
# zfs allow dataset
```

Este comando muestra los permisos que se establecen o permiten en el conjunto de datos especificado. La salida contiene los componentes siguientes:

- Conjuntos de permisos
- Permisos individuales o permisos create-time
- Conjunto de datos local
- Conjuntos de datos locales y descendientes
- Sólo conjuntos de datos descendientes

EJEMPLO 8–6 Visualización de permisos de administración delegados básicos

La siguiente salida indica que el usuario `cindy` tiene los permisos `create`, `destroy`, `mount` y snapshot en el sistema de archivos `tank/cindy`.

```
# zfs allow tank/cindy
       -------------------------------------------------------------
```

EJEMPLO 8–6 Visualización de permisos de administración delegados básicos *(Continuación)*

```
       Local+Descendent permissions on (tank/cindy)
               user cindy create,destroy,mount,snapshot
```

EJEMPLO 8–7 Visualización de permisos de administración delegados complejos

La salida de este ejemplo indica los permisos siguientes en los sistemas de archivos `pool/fred` y `pool`.

Para el sistema de archivos `pool/fred`:

- Se definen dos conjuntos de permisos:
  - `@eng` (`create`, `destroy` , `snapshot`, `mount`, `clone` , `promote`, `rename`)
  - `@simple` (`create`, `mount`)
- Los permisos create-time se establecen para el conjunto de permisos `@eng` y la propiedad `mountpoint`. Create-time significa que, tras la creación de un sistema de archivos, se delegan el conjunto de permisos `@eng` y el permiso para establecer la propiedad `mountpoint`.
- Al usuario `tom` se le delega el conjunto de permisos `@eng`; al usuario `joe` se le conceden los permisos `create`, `destroy` y `mount` para sistemas de archivos locales.
- Al usuario `fred` se le delega el conjunto de permisos `@basic`, así como los permisos `share` y `rename` para los sistemas de archivos locales y descendientes.
- Al usuario `barney` y al grupo de usuarios `staff` se les delega el grupo de permisos `@basic` sólo para sistemas de archivos descendientes.

Para el sistema de archivos `pool`:

- Se define el conjunto de permisos `@simple` (create, destroy, mount).
- Al grupo `staff` se le concede el conjunto de permisos `@simple` en el sistema de archivos local.

A continuación se muestra el resultado de este ejemplo:

```
$ zfs allow pool/fred
---- Permissions on pool/fred --------------------------------------
Permission sets:
        @eng create,destroy,snapshot,mount,clone,promote,rename
        @simple create,mount
Create time permissions:
        @eng,mountpoint
Local permissions:
        user tom @eng
        user joe create,destroy,mount
Local+Descendent permissions:
        user fred @basic,share,rename
        user barney @basic
        group staff @basic
---- Permissions on pool -------------------------------------------
Permission sets:
```

EJEMPLO 8–7 Visualización de permisos de administración delegados complejos *(Continuación)*

```
        @simple create,destroy,mount
Local permissions:
        group staff @simple
```

Eliminación de permisos delegados de ZFS (ejemplos)

El comando `zfs unallow` se usa para eliminar permisos delegados. Por ejemplo, el usuario `cindy` tiene los permisos `create`, `destroy`, `mount` y `snapshot` en el sistema de archivos `tank/cindy`.

```
# zfs allow cindy create,destroy,mount,snapshot tank/home/cindy
# zfs allow tank/home/cindy
---- Permissions on tank/home/cindy --------------------------------
Local+Descendent permissions:
        user cindy create,destroy,mount,snapshot
```

La siguiente sintaxis de `zfs unallow` elimina el permiso `snapshot` del usuario `cindy` del sistema de archivos `tank/home/cindy`:

```
# zfs unallow cindy snapshot tank/home/cindy
# zfs allow tank/home/cindy
---- Permissions on tank/home/cindy --------------------------------
Local+Descendent permissions:
        user cindy create,destroy,mount
cindy% zfs create tank/home/cindy/data
cindy% zfs snapshot tank/home/cindy@today
cannot create snapshot 'tank/home/cindy@today': permission denied
```

En otro ejemplo, el usuario `mark` tiene los siguientes permisos en el sistema de archivos `tank/home/mark`:

```
# zfs allow tank/home/mark
---- Permissions on tank/home/mark --------------------------------
Local+Descendent permissions:
        user mark create,destroy,mount
-------------------------------------------------------------
```

La siguiente sintaxis de `zfs unallow` elimina todos los permisos del usuario `mark` del sistema de archivos `tank/home/mark`:

```
# zfs unallow mark tank/home/mark
```

La siguiente sintaxis de `zfs unallow` elimina un conjunto de permisos del sistema de archivos `tank`.

```
# zfs allow tank
---- Permissions on tank --------------------------------------------
Permission sets:
```

```
        @myset clone,create,destroy,mount,promote,readonly,snapshot
Create time permissions:
        create,destroy,mount
Local+Descendent permissions:
        group staff create,mount
# zfs unallow -s @myset tank
# zfs allow tank
---- Permissions on tank --------------------------------------------
Create time permissions:
        create,destroy,mount
Local+Descendent permissions:
        group staff create,mount
```

◆ ◆ ◆ CAPÍTULO 9

Temas avanzados de Oracle Solaris ZFS

En este capítulo se describen los volúmenes de ZFS, el uso de ZFS en un sistema Solaris con zonas instaladas, agrupaciones raíz alternativas de ZFS y perfiles de derechos de ZFS.

Este capítulo se divide en las secciones siguientes:

Volúmenes de ZFS

Un volumen ZFS es un conjunto de datos que representa un dispositivo de bloques. Los volúmenes de ZFS se identifican como dispositivos en el directorio `/dev/zvol/{dsk,rdsk}/pool`.

En el ejemplo siguiente, se crea un volumen de ZFS de 5 GB, `tank/vol`:

```
# zfs create -V 5gb tank/vol
```

Al crear un volumen, automáticamente se reserva espacio para el tamaño inicial del volumen, a fin de evitar imprevistos. Por ejemplo, si disminuye el tamaño del volumen, los datos podrían dañarse. El cambio del volumen se debe realizar con mucho cuidado.

Además, si crea una instantánea de un volumen que cambia de tamaño, podría provocar incoherencias en el sistema de archivos al intentar restaurar una versión anterior de la instantánea o crear un clon de ésta.

Para obtener información sobre las propiedades de sistemas de archivos que se pueden aplicar a volúmenes, consulte la Tabla 5–1.

Puede mostrar información de las propiedades del volumen ZFS mediante los comandos `zfs get` o `zfs get all`. Por ejemplo:

```
# zfs get all tank/vol
```

El signo de interrogación (?) que se muestra para `volsize` en la salida de `zfs get` indica que hay un valor desconocido porque se ha producido un error de E/S. Por ejemplo:

```
# zfs get -H volsize tank/vol
tank/vol        volsize ?       local
```

En general, un error de E/S indica un problema con un dispositivo de agrupación. Para obtener información sobre la resolución de problemas de dispositivos de agrupación, consulte "Solución de problemas con ZFS" en la página 303.

Si utiliza un sistema Solaris con zonas instaladas, los volúmenes de ZFS no se pueden crear ni clonar en una zona no global. Cualquier intento de hacerlo, fallará. Para obtener información sobre el uso de volúmenes de ZFS en una zona global, consulte "Agregación de volúmenes de ZFS a una zona no global" en la página 291.

Uso de un volumen de ZFS como dispositivo de volcado o intercambio

Durante la instalación de un sistema de archivos raíz ZFS o una migración desde un sistema de archivos raíz UFS, se crea un dispositivo de intercambio en un volumen ZFS de la agrupación raíz ZFS. Por ejemplo:

```
# swap -l
swapfile                  dev    swaplo   blocks     free
/dev/zvol/dsk/rpool/swap 253,3        16  8257520  8257520
```

Durante la instalación de un sistema de archivos raíz ZFS o una migración desde un sistema de archivos raíz UFS, se crea un dispositivo de volcado en un volumen ZFS de la agrupación raíz ZFS. Después de configurarse, no hace falta administrar el dispositivo de volcado. Por ejemplo:

```
# dumpadm
      Dump content: kernel pages
       Dump device: /dev/zvol/dsk/rpool/dump (dedicated)
Savecore directory: /var/crash/
  Savecore enabled: yes
```

Si necesita cambiar el área de intercambio o el dispositivo de volcado después de instalar el sistema, utilice los comandos `swap` y `dumpadm` como en las versiones anteriores de Solaris. Para crear un área de intercambio adicional, cree un volumen ZFS de un tamaño específico y permita el intercambio en dicho dispositivo. Luego, agregue una entrada para el nuevo dispositivo de intercambio en el archivo `/etc/vfstab`. Por ejemplo:

```
# zfs create -V 2G rpool/swap2
# swap -a /dev/zvol/dsk/rpool/swap2
# swap -l
```

```
swapfile                   dev  swaplo blocks   free
/dev/zvol/dsk/rpool/swap  256,1      16 2097136 2097136
/dev/zvol/dsk/rpool/swap2 256,5      16 4194288 4194288
```

No intercambie a un archivo en un sistema de archivos ZFS. La configuración de archivos de intercambio ZFS no es posible.

Para obtener información sobre cómo ajustar el tamaño de los volúmenes de intercambio y volcado, consulte "Ajuste del tamaño de los dispositivos de intercambio y volcado ZFS" en la página 136.

Uso de un volumen de ZFS como un LUN iSCSI

La estructura de software Common Multiprotocol SCSI Target (COMSTAR) permite convertir cualquier host Oracle Solaris en un dispositivo de destino SCSI al que se puede acceder a través de una red de almacenamiento por hosts iniciadores. Puede crear y configurar un volumen de ZFS para compartir como una unidad lógica iSCSI (LUN).

Primero, instale el paquete COMSTAR.

```
# pkg install group/feature/storage-server
```

A continuación, cree un volumen de ZFS para utilizar como un destino iSCSI y, luego, cree la LUN basada en dispositivo de bloqueo SCSI. Por ejemplo:

```
# zfs create -V 2g tank/volumes/v2
# sbdadm create-lu /dev/zvol/rdsk/tank/volumes/v2
Created the following LU:

              GUID                    DATA SIZE           SOURCE
--------------------------------  -------------------  ----------------
600144f000144f1dafaa4c0faff20001  2147483648           /dev/zvol/rdsk/tank/volumes/v2
# sbdadm list-lu
Found 1 LU(s)

              GUID                    DATA SIZE           SOURCE
--------------------------------  -------------------  ----------------
600144f000144f1dafaa4c0faff20001  2147483648           /dev/zvol/rdsk/tank/volumes/v2
```

Puede mostrar las vistas de LUN a todos los clientes o a los clientes seleccionados. Identifique el valor GUID de LUN y luego comparta la vista de LUN. En el siguiente ejemplo, la vista de LUN se comparte con todos los clientes.

```
# stmfadm list-lu
LU Name: 600144F000144F1DAFAA4C0FAFF20001
# stmfadm add-view 600144F000144F1DAFAA4C0FAFF20001
# stmfadm list-view -l 600144F000144F1DAFAA4C0FAFF20001
View Entry: 0
    Host group   : All
    Target group : All
    LUN          : 0
```

El siguiente paso es crear los destinos de iSCSI. Para obtener información sobre la creación de destinos iSCSI, consulte el Capítulo 11, "Configuración de dispositivos de almacenamiento con COMSTAR (tareas)" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.

Un volumen de ZFS como un destino de iSCSI se gestiona como cualquier otro conjunto de datos ZFS, excepto que no se puede cambiar el nombre del conjunto de datos, revertir una instantánea de volumen ni exportar la agrupación mientras los volúmenes de ZFS se comparten como LUN de iSCSI. Verá mensajes similares a los siguientes:

```
# zfs rename tank/volumes/v2 tank/volumes/v1
cannot rename 'tank/volumes/v2': dataset is busy
# zpool export tank
cannot export 'tank': pool is busy
```

Toda la información de configuración de objetivos iSCSI se guarda con el conjunto de datos. Al igual que un sistema de archivos NFS compartido, un objetivo iSCSI que se importa a otro sistema se comparte correspondientemente.

Uso de ZFS en un sistema Solaris con zonas instaladas

Las secciones siguientes explican cómo utilizar ZFS en un sistema con zonas de Oracle Solaris:

Al asociar conjuntos de datos de ZFS con zonas, hay que tener en cuenta los puntos siguientes:

- Puede agregar un sistema de archivos o un clon de ZFS a una zona no global con o sin delegación de control administrativo.
- Puede agregar un volumen de ZFS como dispositivo a zonas no globales.
- Por ahora no es posible asociar instantáneas de ZFS con zonas.

En las secciones siguientes, un conjunto de datos de ZFS hace referencia a un sistema de archivos o un clon.

La agregación de un conjunto de datos permite que la zona no global comparta espacio en el disco con la zona global, si bien el administrador de zona no puede controlar las propiedades ni crear sistemas de archivos en la jerarquía de sistemas de archivos subyacente. Es lo mismo que agregar cualquier otro sistema de archivos a una zona; es aconsejable utilizarlo si la finalidad principal es compartir espacio.

ZFS permite también la delegación de conjuntos de datos a una zona no global, con lo cual el administrador de zona dispone de control absoluto sobre el conjunto de datos y todos sus conjuntos de datos secundarios. El administrador de zona puede crear y destruir sistemas de archivos o clones de ese conjunto de datos, así como modificar las propiedades de los conjuntos de datos. El administrador de zona no puede incidir en los conjuntos de datos que no se hayan agregado a la zona ni sobrepasar las cuotas de nivel superior establecidas en el conjunto de datos delegado.

Al utilizar ZFS en un sistema con zonas Oracle Solaris instaladas hay que tener en cuenta los puntos siguientes:

- Un sistema de archivos ZFS agregado a una zona no global debe tener la propiedad `mountpoint` establecida en `legacy`.
- Cuando tanto un origen `zonepath` como un destino `zonepath` residen en un sistema de archivos ZFS y se encuentran en la misma agrupación, `zoneadm clone` utiliza automáticamente el clon de ZFS para clonar una zona. El comando `zoneadm clone` crea una instantánea de ZFS de la `zonepath` de origen y configura la `zonepath` de destino. No puede utilizar el comando `zfs clone` para clonar una zona. Para obtener más información, consulte la Parte II, "Zonas de Oracle Solaris" de *Administración de Oracle Solaris: zonas de Oracle Solaris, zonas de Oracle Solaris 10 y gestión de recursos*.

Agregación de sistemas de archivos ZFS a una zona no global

Un sistema de archivos ZFS se puede agregar como sistema de archivos genérico si la finalidad es compartir espacio con la zona global. Un sistema de archivos ZFS agregado a una zona no global debe tener la propiedad `mountpoint` establecida en `legacy`. Por ejemplo, si el sistema de archivos `tank/zone/zion` se agregará a una zona no global, establezca la propiedad `mountpoint` en la zona global como se indica a continuación:

```
# zfs set mountpoint=legacy tank/zone/zion
```

Un sistema de archivos ZFS puede agregarse a una zona no global mediante el comando `zonecfg` y el subcomando `add fs`.

En el ejemplo siguiente, un administrador de zona global agrega a la zona no global un sistema de archivos ZFS desde la zona global:

```
# zonecfg -z zion
zonecfg:zion> add fs
zonecfg:zion:fs> set type=zfs
zonecfg:zion:fs> set special=tank/zone/zion
zonecfg:zion:fs> set dir=/opt/data
zonecfg:zion:fs> end
```

Esta sintaxis agrega el sistema de archivos ZFS, `tank/zone/zion`, a la zona `zion` ya configurada, que está montada en `/opt/data`. La propiedad `mountpoint` del sistema de archivos se debe establecer en `legacy` y el sistema de archivos ya no se puede montar en otra ubicación. El administrador de zonas puede crear y destruir archivos en el sistema de archivos. El sistema de archivos no se puede volver a montar en una ubicación distinta; el administrador de zona tampoco puede modificar propiedades del sistema de archivos, por ejemplo `atime`, `readonly` o `compression`. El administrador de zona global se encarga de configurar y controlar las propiedades del sistema de archivos.

Para obtener más información sobre el comando `zonecfg` y la configuración de tipos de recursos con `zonecfg`, consulte Parte II, "Zonas de Oracle Solaris" de *Administración de Oracle Solaris: zonas de Oracle Solaris, zonas de Oracle Solaris 10 y gestión de recursos*.

Delegación de conjuntos de datos a una zona no global

Para cumplir el objetivo principal, que es delegar la administración del almacenamiento a una zona, ZFS permite agregar conjuntos de datos a una zona no global mediante el comando `zonecfg` y el subcomando `add dataset`.

En el ejemplo siguiente, un administrador de zona global delega a la zona no global un sistema de archivos ZFS desde la zona global:

```
# zonecfg -z zion
zonecfg:zion> add dataset
zonecfg:zion:dataset> set name=tank/zone/zion
zonecfg:zion:dataset> set alias=tank
zonecfg:zion:dataset> end
```

A diferencia de agregar un sistema de archivos, esta sintaxis hace que el sistema de archivos ZFS `tank/zone/zion` quede visible en la zona ya configurada `zion`. Dentro de la zona `zion`, no se puede acceder a este sistema de archivos como `tank/zone/zion`, sino como una *agrupación virtual* denominada `tank`. El alias del sistema de archivos delegado proporciona una vista de la agrupación original para la zona como una agrupación virtual. La propiedad del alias especifica el nombre de la agrupación virtual. Si no se especifica ningún alias, se utiliza un alias predeterminado que coincide con el último componente del nombre del sistema de archivos. Si no se proporciona un alias específico, el alias predeterminado en el ejemplo anterior habría sido `zion`.

En conjuntos de datos delegados, el administrador de zona puede establecer las propiedades del sistema de archivos, así como crear sistemas de archivos descendientes. Además, puede crear instantáneas y clones, y controlar toda la jerarquía del sistema de archivos. Si los volúmenes de ZFS se crean en sistemas de archivos delegados, es posible que entren en conflicto con los volúmenes de ZFS que se agregan como recursos de dispositivos. Para obtener más información, consulte la siguiente sección y `dev(7FS)`.

Agregación de volúmenes de ZFS a una zona no global

Puede agregar o crear un volumen de ZFS en una zona no global o puede agregar un acceso a los datos de un volumen en una zona no global de las siguientes formas:

- En una zona no global, un administrador de zona privilegiada puede crear un volumen de ZFS como descendiente de un sistema de archivos previamente delegado. Por ejemplo:

  ```
  # zfs create -V 2g tank/zone/zion/vol1
  ```

  La sintaxis anterior significa que el administrador de zona puede gestionar las propiedades y los datos del volumen en la zona no global.

- En una zona global, utilice el subcomando `zonecfg add dataset` y especifique un volumen de ZFS para agregar a una zona no global. Por ejemplo:

  ```
  # zonecfg -z zion
  zonecfg:zion> add dataset
  zonecfg:zion:dataset> set name=tank/volumes/vol1
  zonecfg:zion:dataset> end
  ```

  La sintaxis anterior significa que el administrador de zona puede gestionar las propiedades y los datos del volumen en la zona no global.

- En una zona global, utilice el subcomando `zonecfg add device` y especifique un volumen de ZFS a cuyos datos se pueda acceder en una zona no global. Por ejemplo:

  ```
  # zonecfg -z zion
  zonecfg:zion> add device
  zonecfg:zion:device> set match=/dev/zvol/dsk/tank/volumes/vol2
  zonecfg:zion:device> end
  ```

  La sintaxis anterior significa que solamente se puede acceder a los datos del volumen en la zona no global.

Uso de grupos de almacenamiento de ZFS en una zona

Las agrupaciones de almacenamiento de ZFS no se pueden crear ni modificar en una zona. El modelo de administración delegada centraliza el control de dispositivos de almacenamiento físicos en la zona global y el control de almacenamiento virtual en zonas no globales. Aunque un conjunto de datos de agrupación se puede agregar a una zona, en una zona no se permite ningún comando que modifique las características físicas de la agrupación, por ejemplo crear, agregar o eliminar dispositivos. Aunque se agreguen dispositivos físicos a una zona mediante el comando `zonecfg` y el subcomando `add device`, o aunque se utilicen archivos, el comando `zpool` no permite la creación de grupos en la zona.

Administración de propiedades de ZFS en una zona

Tras delegar un conjunto de datos a una zona, el administrador de zona puede controlar determinadas propiedades del conjunto. Cuando un conjunto de datos se delega a una zona, todos sus antecesores se ven como conjuntos de datos de sólo lectura, mientras que el conjunto de datos agregado y todos sus descendientes se pueden escribir. Por ejemplo, tenga en cuenta la configuración siguiente:

```
global# zfs list -Ho name
tank
tank/home
tank/data
tank/data/matrix
tank/data/zion
tank/data/zion/home
```

Si se agregara `tank/data/zion` a una zona con el alias `zion` predeterminado, cada conjunto de datos tendría las siguientes propiedades.

Conjunto de datos	Visible	Escribible	Propiedades invariables
`tank`	No	-	-
`tank/home`	No	-	-
`tank/data`	No	-	-
`tank/data/zion`	Sí	Sí	`zoned`, `quota`, `reservation`
`tank/data/zion/home`	Sí	Sí	`zoned`

Tenga en cuenta que cada principal de `tank/zone/zion` es invisible y que puede escribirse en todos los descendientes. El administrador de zona no puede cambiar la propiedad `zoned` ya que existiría riesgo en la seguridad, como se explica en la sección siguiente.

Los usuarios con privilegios en la zona pueden cambiar otras propiedades configurables, excepto `quota` y `reservation`. Este comportamiento permite que el administrador de zona global controle la ocupación de espacio en el disco de todos los conjuntos de datos utilizados por la zona no global.

Además, el administrador de la zona global no puede modificar las propiedades `share.nfs` y `mountpoint` después de la delegación de un conjunto de datos a una zona no global.

Interpretación de la propiedad zoned

Si un conjunto de datos se delega a una zona no global, se debe marcar de modo especial para que determinadas propiedades no se interpreten en el contexto de la zona global. Tras haber delegado un conjunto de datos a una zona no global bajo el control de un administrador de zona, su contenido deja de ser fiable. Como en cualquier sistema de archivos, puede haber binarios `setuid`, vínculos simbólicos o contenido dudoso que podría repercutir negativamente en la seguridad de la zona global. Además, la propiedad `mountpoint` no se puede interpretar en el contexto de la zona global. Por otro lado, el administrador de zona podría afectar al espacio de nombre de la zona global. Para ocuparse de esto último, ZFS utiliza la propiedad `zoned` para indicar que un conjunto de datos se ha delegado a una zona no global en un determinado momento.

La propiedad `zoned` consiste en un valor booleano que se activa automáticamente la primera vez que se inicia una zona que contiene un conjunto de datos de ZFS. Un administrador de zona no tiene necesidad de activar manualmente esta propiedad. Si se establece la propiedad `zoned`, el conjunto de datos no se puede montar ni compartir en la zona global. En el ejemplo siguiente, `tank/zone/zion` se ha delegado a una zona y `tank/zone/global` no se ha delegado:

```
# zfs list -o name,zoned,mountpoint -r tank/zone
NAME                  ZONED  MOUNTPOINT
tank/zone/global        off  /tank/zone/global
tank/zone/zion           on  /tank/zone/zion
# zfs mount
tank/zone/global           /tank/zone/global
tank/zone/zion             /export/zone/zion/root/tank/zone/zion
```

Observe la diferencia entre la propiedad `mountpoint` y el directorio en que está montado el conjunto de datos `tank/zone/zion.` La propiedad `mountpoint` refleja la propiedad como almacenada en disco, no donde el conjunto de datos está montado en el sistema.

Si se elimina un conjunto de datos de una zona o se destruye una zona, la propiedad `zoned` *no* se elimina de forma automática. Este comportamiento se debe a los riegos de seguridad inherentes a estas tareas. Debido a que un usuario que no es de confianza dispone de acceso completo al conjunto de datos y sus descendientes, la propiedad `mountpoint` podría definirse con valores incorrectos o podría haber binarios `setuid` en los sistemas de archivos.

Para prevenir riesgos en la seguridad, el administrador de zona global debe suprimir manualmente la propiedad `zoned` si se desea volver a utilizar el conjunto de datos. Antes de establecer la propiedad `zoned` en `off`, compruebe que la propiedad `mountpoint` del conjunto de datos y todos sus descendientes tengan valores razonables y que no haya binarios `setuid`, o desactive la propiedad `setuid`.

Tras haber comprobado que no queden puntos débiles en la seguridad, la propiedad `zoned` se puede desactivar mediante los comandos `zfs set` o `zfs inherit`. Si la propiedad `zoned` se desactiva mientras un conjunto de datos se utiliza en una zona, el sistema podría manifestar un comportamiento impredecible. La propiedad se debe modificar únicamente si se tiene la certeza de que ninguna zona no global no está utilizando el conjunto de datos.

Copia de zonas a otros sistemas

Cuando deba migrar las necesidades de una o varias zonas a otro sistema, considere el uso de los comandos `zfs send` y `zfs receive`. Según el escenario, puede ser mejor utilizar flujos de replicación o flujos recursivos.

Los ejemplos de esta sección describen cómo copiar datos de zona entre los sistemas. Se necesitan pasos adicionales para transferir la configuración de cada zona y conectar cada zona al nuevo sistema. Para obtener más información, consulte la Parte II, "Zonas de Oracle Solaris" de *Administración de Oracle Solaris 11.1: zonas de Oracle Solaris, zonas de Oracle Solaris 10 y gestión de recursos*.

Si todas las zonas de un sistema deben moverse a otro sistema, considere utilizar un flujo de replicación para que se preserven las instantáneas y los clones. Las instantáneas y los clones son utilizados en gran medida por los comandos `pkg update`, `beadm create` y `zoneadm clone`.

En el siguiente ejemplo, las zonas de `sysA` se instalan en el sistema de archivos `rpool/zones` y deben copiarse en el sistema de archivos `tank/zones`, en `sys`. Los siguientes comandos crean una instantánea y copian los datos en `sysB` mediante un flujo de replicación:

```
sysA# zfs snapshot -r rpool/zones@send-to-sysB
sysA# zfs send -R rpool/zones@send-to-sysB | ssh sysB zfs receive -d tank
```

En el siguiente ejemplo, una de varias zonas de `sysC` se copia en `sysD`. Supongamos que el comando `ssh` no está disponible, pero se encuentra disponible una instancia del servidor NFS. Es posible utilizar los siguientes comandos para generar un flujo recursivo `zfs send` sin necesidad de preocuparse por si la zona es un clon de otra zona.

```
sysC# zfs snapshot -r rpool/zones/zone1@send-to-nfs
sysC# zfs send -rc rpool/zones/zone1@send-to-nfs > /net/nfssrv/export/scratch/zone1.zfs
sysD# zfs create tank/zones
sysD# zfs receive -d tank/zones < /net/nfssrv/export/scratch/zone1.zfs
```

Uso de agrupaciones raíz de ZFS alternativas

Cuando se crea una agrupación, queda intrínsecamente vinculada con el sistema host. El sistema host mantiene información sobre la agrupación para detectar si está disponible o no. Si bien es útil en el funcionamiento normal, esta información puede suponer un obstáculo al iniciar desde otro soporte o al crear una agrupación en un soporte extraíble. Para solucionar este problema, ZFS proporciona una función de agrupaciones *raíz alternativas*. Una agrupación raíz alternativa no se mantiene de un reinicio del sistema a otro, y todos los puntos de montaje se modifican para vincularse con la raíz de la agrupación.

Creación de agrupaciones raíz de ZFS alternativas

La finalidad más habitual por la que se crea una agrupación raíz alternativa es utilizarla con soportes extraíbles. En estos casos, los usuarios suelen preferir un solo sistema de archivos, montado en algún lugar seleccionado en el sistema de destino. Si se crea una agrupación raíz alternativa mediante la opción `zpool create -R`, el punto de montaje del sistema de archivos raíz se establece automáticamente en `/`, que es el equivalente del valor raíz alternativo.

En el ejemplo siguiente, un grupo denominado `morpheus` se crea con `/mnt` como ruta de acceso root alternativa:

```
# zpool create -R /mnt morpheus c0t0d0
# zfs list morpheus
NAME                  USED  AVAIL  REFER  MOUNTPOINT
morpheus             32.5K  33.5G     8K  /mnt
```

Observe el sistema de archivos único `morpheus`, cuyo punto de montaje es el root alternativo del grupo, `/mnt`. El punto de montaje que se guarda en disco es `/` y la ruta completa de `/mnt` sólo se interpreta en el contexto inicial de creación de la agrupación. Este sistema de archivos se puede exportar e importar bajo una agrupación raíz alternativa arbitraria en otro sistema, mediante sintaxis de *valor raíz alternativo* `-R`.

```
# zpool export morpheus
# zpool import morpheus
cannot mount '/': directory is not empty
# zpool export morpheus
# zpool import -R /mnt morpheus
# zfs list morpheus
NAME                  USED  AVAIL  REFER  MOUNTPOINT
morpheus             32.5K  33.5G     8K  /mnt
```

Importación de agrupaciones raíz alternativas

Las agrupaciones también se pueden importar mediante una raíz alternativa. Esta función posibilita situaciones de recuperación en que los puntos de montaje no se deben interpretar en el contexto del root actual, sino en determinados directorios temporales en los que se pueden efectuar reparaciones. Esta función también es apta para montar soportes extraíbles como se ha explicado anteriormente.

En el ejemplo siguiente, un grupo denominado `morpheus` se importa con `/mnt` como ruta de acceso root alternativa: En este ejemplo se da por sentado que `morpheus` se ha exportado previamente.

```
# zpool import -R /a pool
# zpool list morpheus
NAME   SIZE   ALLOC  FREE    CAP  HEALTH  ALTROOT
pool  44.8G    78K  44.7G     0%  ONLINE  /a
# zfs list pool
```

```
NAME   USED  AVAIL  REFER  MOUNTPOINT
pool  73.5K  44.1G    21K  /a/pool
```

◆ ◆ ◆ CAPÍTULO 10

Recuperación de agrupaciones y solución de problemas de Oracle Solaris ZFS

En este capítulo se explica la forma de identificar y solucionar errores de ZFS. También se proporciona información para la prevención de errores.

Este capítulo se divide en las secciones siguientes:

Para obtener información sobre la recuperación de agrupaciones raíz completas, consulte el Capítulo 11, "Archivado de instantáneas y recuperación de agrupaciones raíz".

Resolución de problemas de espacio ZFS

Revise las siguientes secciones si no está seguro de cómo ZFS informa la contabilización del sistema de archivos y el espacio de agrupación. También revise "Cálculo del espacio de ZFS" en la página 34.

Informes de espacio del sistema de archivos

Los comandos `zpool list` y `zfs list` son mejores que los comandos `df` y `du` anteriores para determinar el espacio disponible de la agrupación y el sistema de archivos. Con los comandos heredados, no se puede distinguir fácilmente entre el espacio disponible de la agrupación y el

del sistema de archivos. Además, los comandos heredados no contabilizan el espacio que consumen los sistemas de archivos descendientes o las instantáneas.

Por ejemplo, la siguiente agrupación raíz (`rpool`) tiene 5,46 GB asignados y 68,5 GB libres.

```
# zpool list rpool
NAME   SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
rpool   74G  5.46G  68.5G   7%  1.00x  ONLINE  -
```

Si compara la contabilización del espacio de la agrupación con la contabilización del espacio del sistema de archivos revisando la columna `USED` de los sistemas de archivos individuales, puede ver que el espacio de la agrupación informado en `ALLOC` está contabilizado en el total de `USED` de los sistemas de archivos. Por ejemplo:

```
# zfs list -r rpool
NAME                      USED  AVAIL  REFER  MOUNTPOINT
rpool                    5.41G  67.4G  74.5K  /rpool
rpool/ROOT               3.37G  67.4G    31K  legacy
rpool/ROOT/solaris       3.37G  67.4G  3.07G  /
rpool/ROOT/solaris/var    302M  67.4G   214M  /var
rpool/dump               1.01G  67.5G  1000M  -
rpool/export             97.5K  67.4G    32K  /rpool/export
rpool/export/home        65.5K  67.4G    32K  /rpool/export/home
rpool/export/home/admin  33.5K  67.4G  33.5K  /rpool/export/home/admin
rpool/swap               1.03G  67.5G  1.00G  -
```

Informes de espacio de la agrupación de almacenamiento ZFS

El valor de tamaño (SIZE) que informa el comando `zpool list` en general es la cantidad de espacio físico en disco de la agrupación, pero esto varía según el nivel de redundancia de la agrupación. Consulte los ejemplos que se proporcionan a continuación. El comando `zfs list` muestra el espacio utilizable que está disponible para sistemas de archivos, que se calcula con el espacio en disco menos la carga de metadatos de redundancia de la agrupación ZFS, si es que hay.

- **Agrupación de almacenamiento no redundante**: cuando una agrupación se crea con un disco de 136 GB, el comando `zpool list` informa `SIZE` y los valores iniciales de `FREE` como 136 GB. El espacio inicial de `AVAIL` informado por el comando `zfs list` es de 134 GB, debido a una pequeña cantidad de carga de metadatos de la agrupación. Por ejemplo:

  ```
  # zpool create tank c0t6d0
  # zpool list tank
  NAME   SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
  tank   136G  95.5K   136G     0%  1.00x  ONLINE  -
  # zfs list tank
  NAME   USED  AVAIL  REFER  MOUNTPOINT
  tank    72K   134G    21K  /tank
  ```

- **Agrupación de almacenamiento reflejada**: cuando una agrupación se crea con dos discos de 136 GB, el comando `zpool list` informa `SIZE` como 136 GB y el valor inicial de `FREE` como 136 GB. Este informe se denomina valor de espacio *desinflado*. El espacio inicial de `AVAIL` informado por el comando `zfs list` es de 134 GB, debido a una pequeña cantidad de carga de metadatos de la agrupación. Por ejemplo:

```
# zpool create tank mirror c0t6d0 c0t7d0
# zpool list tank
NAME   SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
tank   136G  95.5K   136G     0%  1.00x  ONLINE  -
# zfs list tank
NAME   USED  AVAIL  REFER  MOUNTPOINT
tank    72K   134G    21K  /tank
```

- **Agrupación de almacenamiento RAID-Z**: cuando una agrupación `raidz2` se crea con tres discos de 136 GB, el comando `zpool list` informa `SIZE` como 408 GB y el valor inicial de `FREE` como 408 GB. Este informe se conoce como valor de espacio en disco *inflado*, que incluye carga de redundancia, como la información de paridad. El espacio inicial de `AVAIL` informado por el comando `zfs list` es de 133 GB, debido a la carga de redundancia de la agrupación. La diferencia de espacio entre la salida de `zpool list` y `zfs list` para una agrupación RAID-Z se debe a que `zpool list` informa el espacio de agrupación aumentado.

```
# zpool create tank raidz2 c0t6d0 c0t7d0 c0t8d0
# zpool list tank
NAME   SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
tank   408G   286K   408G     0%  1.00x  ONLINE  -
# zfs list tank
NAME   USED  AVAIL  REFER  MOUNTPOINT
tank  73.2K   133G  20.9K  /tank
```

Identificación de errores de ZFS

Como combinación de sistema de archivos y administrador de volúmenes, ZFS puede presentar una amplia modalidad de errores. Este capítulo comienza con una breve introducción de los diversos errores y posteriormente explica el modo de identificarlos en un sistema que está en funcionamiento. Al final del capítulo, se proporcionan instrucciones para solucionar los problemas. ZFS puede tener tres tipos básicos de errores:

- "Dispositivos que faltan en un grupo de almacenamiento de ZFS" en la página 300
- "Dispositivos dañados de un grupo de almacenamiento de ZFS" en la página 300
- "Datos dañados de ZFS" en la página 300

En una misma agrupación se pueden dar los tres errores, con lo cual un procedimiento completo de reparación implica detectar y corregir un error, luego ocuparse del siguiente error y así sucesivamente.

Dispositivos que faltan en un grupo de almacenamiento de ZFS

Si un dispositivo ha desaparecido totalmente del sistema, ZFS detecta que dicho dispositivo no se puede abrir y le asigna el estado `REMOVED`. Según el nivel de repetición de datos que tenga la agrupación, la desaparición no tiene por qué significar que toda la agrupación deje de estar disponible. Si se elimina un disco de un dispositivo RAID-Z o reflejado, la agrupación sigue estando disponible. Es posible que una agrupación tenga el estado `UNAVAIL`, lo que significa que no se puede acceder a los datos hasta que se vuelva a conectar el dispositivo, en las siguientes condiciones:

- Si se eliminan todos los componentes de un reflejo
- Si se elimina más de un dispositivo en un RAID-Z (`raidz1`)
- Si se elimina un dispositivo de nivel superior en una configuración de un solo disco

Dispositivos dañados de un grupo de almacenamiento de ZFS

El término "dañado" se aplica a una amplia diversidad de errores. Entre otros, están los errores siguientes:

- Errores transitorios de E/S debido a discos o controladores incorrectos
- Datos en disco dañados por rayos cósmicos
- Errores de controladores debidos a datos que se transfieren o reciben de ubicaciones incorrectas
- Anulación involuntaria de partes del dispositivo físico por parte de un usuario

En determinados casos, estos errores son transitorios, por ejemplo errores aleatorios de E/S mientras el controlador tiene problemas. En otros, las consecuencias son permanentes, por ejemplo la corrupción del disco. Aun así, el hecho de que los daños sean permanentes no implica necesariamente que el error se repita más adelante. Por ejemplo, si sobrescribe involuntariamente parte de un disco, no se ha producido ningún error de hardware y no hace falta reemplazar el dispositivo. No resulta nada fácil identificar con exactitud lo que ha sucedido en un dispositivo. Ello se aborda en mayor profundidad más adelante en otra sección.

Datos dañados de ZFS

El deterioro de datos tiene lugar cuando uno o varios errores de dispositivos (dañados o que faltan) afectan a un dispositivo virtual de nivel superior. Por ejemplo, la mitad de un reflejo puede sufrir innumerables errores sin causar la más mínima corrupción de datos. Si se detecta un error en la otra parte del reflejo, en la misma ubicación exacta, se producirán datos dañados como resultado.

Los datos quedan permanentemente dañados y deben tratarse de forma especial durante la reparación. Aunque se reparen o reemplacen los dispositivos subyacentes, los datos originales se pierden irremisiblemente. En estas circunstancias, casi siempre se requiere la restauración de datos a partir de copias de seguridad. Los errores de datos se registran conforme se detectan. Como se explica en la sección siguiente, pueden controlarse mediante limpiezas de agrupación rutinarias. Si se quita un bloque dañado, el siguiente pase de limpieza reconoce que el deterioro ya no está presente y suprime del sistema cualquier indicio de error.

Comprobación de integridad de sistema de archivos ZFS

En ZFS no hay una utilidad `fsck` equivalente. Esta utilidad se ha venido utilizando con dos fines: para reparaciones de sistema de archivos y para validaciones de dichos sistemas.

Reparación de sistema de archivos

En los sistemas de archivos tradicionales, el método de escritura de datos es intrínsecamente vulnerable a errores imprevistos que generan incoherencias en el sistema. Debido a que un sistema de archivos tradicional no es transaccional, puede haber bloques sin referenciar, recuentos de vínculos erróneos u otras estructuras de sistema de archivos no coherentes. La agregación de diarios soluciona algunos de estos problemas, pero puede presentar otros problemas si el registro no se puede invertir. La existencia de datos incoherentes en el disco de una configuración ZFS sólo puede ser debida a un error de hardware (en cuyo caso, la agrupación debería haber sido redundante) o porque hay un error en el software de ZFS.

La utilidad `fsck` soluciona problemas conocidos específicos de sistemas de archivos UFS. Casi todos los problemas de agrupación de almacenamiento ZFS suelen estar relacionados con errores de hardware o fallos de alimentación. Muchos se pueden evitar utilizando agrupaciones redundantes. Si una agrupación se ha dañado por un error de hardware o un fallo de alimentación, consulte "Reparación de daños en las agrupaciones de almacenamiento de ZFS" en la página 326.

Si la agrupación no es redundante, siempre existe el riesgo de que los daños en el sistema de archivos lleguen a hacer que parte o todos los datos queden inaccesibles.

Validación de sistema de archivos

Aparte de reparar sistemas de archivos, la utilidad `fsck` comprueba que los datos en disco no tengan problemas. El procedimiento habitual para esta tarea consiste en desmontar el sistema de archivos y ejecutar la utilidad `fsck`, seguramente con el sistema en modo monousuario durante el proceso. Esta situación da como resultado un tiempo de inactividad proporcional al tamaño del sistema de archivos que se comprueba. En lugar de hacer que una determinada

utilidad realice la comprobación pertinente, ZFS brinda un mecanismo para ejecutar una comprobación rutinaria de todas las incoherencias. Esta función, denominada *limpieza*, se suele utilizar en la memoria y en otros sistemas como método para detectar y evitar errores antes de que deriven en errores de hardware o software.

Control de la limpieza de datos de ZFS

Cuando ZFS detecta un error, ya sea mediante el proceso de limpieza o al acceder a un archivo por algún motivo, el error se registra internamente para poder disponer de una visión general inmediata de todos los errores conocidos de la agrupación.

Limpieza explícita de datos de ZFS

La forma más sencilla de comprobar la integridad de los datos es ejecutar una limpieza explícita de todos los datos de la agrupación. Este proceso afecta a todos los datos del grupo y verifica que se puedan leer todos los bloques. El proceso de limpieza transcurre todo lo deprisa que permiten los dispositivos, aunque la prioridad de cualquier E/S quede por debajo de las operaciones normales. Esta operación puede incidir negativamente en el rendimiento, aunque los datos de la agrupación deberían seguir siendo utilizables casi del modo habitual. Para iniciar una limpieza explícita, utilice el comando `zpool scrub`. Por ejemplo:

```
# zpool scrub tank
```

El estado de la limpieza actual puede verse mediante el comando `zpool status`. Por ejemplo:

```
# zpool status -v tank
  pool: tank
 state: ONLINE
 scan: scrub in progress since Mon Jun  7 12:07:52 2010
    201M scanned out of 222M at 9.55M/s, 0h0m to go
    0 repaired, 90.44% done
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c1t0d0  ONLINE       0     0     0
            c1t1d0  ONLINE       0     0     0

errors: No known data errors
```

Sólo puede haber una operación de limpieza activa por agrupación.

Con la opción `-s` se puede detener una operación de limpieza en curso. Por ejemplo:

```
# zpool scrub -s tank
```

En la mayoría de los casos, una operación de limpieza para asegurar la integridad de los datos debe continuar hasta finalizar. Si cree que la limpieza afecta negativamente al rendimiento del sistema, puede detenerla.

La ejecución rutinaria de limpiezas garantiza la E/S continua en todos los discos del sistema. La ejecución rutinaria de limpiezas tiene el inconveniente de impedir que los discos inactivos pasen a la modalidad de bajo consumo. Si en general el sistema efectúa E/S permanentemente, o si el consumo de energía no es ningún problema, se puede prescindir de este tema.

Para obtener más información sobre la interpretación de la salida de `zpool status`, consulte "Consulta del estado de una agrupación de almacenamiento de ZFS" en la página 93.

Limpieza y actualización de la duplicación de datos de ZFS

Al reemplazar un dispositivo, se inicia una operación de actualización de duplicación de datos para transferir datos de las copias correctas al nuevo dispositivo. Este proceso es una forma de limpieza de disco. Por lo tanto, una acción de este tipo sólo puede darse en la agrupación en un momento determinado. Si hay una operación de limpieza en curso, una operación de creación de reflejo suspende la limpieza en curso y la reinicia una vez concluida la creación de reflejo.

Para obtener más información sobre la actualización de duplicación de datos, consulte "Visualización del estado de la actualización de duplicación de datos" en la página 321.

Solución de problemas con ZFS

En las secciones siguientes se explica la manera de identificar y resolver problemas en los sistemas de archivos o agrupaciones de almacenamiento de ZFS:

- "Cómo establecer si una agrupación de almacenamiento de ZFS tiene problemas" en la página 304
- "Revisión de la salida de `zpool status`" en la página 305
- "Creación de informes del sistema sobre mensajes de error de ZFS" en la página 308

Las funciones siguientes son válidas para identificar problemas en la configuración de ZFS:

- Se puede mostrar información detallada de agrupaciones de almacenamiento de ZFS utilizando el comando `zpool status`.
- Las notificaciones de errores en agrupaciones y dispositivos se realizan través de mensajes de diagnóstico de ZFS/FMA.
- Los comandos anteriores de ZFS que modificaban la información sobre el estado de las agrupaciones se ven ahora mediante el comando `zpool history`.

Casi todas las resoluciones de problemas de ZFS implican el uso del comando `zpool status`. Este comando analiza los errores de un sistema e identifica el problema más grave, sugiere una acción y proporciona un vínculo a documentación técnica para obtener más información. Aunque pueda haber varios problemas, el comando sólo identifica un problema de la agrupación. Por ejemplo, los errores de datos dañados generalmente denotan que ha fallado alguno de los dispositivos, pero la sustitución del dispositivo defectuoso podría no solucionar todos los problemas de deterioro de datos.

Además, un motor de diagnóstico de ZFS detecta y notifica errores de agrupaciones y dispositivos. También se notifican errores de suma de comprobación, E/S, dispositivos y agrupaciones asociados con errores de dispositivos o agrupaciones. Los errores de ZFS indicados por `fmd` se muestran en la consola y el archivo de mensajes del sistema. En la mayoría de los casos, el mensaje de `fmd` remite al comando `zpool status` para obtener más instrucciones sobre recuperación.

A continuación se expone el proceso básico de recuperación:

- Si procede, utilice el comando `zpool history` para identificar los comandos de ZFS anteriores que han desembocado en la situación de error. Por ejemplo:

  ```
  # zpool history tank
  History for 'tank':
  2010-07-15.12:06:50 zpool create tank mirror c0t1d0 c0t2d0 c0t3d0
  2010-07-15.12:06:58 zfs create tank/eric
  2010-07-15.12:07:01 zfs set checksum=off tank/eric
  ```

  Las sumas de comprobación de esta salida están desactivadas para el sistema de archivos `tank/eric`. No se recomienda esta configuración.
- Identifique los errores mediante los mensajes de `fmd` que aparecen en la consola del sistema o en el archivo `/var/adm/messages`.
- El comando `zpool status -x` proporciona más instrucciones de reparación.
- Repare los fallos, mediante las siguientes operaciones:
  - Reemplazar el dispositivo no disponible o faltante, y conectarlo.
  - Restauración de la configuración defectuosa o los datos dañados a partir de una copia de seguridad.
  - Verificación de la recuperación mediante el comando `zpool status - x`.
  - Copia de seguridad de la configuración que se ha restaurado, si procede.

En esta sección se explica la forma de interpretar la salida `zpool status` para diagnosticar el tipo de fallos que se pueden producir. Si bien el comando ejecuta automáticamente casi todo el proceso, es importante comprender con exactitud los problemas que se identifican para poder diagnosticar el tipo de error. Las siguientes secciones describen cómo solucionar los diversos problemas que pueden producirse.

Cómo establecer si una agrupación de almacenamiento de ZFS tiene problemas

La forma más fácil de determinar si un sistema tiene problemas conocidos es mediante el comando `zpool status -x`. Este comando sólo describe agrupaciones que presentan problemas. Si no hay agrupaciones cuyo estado es defectuoso, el comando muestra lo siguiente:

```
# zpool status -x
all pools are healthy
```

Sin el indicador `-x`, el comando muestra el estado completo de todas las agrupaciones (o de la agrupación solicitada, si se indica en la línea de comandos), incluso si las agrupaciones están en buen estado.

Para obtener más información sobre las opciones de línea de comandos en la salida de `zpool status`, consulte "Consulta del estado de una agrupación de almacenamiento de ZFS" en la página 93.

Revisión de la salida de `zpool status`

La salida completa de `zpool status` se parece a la siguiente:

```
# zpool status pond
  pool: pond
 state: DEGRADED
status: One or more devices are unavailable in response to persistent errors.
        Sufficient replicas exist for the pool to continue functioning in a
        degraded state.
action: Determine if the device needs to be replaced, and clear the errors
        using 'zpool clear' or 'fmadm repaired', or replace the device
        with 'zpool replace'.
        Run 'zpool status -v' to see device specific details.
  scan: scrub repaired 0 in 0h0m with 0 errors on Wed Jun 20 13:16:09 2012
config:

        NAME                      STATE     READ WRITE CKSUM
        pond                      DEGRADED     0     0     0
          mirror-0                ONLINE       0     0     0
            c0t5000C500335F95E3d0 ONLINE       0     0     0
            c0t5000C500335F907Fd0 ONLINE       0     0     0
          mirror-1                DEGRADED     0     0     0
            c0t5000C500335BD117d0 ONLINE       0     0     0
            c0t5000C500335DC60Fd0 UNAVAIL      0     0     0

errors: No known data errors
```

Esta salida se describe en la siguiente sección.

Información sobre el estado general de la agrupación

Esta sección de la salida de `zpool status` contiene los campos siguientes (algunos de ellos sólo se muestran cuando hay agrupaciones con problemas):

`pool` El nombre de la agrupación.

`state` Estado actual de la agrupación. Esta información se refiere únicamente a la capacidad de la agrupación de proporcionar el nivel pertinente de repetición.

`status` Describe cuál es el problema que afecta a la agrupación. Si no se detectan errores, este campo se omite.

`acción`	Acción recomendada para la reparación de errores. Si no se detectan errores, este campo se omite.
`see`	Referencia a información técnica que contiene datos detallados sobre reparaciones. Los artículos en línea se actualizan con más frecuencia que esta guía. Por lo tanto, debe consultarlos para informarse sobre los procedimientos de reparación más recientes. Si no se detectan errores, este campo se omite.
`scrub`	Identifica el estado actual de una operación de limpieza, que puede contener la fecha y hora de conclusión de la última operación de limpieza, una limpieza en curso o si no se ha solicitado ninguna operación de limpieza.
`errors`	Identifica errores conocidos de datos o la ausencia de esta clase de errores.

Información de configuración de la agrupación

El campo `config` de la salida de `zpool status` describe la configuración de los dispositivos que conforman la agrupación, además de su estado y los posibles errores generados por los dispositivos. El estado puede ser uno de los siguientes: `ONLINE`, `FAULTED`, `DEGRADED` o `SUSPENDED`. Si el estado es cualquiera de ellos menos `ONLINE`, significa que se pone el peligro la tolerancia a errores del grupo.

La segunda sección de la salida de configuración muestra estadísticas de errores. Dichos errores se dividen en tres categorías:

- `READ`: errores de E/S al emitir una solicitud de lectura
- `WRITE`: errores de E/S al emitir una solicitud de escritura
- `CKSUM`: errores de suma de comprobación, lo que significa que el dispositivo ha devuelto datos dañados como resultado de una solicitud de lectura

Estos errores son aptos para determinar si los daños son permanentes. Una cantidad pequeña de errores de E/S puede denotar un corte temporal del suministro; una cantidad grande puede denotar un problema permanente en el dispositivo. Estos errores no necesariamente corresponden a datos dañados según la interpretación de las aplicaciones. Si el dispositivo se encuentra en una configuración redundante, los dispositivos podrían mostrar errores irreparables, aunque no aparezcan errores en el reflejo o el nivel de dispositivos RAID-Z. En estos casos, ZFS ha recuperado correctamente los datos en buen estado e intentado reparar los datos dañados a partir de réplicas existentes.

Para obtener más información sobre la interpretación de estos errores, consulte "Cómo determinar el tipo de error en dispositivos" en la página 313.

En la última columna de la salida de `zpool status` se muestra información complementaria adicional. Dicha información se expande en el campo `state` para ayudar en el diagnóstico de modos de errores. Si un dispositivo tiene el estado `UNAVAIL`, este campo indica que no se puede

acceder al dispositivo o que los datos del dispositivo están dañados. Si se ejecuta la actualización de la duplicación de datos, el dispositivo muestra el progreso del proceso.

Para obtener información sobre el control del progreso de la actualización de duplicación de datos, consulte "Visualización del estado de la actualización de duplicación de datos" en la página 321.

Estado del proceso de limpieza

La sección de limpieza de la salida de `zpool status` describe el estado actual de cualquier operación de limpieza explícita. Esta información es diferente de si se detectan errores en el sistema, aunque es válida para determinar la exactitud de la información sobre datos dañados. Si la última operación de limpieza ha concluido correctamente, lo más probable es que se haya detectado cualquier tipo de datos dañados.

Se proporcionan los siguientes mensajes de estado de limpieza de `zpool status`:

- Informe de limpieza en curso. Por ejemplo:

```
scan: scrub in progress since Wed Jun 20 14:56:52 2012
    529M scanned out of 71.8G at 48.1M/s, 0h25m to go
    0 repaired, 0.72% done
```

- Mensaje de limpieza finalizada. Por ejemplo:

```
scan: scrub repaired 0 in 0h11m with 0 errors on Wed Jun 20 15:08:23 2012
```

- Mensaje de cancelación de limpieza en curso. Por ejemplo:

```
scan: scrub canceled on Wed Jun 20 16:04:40 2012
```

Los mensajes de limpieza completada se mantienen entre reinicios de sistema.

Para obtener más información sobre la limpieza de datos y la forma de interpretar esa información, consulte "Comprobación de integridad de sistema de archivos ZFS" en la página 301.

Errores de datos dañados

El comando `zpool status` muestra también si hay errores conocidos asociados con el grupo. Estos errores se pueden haber detectado durante la limpieza de datos o en el transcurso del funcionamiento normal. ZFS mantiene un registro constante de todos los errores de datos asociados con una agrupación. El registro se reinicia cada vez que concluye una limpieza total del sistema.

Los errores de datos dañados siempre son fatales. El hecho de que existan denota que al menos una aplicación ha tenido un error de E/S debido a los datos dañados de la agrupación. Los errores de dispositivos en una agrupación redundante no generan datos dañados ni forman

parte de este registro. De forma predeterminada, sólo se muestra el número de errores detectados. La opción `zpool status -v` proporciona una lista completa de errores con los detalles. Por ejemplo:

```
# zpool status -v tank
  pool: tank
 state: ONLINE
status: One or more devices has experienced an error resulting in data
        corruption.  Applications may be affected.
action: Restore the file in question if possible. Otherwise restore the
        entire pool from backup.
   see: http://support.oracle.com/msg/ZFS-8000-8A
  scan: scrub repaired 0 in 0h0m with 2 errors on Fri Jun 29 16:58:58 2012
config:

        NAME             STATE     READ WRITE CKSUM
        tank             ONLINE       2     0     0
          c8t0d0         ONLINE       0     0     0
          c8t1d0         ONLINE       2     0     0

errors: Permanent errors have been detected in the following files:

        /tank/file.1
```

El comando `fmd` muestra un mensaje parecido en la consola del sistema y el archivo `/var/adm/messages`. Con el comando `fmdump` se puede hacer un seguimiento de estos mensajes.

Para obtener más información sobre la interpretación de errores sobre corrupción de datos, consulte "Identificación del tipo de corrupción de datos" en la página 324.

Creación de informes del sistema sobre mensajes de error de ZFS

Aparte de hacer un constante seguimiento de los errores en la agrupación, ZFS muestra mensajes de `syslog` cuando se generan eventos de interés. Las siguientes situaciones generan eventos de notificación:

- **Transición de estados del dispositivo**: si un dispositivo pasa a tener el estado `FAULTED`, ZFS registra un mensaje que indica que la tolerancia a errores del grupo puede estar en peligro. Se envía un mensaje parecido si el dispositivo se conecta posteriormente, con lo cual la agrupación se recupera del error.
- **Datos dañados**: si se detecta cualquier tipo de datos dañados, ZFS registra un mensaje en el que se indica su ubicación y el momento en que tiene lugar. Este mensaje se registra sólo la primera vez que se detecta. Los accesos posteriores no generan ningún mensaje.

- **Errores de agrupaciones y de dispositivos**: si tiene lugar un error de agrupación o dispositivo, el daemon del administrador de errores informa de dichos errores mediante mensajes de `syslog` y mediante el comando `fmdump`.

Si ZFS detecta un error de dispositivo y se recupera automáticamente, no se genera ninguna notificación. Esta clase de errores no supone ningún fallo en la redundancia de la agrupación ni la integridad de los datos. Además, esta clase de errores suele ser fruto de un problema de controlador provisto de su propio conjunto de mensajes de error.

Reparación de una configuración de ZFS dañada

ZFS mantiene una caché de agrupaciones activas y su configuración en el sistema de archivos raíz. Si este archivo se daña o se desincroniza respecto a la información de configuración que se almacena en disco, no se podrá abrir la agrupación. ZFS procura evitar esta situación, si bien siempre se pueden producir daños arbitrarios debido a la naturaleza del almacenamiento subyacente. Al final termina desapareciendo una agrupación del sistema cuando lo normal es que estuviera disponible. Esta situación también puede presentarse como una configuración parcial en la que falta un número no determinado de dispositivos virtuales de nivel superior. Sea como sea, la configuración se puede recuperar exportando la agrupación (si está visible) y volviéndola a importar.

Para obtener información sobre importación y exportación de agrupaciones, consulte "Migración de agrupaciones de almacenamiento de ZFS" en la página 106.

Resolución de un dispositivo que no se encuentra

Si no se puede abrir un dispositivo, se muestra como `UNAVAIL` en la salida de `zpool status`. Este estado indica que ZFS no ha podido abrir el dispositivo la primera vez que se accedió a la agrupación, o que desde entonces el dispositivo ya no está disponible. Si el dispositivo hace que no quede disponible un dispositivo virtual de alto nivel, el grupo queda completamente inaccesible. De lo contrario, podría verse en peligro la tolerancia a errores del grupo. En cualquier caso, el dispositivo sólo tiene que volver a conectarse al sistema para restablecer el funcionamiento normal. Si necesita reemplazar un dispositivo con el estado `UNAVAIL` porque ha fallado, consulte "Sustitución de un dispositivo de un grupo de almacenamiento de ZFS" en la página 315.

Si un dispositivo está en estado UNAVAIL en una agrupación raíz o una agrupación raíz reflejada, consulte las referencias siguientes:

- **Error en disco de agrupación raíz reflejada**: "Inicio desde un disco alternativo en una agrupación raíz ZFS reflejada" en la página 138
- **Cómo reemplazar un disco en una agrupación raíz**

- "Cómo reemplazar un disco en una agrupación raíz ZFS (SPARC o x86/VTOC)" en la página 128
- "Cómo reemplazar un disco en una agrupación raíz ZFS (SPARC o x86/EFI [GPT])" en la página 130

- **Recuperación después de un desastre de la agrupación raíz completa**: Capítulo 11, "Archivado de instantáneas y recuperación de agrupaciones raíz"

Por ejemplo, en pantalla puede aparecer un mensaje parecido al siguiente procedente de `fmd` tras un error de dispositivo:

```
SUNW-MSG-ID: ZFS-8000-QJ, TYPE: Fault, VER: 1, SEVERITY: Minor
EVENT-TIME: Wed Jun 20 13:09:55 MDT 2012
PLATFORM: ORCL,SPARC-T3-4, CSN: 1120BDRCCD, HOSTNAME: tardis
SOURCE: zfs-diagnosis, REV: 1.0
EVENT-ID: e13312e0-be0a-439b-d7d3-cddaefe717b0
DESC: Outstanding dtls on ZFS device 'id1,sd@n5000c500335dc60f/a' in pool 'pond'.
AUTO-RESPONSE: No automated response will occur.
IMPACT: None at this time.
REC-ACTION: Use 'fmadm faulty' to provide a more detailed view of this event.
Run 'zpool status -lx' for more information. Please refer to the associated
reference document at http://support.oracle.com/msg/ZFS-8000-QJ for the latest
service procedures and policies regarding this diagnosis.
```

Para ver información más detallada sobre el problema del dispositivo y la resolución, utilice el comando `zpool status -v`. Por ejemplo:

```
# zpool status -v
  pool: pond
 state: DEGRADED
status: One or more devices are unavailable in response to persistent errors.
        Sufficient replicas exist for the pool to continue functioning in a
        degraded state.
action: Determine if the device needs to be replaced, and clear the errors
        using 'zpool clear' or 'fmadm repaired', or replace the device
        with 'zpool replace'.
  scan: scrub repaired 0 in 0h0m with 0 errors on Wed Jun 20 13:16:09 2012
config:

        NAME                      STATE     READ WRITE CKSUM
        pond                      DEGRADED     0     0     0
          mirror-0                ONLINE       0     0     0
            c0t5000C500335F95E3d0 ONLINE       0     0     0
            c0t5000C500335F907Fd0 ONLINE       0     0     0
          mirror-1                DEGRADED     0     0     0
            c0t5000C500335BD117d0 ONLINE       0     0     0
            c0t5000C500335DC60Fd0 UNAVAIL      0     0     0

device details:

        c0t5000C500335DC60Fd0    UNAVAIL          cannot open
        status: ZFS detected errors on this device.
                The device was missing.
           see: http://support.oracle.com/msg/ZFS-8000-LR for recovery
```

En esta salida, puede ver que el dispositivo `c0t5000C500335DC60Fd0` no funciona. Si determina que se trata de un dispositivo defectuoso, sustitúyalo.

Si es necesario, utilice el comando `zpool online` para conectar el dispositivo reemplazado. Por ejemplo:

Comunique a FMA que el dispositivo ha sido reemplazado si la salida de `fmadm faulty` identifica el error del dispositivo. Por ejemplo:

```
# fmadm faulty
--------------- ------------------------------------  -------------- ---------
TIME            EVENT-ID                              MSG-ID         SEVERITY
--------------- ------------------------------------  -------------- ---------
Jun 20 13:15:41 3745f745-371c-c2d3-d940-93acbb881bd8  ZFS-8000-LR    Major

Problem Status    : solved
Diag Engine       : zfs-diagnosis / 1.0
System
    Manufacturer  : unknown
    Name          : ORCL,SPARC-T3-4
    Part_Number   : unknown
    Serial_Number : 1120BDRCCD
    Host_ID       : 84a02d28

----------------------------------------
Suspect 1 of 1 :
   Fault class : fault.fs.zfs.open_failed
   Certainty   : 100%
   Affects     : zfs://pool=86124fa573cad84e/vdev=25d36cd46e0a7f49/pool_name=pond/vdev_
name=id1,sd@n5000c500335dc60f/a
   Status      : faulted and taken out of service

   FRU
     Name             : "zfs://pool=86124fa573cad84e/vdev=25d36cd46e0a7f49/pool_name=pond/vdev_
name=id1,sd@n5000c500335dc60f/a"
        Status        : faulty

Description : ZFS device 'id1,sd@n5000c500335dc60f/a' in pool 'pond' failed to
              open.

Response    : An attempt will be made to activate a hot spare if available.

Impact      : Fault tolerance of the pool may be compromised.

Action      : Use 'fmadm faulty' to provide a more detailed view of this event.
              Run 'zpool status -lx' for more information. Please refer to the
              associated reference document at
              http://support.oracle.com/msg/ZFS-8000-LR for the latest service
              procedures and policies regarding this diagnosis.
```

Extraiga la cadena en la sección `Affects:` de la salida `fmadm faulty` e inclúyala con el siguiente comando para informar a FMA que se reemplazó el dispositivo:

```
# fmadm repaired zfs://pool=86124fa573cad84e/vdev=25d36cd46e0a7f49/pool_name=pond/vdev_
name=id1,sd@n5000c500335dc60f/a
```

```
fmadm: recorded repair to of zfs://pool=86124fa573cad84e/vdev=25d36cd46e0a7f49/pool_name=pond/vdev_
name=id1,sd@n5000c500335dc60f/a
```

Como último paso, confirme que la agrupación con el dispositivo reemplazado está en buen estado. Por ejemplo:

```
# zpool status -x tank
pool 'tank' is healthy
```

Cómo volver a conectar físicamente un dispositivo

La forma de volver a conectar un dispositivo que falta depende del tipo de dispositivo. Si es una unidad de red, se debe restaurar la conectividad a la red. Si se trata de un dispositivo USB u otro medio extraíble, debe volverse a conectar al sistema. Si consiste en un disco local, podría haber fallado un controlador de tal forma que el dispositivo ya no estuviera visible en el sistema. En tal caso, el controlador se debe reemplazar en el punto en que los discos vuelvan a estar disponibles. Pueden darse otros problemas, según el tipo de hardware y su configuración. Si una unidad falla y ya no está visible en el sistema, el dispositivo debe tratarse como si estuviera dañado. Siga los procedimientos que se indican en "Sustitución o reparación de un dispositivo dañado" en la página 313.

Una agrupación puede tener el estado `SUSPENDED` si la conectividad del dispositivo se ve comprometida. Una agrupación `SUSPENDED` se mantiene en estado de espera hasta que se resuelve el problema del dispositivo. Por ejemplo:

```
# zpool status cybermen
  pool: cybermen
 state: SUSPENDED
status: One or more devices are unavailable in response to IO failures.
        The pool is suspended.
action: Make sure the affected devices are connected, then run 'zpool clear' or
        'fmadm repaired'.
        Run 'zpool status -v' to see device specific details.
   see: http://support.oracle.com/msg/ZFS-8000-HC
  scan: none requested
config:

        NAME           STATE     READ WRITE CKSUM
        cybermen       UNAVAIL      0    16     0
            c8t3d0     UNAVAIL      0     0     0
            c8t1d0     UNAVAIL      0     0     0
```

Después de restablecer la conectividad del dispositivo, borre la agrupación o los errores del dispositivo.

```
# zpool clear cybermen
# fmadm repaired zfs://pool=name/vdev=guid
```

Notificación de ZFS sobre disponibilidad de dispositivos

Después de que un dispositivo se vuelve a conectar al sistema, ZFS puede detectar o no automáticamente su disponibilidad. Si la agrupación tenía antes el estado `UNAVAIL` o `SUSPENDED`, o si el sistema se reinició como parte del procedimiento `attach`, ZFS vuelve a explorar automáticamente todos los dispositivos cuando intenta abrir la agrupación. Si la agrupación se había degradado y el dispositivo se reemplazó cuando el sistema estaba en ejecución, se debe notificar a ZFS que el dispositivo ya está disponible y listo para abrirse de nuevo mediante el comando `zpool online`. Por ejemplo:

```
# zpool online tank c0t1d0
```

Para obtener más información sobre la conexión de dispositivos, consulte "Cómo conectar un dispositivo" en la página 79.

Sustitución o reparación de un dispositivo dañado

Esta sección describe la forma de determinar tipos de errores en dispositivos, eliminar errores transitorios y reemplazar un dispositivo.

Cómo determinar el tipo de error en dispositivos

El concepto *dispositivo dañado* es bastante ambiguo; puede referirse a varias situaciones:

- **Deterioro de bits**: con el tiempo, eventos aleatorios como campos magnéticos o rayos cósmicos pueden causar anomalías en los bits almacenados en el disco. Son eventos relativamente poco frecuentes, pero lo suficientemente habituales como para causar daños en datos de sistemas grandes o con procesos de larga duración.
- **Lecturas o escrituras de ubicaciones incorrectas**: los errores de firmware o hardware pueden hacer que lecturas o escrituras de bloques enteros hagan referencia a ubicaciones incorrectas en el disco. Suelen ser errores transitorios, pero si se producen en grandes cantidades podrían denotar una unidad defectuosa.
- **Error de administrador**: los administradores pueden sobrescribir inadvertidamente porciones del disco con datos dañados (por ejemplo, sobrescribir porciones de `/dev/zero` en el disco) que afectarán el disco de manera permanente. Estos errores siempre son transitorios.

- **Interrupción temporal del suministro de energía**: durante un determinado periodo de tiempo, quizá no se pueda acceder a un disco, lo que puede provocar errores de E/S. Esta situación se suele asociar con dispositivos conectados a redes, aunque los discos locales también pueden sufrir interrupciones temporales de suministro de energía. Estos errores pueden ser transitorios o no.
- **Hardware dañado o inestable**: esta situación constituye un cajón de sastre de todos los problemas que puede presentar un hardware defectuoso, entre los que se pueden citar errores persistentes de E/S y transportes defectuosos que causan errores aleatorios. Estos errores suelen ser permanentes.
- **Dispositivo sin conexión**: si un dispositivo está sin conexión, se supone que el administrador le ha asignado este estado porque presenta algún problema. El administrador que asigna este estado al dispositivo puede establecer si dicha suposición es correcta.

El diagnóstico exacto de la naturaleza del problema puede resultar un proceso complicado. El primer paso es examinar la cantidad de errores en la salida de `zpool status`. Por ejemplo:

```
# zpool status -v tank
  pool: tank
 state: ONLINE
status: One or more devices has experienced an error resulting in data
        corruption.  Applications may be affected.
action: Restore the file in question if possible. Otherwise restore the
        entire pool from backup.
   see: http://support.oracle.com/msg/ZFS-8000-8A
config:

        NAME            STATE     READ WRITE CKSUM
        tank            ONLINE       2     0     0
          c8t0d0        ONLINE       0     0     0
          c8t0d0        ONLINE       2     0     0

errors: Permanent errors have been detected in the following files:

        /tank/file.1
```

Los errores pueden ser de E/S o de suma de comprobación, y pueden denotar el posible tipo de defecto. El funcionamiento normal prevé muy pocos errores (sólo unos pocos en periodos de tiempo prolongados). Si detecta una gran cantidad de errores, probablemente denote la inminencia de un error o la inutilización completa de un dispositivo. Pero un error de administrador también puede derivar en grandes cantidades de errores. El registro del sistema `syslog` es la otra fuente de información. Si el registro tiene una gran cantidad de mensajes de controlador de canal de fibra o SCSI, es probable que la situación sea sintomática de graves problemas de hardware. Si no se generan mensajes de `syslog`, es probable que los daños sean transitorios.

El objetivo es responder a la pregunta siguiente:

¿Es probable que este dispositivo vuelva a tener un error?

Los errores que suceden sólo una vez se consideran *transitorios* y no denotan problemas potenciales. Los errores continuos o suficientemente graves como para indicar problemas potenciales en el hardware se consideran errores *fatales*. El hecho de determinar el tipo de error trasciende el ámbito de cualquier software automatizado que haya actualmente en ZFS, por lo cual eso es una tarea propia de los administradores. Una vez determinado el error, se puede llevar a cabo la acción pertinente. Suprima los errores transitorios o reemplace los dispositivos con errores fatales. Estos procedimientos de reparación se explican en las secciones siguientes.

Aun en caso de que los errores de dispositivos se consideren transitorios, se pueden haber generado errores incorregibles en los datos de la agrupación. Estos errores precisan procedimientos especiales de reparación, incluso si el dispositivo subyacente se considera que está en buen estado o se ha reparado. Para obtener más información sobre cómo reparar errores de datos, consulte "Reparación de datos dañados" en la página 323.

Supresión de errores transitorios

Si los errores en dispositivos se consideran transitorios, en el sentido de que es poco probable que incidan más adelante en el buen estado del dispositivo, se pueden suprimir tranquilamente para indicar que no se ha producido ningún error fatal. Para suprimir los recuentos de errores de RAID-Z o dispositivos reflejados, utilice el comando `zpool clear`. Por ejemplo:

```
# zpool clear tank c1t1d0
```

Esta sintaxis suprime todos los errores de dispositivo y recuentos de errores de datos asociados con el dispositivo.

Utilice la sintaxis siguiente para suprimir todos los errores asociados con los dispositivos virtuales de una agrupación y para suprimir los recuentos de errores de datos asociados con la agrupación:

```
# zpool clear tank
```

Para obtener más información sobre la supresión de errores de dispositivos, consulte "Borrado de errores de dispositivo de agrupación de almacenamiento" en la página 80.

Sustitución de un dispositivo de un grupo de almacenamiento de ZFS

Si los daños en un dispositivo son permanentes o es posible que lo sean en el futuro, dicho dispositivo debe reemplazarse. El hecho de que el dispositivo pueda sustituirse o no depende de la configuración.

- "Cómo determinar si un dispositivo se puede reemplazar o no" en la página 316

Cómo determinar si un dispositivo se puede reemplazar o no

Si el dispositivo que se reemplazará forma parte de una configuración redundante, deben existir suficientes réplicas desde las que se puedan recuperar los datos en buen estado. Por ejemplo, si dos discos en un reflejo de cuatro vías tienen el estado `UNAVAIL`, se puede reemplazar cualquiera de ellos porque hay réplicas en buen estado. Sin embargo, si dos discos en un dispositivo virtual RAID-Z (`raidz1`) de cuatro vías tienen el estado `UNAVAIL`, ninguno de ellos se puede reemplazar porque no se dispone de suficientes réplicas desde las cuales recuperar los datos. Si el dispositivo está dañado pero tiene conexión, se puede reemplazar siempre que la agrupación no tenga el estado `UNAVAIL`. Sin embargo, cualquier dato dañado del dispositivo se copia al nuevo dispositivo a menos que haya suficientes réplicas con datos correctos.

En la configuración siguiente, el disco `c1t1d0` se puede reemplazar y los datos de la agrupación se copian de la réplica en buen estado, `c1t0d0`.

```
    mirror            DEGRADED
    c1t0d0             ONLINE
    c1t1d0             UNAVAIL
```

El disco `c1t0d0` también se puede reemplazar, aunque no es factible la recuperación automática de datos debido a la falta de réplicas en buen estado.

En la configuración siguiente, no se puede reemplazar ninguno de los discos `UNAVAIL`. Los discos `ONLINE` tampoco pueden reemplazarse porque la agrupación tiene el estado `UNAVAIL`.

```
    raidz1              UNAVAIL
    c1t0d0              ONLINE
    c2t0d0              UNAVAIL
    c3t0d0              UNAVAIL
    c4t0d0              ONLINE
```

En la configuración siguiente, el disco de nivel superior tampoco se puede reemplazar, si bien en el disco nuevo se va a copiar cualquier dato dañado.

```
c1t0d0         ONLINE
c1t1d0         ONLINE
```

Si cualquiera de los discos tiene el estado `UNAVAIL`, no se puede realizar ningún reemplazo porque la agrupación tiene el estado `UNAVAIL`.

Dispositivos que no se pueden reemplazar

Si la pérdida de un dispositivo hace que la agrupación tenga el estado `UNAVAIL`, o si el dispositivo contiene demasiados errores de datos en una configuración no redundante, el dispositivo no

puede reemplazarse de forma segura. Si la redundancia es insuficiente, no es posible restaurar con datos en buen estado el dispositivo dañado. En este caso, la única posibilidad es destruir la agrupación, volver a crear la configuración y, a continuación, restaurar los datos desde una copia de seguridad.

Para obtener más información sobre cómo restaurar todo un grupo, consulte "Reparación de daños en las agrupaciones de almacenamiento de ZFS" en la página 326.

Sustitución de un dispositivo de un grupo de almacenamiento de ZFS

Tras determinar que se puede reemplazar un dispositivo, utilice el comando `zpool replace` para reemplazarlo. Si va a reemplazar el dispositivo dañado con otro diferente, utilice sintaxis como ésta:

```
# zpool replace tank c1t1d0 c2t0d0
```

Este comando migra datos al dispositivo nuevo desde el dispositivo dañado, o de otros dispositivos de la agrupación si la configuración es redundante. Cuando finaliza el comando, desconecta el dispositivo dañado de la configuración. Es entonces cuando el dispositivo se puede eliminar del sistema. Si ya ha eliminado el dispositivo y lo ha reemplazado por uno nuevo en la misma ubicación, utilice la forma de un solo dispositivo del comando. Por ejemplo:

```
# zpool replace tank c1t1d0
```

Este comando selecciona un disco sin formato, le aplica el formato correspondiente y actualiza la duplicación de datos a partir del resto de la configuración.

Para obtener más información acerca del comando `zpool replace`, consulte "Sustitución de dispositivos en un grupo de almacenamiento" en la página 81.

EJEMPLO 10–1 Reemplazo de un disco SATA en una agrupación de almacenamiento ZFS

El siguiente ejemplo muestra cómo reemplazar un dispositivo (`c1t3d0`) en una agrupación de almacenamiento reflejada `tank` en un sistema con dispositivos SATA. Para reemplazar el disco `c1t3d0` con un nuevo disco en la misma ubicación (`c1t3d0`), desconfigure el disco antes de intentar reemplazarlo. Si el disco que se debe reemplazar no es un disco SATA, consulte "Sustitución de dispositivos en un grupo de almacenamiento" en la página 81.

Los pasos básicos son:

- Desconectar el disco (`c1t3d0`) que se va a sustituir. No puede anular la configuración de un disco SATA en uso.
- Utilizar el comando `cfgadm` para identificar el disco SATA (`c1t3d0`) cuya configuración se debe anular y llevar a cabo esta acción. La agrupación se degradará con el disco desconectado en esta configuración reflejada, pero la agrupación seguirá estando disponible.

EJEMPLO 10–1 Reemplazo de un disco SATA en una agrupación de almacenamiento ZFS *(Continuación)*

- Sustituir físicamente el disco (`c1t3d0`). Antes de eliminar físicamente la unidad `UNAVAIL`, asegúrese de que el LED azul `Ready to Remove` esté encendido.
- Volver a configurar el disco SATA (`c1t3d0`).
- Conectar el disco nuevo (`c1t3d0`).
- Ejecutar el comando `zpool replace` para reemplazar el disco (`c1t3d0`).

Nota – Si ha configurado previamente la propiedad de agrupación `autoreplace` como `on`, se dará formato y se sustituirá automáticamente cualquier dispositivo nuevo que se detecte en la misma ubicación física como dispositivo que antes pertenecía a la agrupación, mediante el comando `zpool replace`. Es posible que el hardware no sea compatible con esta función.

- Si un disco fallido se sustituye automáticamente por un repuesto en marcha, puede que deba desconectarlo después de dicha sustitución. Por ejemplo, si `c2t4d0` es aún un repuesto en marcha activo después de sustituir el disco fallido, desconéctelo.

  ```
  # zpool detach tank c2t4d0
  ```

- Si FMA informa un dispositivo fallido, debe reparar las fallas del dispositivo.

  ```
  # fmadm faulty
  # fmadm repaired zfs://pool=name/vdev=guid
  ```

El ejemplo siguiente detalla los pasos para reemplazar un disco en una agrupación de almacenamiento de ZFS.

```
# zpool offline tank c1t3d0
# cfgadm | grep c1t3d0
sata1/3::dsk/c1t3d0            disk         connected    configured   ok
# cfgadm -c unconfigure sata1/3
Unconfigure the device at: /devices/pci@0,0/pci1022,7458@2/pci11ab,11ab@1:3
This operation will suspend activity on the SATA device
Continue (yes/no)? yes
# cfgadm | grep sata1/3
sata1/3                        disk         connected    unconfigured ok
<Physically replace the failed disk c1t3d0>
# cfgadm -c configure sata1/3
# cfgadm | grep sata1/3
sata1/3::dsk/c1t3d0            disk         connected    configured   ok
# zpool online tank c1t3d0
# zpool replace tank c1t3d0
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: resilver completed after 0h0m with 0 errors on Tue Feb  2 13:17:32 2010
config:
```

EJEMPLO 10–1 Reemplazo de un disco SATA en una agrupación de almacenamiento ZFS *(Continuación)*

```
        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c0t1d0  ONLINE       0     0     0
            c1t1d0  ONLINE       0     0     0
          mirror-1  ONLINE       0     0     0
            c0t2d0  ONLINE       0     0     0
            c1t2d0  ONLINE       0     0     0
          mirror-2  ONLINE       0     0     0
            c0t3d0  ONLINE       0     0     0
            c1t3d0  ONLINE       0     0     0

errors: No known data errors
```

Tenga en cuenta que el comando `zpool output` anterior podría mostrar tanto los discos nuevos como los antiguos en un encabezado *replacing*. Por ejemplo:

```
replacing     DEGRADED     0     0    0
  c1t3d0s0/o  FAULTED      0     0    0
  c1t3d0      ONLINE       0     0    0
```

Este texto indica que el proceso de sustitución está en curso y se está actualizando la duplicación de datos.

Si va a reemplazar un disco (`c1t3d0`) con otro (`c4t3d0`), sólo tiene que ejecutar el comando `zpool replace`. Por ejemplo:

```
# zpool replace tank c1t3d0 c4t3d0
# zpool status
  pool: tank
 state: DEGRADED
 scrub: resilver completed after 0h0m with 0 errors on Tue Feb  2 13:35:41 2010
config:

        NAME             STATE     READ WRITE CKSUM
        tank             DEGRADED     0     0     0
          mirror-0       ONLINE       0     0     0
            c0t1d0       ONLINE       0     0     0
            c1t1d0       ONLINE       0     0     0
          mirror-1       ONLINE       0     0     0
            c0t2d0       ONLINE       0     0     0
            c1t2d0       ONLINE       0     0     0
          mirror-2       DEGRADED     0     0     0
            c0t3d0       ONLINE       0     0     0
            replacing    DEGRADED     0     0     0
              c1t3d0     OFFLINE      0     0     0
              c4t3d0     ONLINE       0     0     0

errors: No known data errors
```

EJEMPLO 10–1 Reemplazo de un disco SATA en una agrupación de almacenamiento ZFS *(Continuación)*

Es posible que deba ejecutar el comando `zpool status` varias veces hasta finalizar la sustitución del disco.

```
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: resilver completed after 0h0m with 0 errors on Tue Feb  2 13:35:41 2010
config:

        NAME          STATE     READ WRITE CKSUM
        tank          ONLINE       0     0     0
          mirror-0    ONLINE       0     0     0
            c0t1d0    ONLINE       0     0     0
            c1t1d0    ONLINE       0     0     0
          mirror-1    ONLINE       0     0     0
            c0t2d0    ONLINE       0     0     0
            c1t2d0    ONLINE       0     0     0
          mirror-2    ONLINE       0     0     0
            c0t3d0    ONLINE       0     0     0
            c4t3d0    ONLINE       0     0     0
```

EJEMPLO 10–2 Sustitución de un dispositivo de registro que presenta errores

ZFS identifica errores del registro de intento en la salida del comando `zpool status`. Diagnosis de arquitectura de administración fallida (FMA) informa de dichos errores también. Ambos, ZFS y FMA, describen cómo recuperarse de un error de intento de registro.

El ejemplo siguiente muestra cómo recuperar un dispositivo de registro (`c0t5d0`) que presenta errores en la agrupación de almacenamiento, (`pool`). Los pasos básicos son:

- Revisar el resultado de `zpool status -x` y el mensaje de diagnóstico de FMA que se describen a continuación:

  `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&doctype=REFERENCE&alias=EVENT:ZFS-8000-K4`
- Reemplazar físicamente el dispositivo de registro que presenta errores.
- Conectar el dispositivo de registro.
- Borrar la condición de error de la agrupación.
- Repare el error de la FMA.

Por ejemplo, si el sistema se cierra bruscamente antes de que las operaciones de escritura sincrónica se confirmen en una agrupación con un dispositivo de registro independiente, se muestran mensajes parecidos al siguiente:

EJEMPLO 10–2 Sustitución de un dispositivo de registro que presenta errores *(Continuación)*

```
# zpool status -x
  pool: pool
 state: FAULTED
status: One or more of the intent logs could not be read.
        Waiting for adminstrator intervention to fix the faulted pool.
action: Either restore the affected device(s) and run 'zpool online',
        or ignore the intent log records by running 'zpool clear'.
 scrub: none requested
config:

        NAME          STATE     READ WRITE CKSUM
        pool          FAULTED      0     0     0 bad intent log
          mirror-0    ONLINE       0     0     0
            c0t1d0    ONLINE       0     0     0
            c0t4d0    ONLINE       0     0     0
        logs          FAULTED      0     0     0 bad intent log
          c0t5d0      UNAVAIL      0     0     0 cannot open
<Physically replace the failed log device>
# zpool online pool c0t5d0
# zpool clear pool
# fmadm faulty
# fmadm repair zfs://pool=name/vdev=guid
```

Puede resolver el error del dispositivo de registro como se indica a continuación:

- Sustituya o recupere el dispositivo de registro. En este ejemplo, el dispositivo de registro es `c0t5d0`.
- Vuelva a conectar el dispositivo de registro.

  ```
  # zpool online pool c0t5d0
  ```

- Restablezca la condición de error del dispositivo de registro que presenta errores.

  ```
  # zpool clear pool
  ```

Si desea recuperarse de este error sin reemplazar el dispositivo de registro que presenta errores, puede borrar el error con el comando `zpool clear`. En esta situación, la agrupación no funcionará correctamente y los registros se escribirán en la agrupación principal hasta que se sustituya el dispositivo de registro independiente.

Considere el uso de dispositivos de registro reflejados para evitar los casos de error en el dispositivo de registro.

Visualización del estado de la actualización de duplicación de datos

El proceso de reemplazar un dispositivo puede tardar una considerable cantidad de tiempo, según el tamaño del dispositivo y la cantidad de datos que haya en la agrupación. El proceso de transferir datos de un dispositivo a otro, denominado *actualización de la duplicación de datos*, se puede controlar mediante el comando `zpool status`.

Se proporcionan los siguientes mensajes de estado de reconstrucción de `zpool status`:

- Informe de reconstrucción en curso. Por ejemplo:

```
scan: resilver in progress since Mon Jun  7 09:17:27 2010
   13.3G scanned out of 16.2G at 18.5M/s, 0h2m to go
   13.3G resilvered, 82.34% done
```

- Mensaje de reconstrucción finalizada. Por ejemplo:

```
resilvered 16.2G in 0h16m with 0 errors on Mon Jun  7 09:34:21 2010
```

Los mensajes de reconstrucción completada se mantienen entre reinicios del sistema.

Los sistemas de archivos tradicionales actualizan duplicaciones de datos en los bloques. Debido a que ZFS suprime la disposición artificial de capas de Volume Manager, puede ejecutar la actualización de duplicación de datos de manera más potente y controlada. Esta función presenta dos ventajas principales:

- ZFS sólo actualiza la duplicación de los datos necesarios. En caso de una breve interrupción del suministro (en contraposición a un reemplazo completo del dispositivo), la actualización de duplicación de datos del disco puede hacerse en cuestión de segundos. Si se reemplaza todo un disco, el tiempo que implica el proceso de actualización de duplicación de datos es proporcional a la cantidad de datos que se utilizan en disco. La sustitución de un disco de 500 GB puede ser cuestión de segundos si la agrupación sólo tiene unos cuantos gigabytes de espacio usado en el disco.
- La actualización de duplicación de datos es un proceso seguro que se puede interrumpir. Si el sistema se queda sin conexión o se reinicia, el proceso de actualización de duplicación de datos reanuda la tarea exactamente en el punto en que se había interrumpido, sin que haga falta hacer nada.

Para observar el progreso de la actualización de duplicación de datos, utilice el comando `zpool status`. Por ejemplo:

```
# zpool status tank
  pool: tank
 state: ONLINE
status: One or more devices is currently being resilvered.  The pool will
        continue to function, possibly in a degraded state.
action: Wait for the resilver to complete.
 scan: resilver in progress since Mon Jun  7 10:49:20 2010
    54.6M scanned out of 222M at 5.46M/s, 0h0m to go
    54.5M resilvered, 24.64% done
config:

        NAME             STATE     READ WRITE CKSUM
        tank             ONLINE       0     0     0
          mirror-0       ONLINE       0     0     0
            replacing-0  ONLINE       0     0     0
              c1t0d0     ONLINE       0     0     0
              c2t0d0     ONLINE       0     0     0  (resilvering)
            c1t1d0       ONLINE       0     0     0
```

En este ejemplo, el disco `c1t0d0` se sustituye por `c2t0d0`. Este evento se refleja en la salida del estado mediante la presencia del dispositivo virtual que `reemplaza` en la configuración. Este dispositivo no es real ni sirve para crear una agrupación. La única finalidad de este dispositivo es mostrar el proceso de actualización de duplicación de datos e identificar el dispositivo que se va a reemplazar.

Cualquier agrupación sometida al proceso de actualización de duplicación de datos adquiere el estado `ONLINE` o `DEGRADED`, porque hasta que no haya finalizado dicho proceso es incapaz de proporcionar el nivel necesario de redundancia. La actualización de duplicación de datos se ejecuta lo más deprisa posible, si bien la E/S siempre se programa con una prioridad inferior a la E/S solicitada por el usuario, para que repercuta en el sistema lo menos posible. Tras finalizarse la actualización de duplicación de datos, la configuración asume los parámetros nuevos. Por ejemplo:

```
# zpool status tank
  pool: tank
 state: ONLINE
 scrub: resilver completed after 0h1m with 0 errors on Tue Feb  2 13:54:30 2010
config:

        NAME        STATE     READ WRITE CKSUM
        tank        ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c2t0d0  ONLINE       0     0     0  377M resilvered
            c1t1d0  ONLINE       0     0     0

errors: No known data errors
```

La agrupación pasa de nuevo al estado `ONLINE` y el disco dañado original (`c1t0d0`) desaparece de la configuración.

Reparación de datos dañados

En las secciones siguientes se explica el procedimiento para identificar el tipo de corrupción de datos y, si es factible, cómo reparar los datos.

- "Identificación del tipo de corrupción de datos" en la página 324
- "Reparación de un archivo o directorio dañado" en la página 325
- "Reparación de daños en las agrupaciones de almacenamiento de ZFS" en la página 326

Para reducir al mínimo las posibilidades de que los datos sufran daños, ZFS utiliza sumas de comprobación, redundancia y datos que se reparan a sí mismos. Ahora bien, los datos se pueden dañar si una agrupación no es redundante, cuando una agrupación está en estado "degraded" o si se combina una improbable serie de eventos para dañar varias copias de determinados datos. Sea cual sea el origen, el resultado es el mismo: los datos quedan dañados y no se puede acceder a ellos. Las medidas requeridas dependen del tipo de datos dañados y su valor relativo. Se pueden dañar dos tipos básicos de datos:

- Metadatos de grupo: para abrir un grupo y acceder a conjuntos de datos, ZFS debe analizar cierta cantidad de datos. Si se dañan estos datos, quedará inaccesible toda la agrupación o partes de la jerarquía del conjuntos de datos.
- Datos de objeto: en este caso, se daña un determinado archivo o directorio. Ello puede hacer que no se pueda acceder a una parte del archivo o directorio, o causar la interrupción del objeto.

Los datos se verifican durante el funcionamiento normal y durante el proceso de limpieza. Para obtener más información sobre cómo verificar la integridad de datos de agrupaciones, consulte "Comprobación de integridad de sistema de archivos ZFS" en la página 301.

Identificación del tipo de corrupción de datos

De forma predeterminada, el comando `zpool status` avisa únicamente de la presencia de daños, pero no indica su ubicación. Por ejemplo:

```
# zpool status tank
  pool: tank
 state: ONLINE
status: One or more devices has experienced an error resulting in data
        corruption.  Applications may be affected.
action: Restore the file in question if possible. Otherwise restore the
        entire pool from backup.
   see: http://support.oracle.com/msg/ZFS-8000-8A
config:

        NAME                      STATE     READ WRITE CKSUM
        tank                      ONLINE       4     0     0
          c0t5000C500335E106Bd0   ONLINE       0     0     0
          c0t5000C500335FC3E7d0   ONLINE       4     0     0


errors: 2 data errors, use '-v' for a list
```

Cada error indica solamente que ha habido un error en un determinado momento. Eso no significa que cada error siga estando en el sistema. Éste es el caso en circunstancias normales. Determinadas interrupciones temporales del suministro pueden provocar daños en los datos que se reparan automáticamente cuando finaliza dicha interrupción. Se garantiza la ejecución completa de un proceso de limpieza del grupo para examinar cada bloque activo del grupo, con lo cual el registro de errores se reinicia cuando concluye la limpieza. Si considera que ya no hay errores y no quiere esperar a que finalice la limpieza, reinicie todos los errores de la agrupación mediante el comando `zpool online`.

Si los dañados afectan a metadatos de toda la agrupación, la salida difiere ligeramente. Por ejemplo:

```
# zpool status -v morpheus
  pool: morpheus
    id: 13289416187275223932
 state: UNAVAIL
status: The pool metadata is corrupted.
action: The pool cannot be imported due to damaged devices or data.
   see: http://support.oracle.com/msg/ZFS-8000-72
config:

        morpheus    FAULTED   corrupted data
          c1t10d0   ONLINE
```

Si los daños afectan a toda la agrupación, ésta pasa al estado `FAULTED` , ya que posiblemente no podrá proporcionar el nivel de redundancia requerido.

Reparación de un archivo o directorio dañado

Si un archivo o directorio resultasen dañados, según el tipo de corrupción, el sistema podría seguir funcionando. Si en el sistema no hay copias de los datos de buena calidad, cualquier daño que tenga lugar será irreparable. Si los datos son importantes, la única alternativa es recuperarlos a partir de una copia de seguridad. Aun así, esta situación quizá se pueda solventar sin tener que restaurar todo el grupo.

Si se ha dañado un bloque de datos de archivo, el archivo se puede eliminar sin problemas; de este modo, el error desaparece del sistema. Utilice el comando `zpool status -v` para ver en pantalla una lista con nombres de archivos que tienen errores constantes. Por ejemplo:

```
# zpool status tank -v
  pool: tank
 state: ONLINE
status: One or more devices has experienced an error resulting in data
        corruption.  Applications may be affected.
action: Restore the file in question if possible. Otherwise restore the
        entire pool from backup.
   see: http://support.oracle.com/msg/ZFS-8000-8A
config:

        NAME                        STATE     READ WRITE CKSUM
        tank                        ONLINE       4     0     0
          c0t5000C500335E106Bd0     ONLINE       0     0     0
          c0t5000C500335FC3E7d0     ONLINE       4     0     0

errors: Permanent errors have been detected in the following files:
        /tank/file.1
        /tank/file.2
```

La lista de nombres de archivos con errores constantes se puede describir del modo siguiente:

- Si se busca la ruta de acceso del archivo y se monta el conjunto de datos, se muestra en pantalla toda la ruta del archivo. Por ejemplo:

```
/monkey/a.txt
```

- Si se busca la ruta de acceso del archivo pero el conjunto de datos no se monta, en pantalla se muestra el nombre del conjunto de datos sin una barra inclinada (/), seguido de la ruta de acceso del conjunto de datos al archivo. Por ejemplo:

  ```
  monkey/ghost/e.txt
  ```

- Si no se puede trasladar correctamente el número de objeto a una ruta de archivo, ya sea por un error o porque el objeto no tiene asociada ninguna ruta de archivo auténtica, como en el caso de `dnode_t`, en pantalla se muestra nombre del conjunto de datos seguido del número de objeto. Por ejemplo:

  ```
  monkey/dnode:<0x0>
  ```

- Si se daña un objeto del conjunto de metaobjetos, en pantalla se muestra un etiqueta especial de `<metadata>`, seguida del número de objeto.

Si los daños se dan en un directorio o en los metadatos de un archivo, la única alternativa es colocar el archivo en otra ubicación. Puede colocar cualquier archivo o directorio en una ubicación menos apropiada para poder restaurar el objeto original.

Reparación de datos dañados con referencias de varios bloques

Si un sistema de archivos dañado tiene datos dañados con las referencias de varios bloques (por ejemplo, de instantáneas), el comando `zpool status -v` no mostrará **todas** las rutas de datos dañados. El algoritmo de limpieza de ZFS recorre la agrupación y pasa por cada bloque de datos una sola vez. Puede informar los daños solamente la **primera** vez que se encuentran. Por lo tanto, sólo genera una única ruta de acceso al archivo afectado. Tenga en cuenta que esto también se aplica a bloques dañados en los que se han eliminado datos duplicados.

Si tiene datos dañados, y el comando `zpool status -v` identifica que esa instantánea de datos se ve afectada, considere la posibilidad de buscar otras rutas dañadas.

```
# find mount-point -inum $inode -print
# find mount-point/.zfs/snapshot -inum $inode -print
```

El primer comando busca el número de inode de los datos dañados informados en el sistema de archivos especificado y todas las instantáneas. El segundo comando busca instantáneas con el mismo número de inode.

Reparación de daños en las agrupaciones de almacenamiento de ZFS

Si los metadatos de una agrupación resultan dañados de tal manera que es imposible abrir la agrupación o importarla, puede realizar alguna de las siguientes acciones:

- Intentar recuperar la agrupación mediante el comando `zpool clear - F` o el comando `zpool import -F`. Estos comandos intentan restaurar un estado operativo de las transacciones de agrupación más recientes. Puede utilizar el comando `zpool status` para revisar una agrupación dañada y el procedimiento de recuperación recomendado. Por ejemplo:

```
# zpool status
  pool: tpool
 state: UNAVAIL
status: The pool metadata is corrupted and the pool cannot be opened.
action: Recovery is possible, but will result in some data loss.
        Returning the pool to its state as of Fri Jun 29 17:22:49 2012
        should correct the problem.  Approximately 5 seconds of data
        must be discarded, irreversibly.  Recovery can be attempted
        by executing 'zpool clear -F tpool'.  A scrub of the pool
        is strongly recommended after recovery.
   see: http://support.oracle.com/msg/ZFS-8000-72
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
         tpool      UNAVAIL      0     0     1  corrupted data
          c1t1d0    ONLINE       0     0     2
          c1t3d0    ONLINE       0     0     4
```

El proceso de recuperación como se describe en la salida anterior consiste en utilizar el siguiente comando:

```
# zpool clear -F tpool
```

Si intenta importar una agrupación de almacenamiento dañada, se muestran mensajes parecidos al siguiente:

```
# zpool import tpool
cannot import 'tpool': I/O error
        Recovery is possible, but will result in some data loss.
        Returning the pool to its state as of Fri Jun 29 17:22:49 2012
        should correct the problem.  Approximately 5 seconds of data
        must be discarded, irreversibly.  Recovery can be attempted
        by executing 'zpool import -F tpool'.  A scrub of the pool
        is strongly recommended after recovery.
```

El proceso de recuperación como se describe en la salida anterior consiste en utilizar el siguiente comando:

```
# zpool import -F tpool
Pool tpool returned to its state as of Fri Jun 29 17:22:49 2012.
Discarded approximately 5 seconds of transactions
```

Si la agrupación dañada está en el archivo `zpool.cache`, el problema se descubre al iniciar el sistema, y dicha agrupación se notifica en el comando `zpool status`. Si la agrupación no está en el archivo `zpool.cache`, no se importará ni se abrirá correctamente, y cuando intente importarla aparecerán mensajes que indicarán que está dañada.

- Puede importar una agrupación dañada en el modo de sólo lectura. Este método le permite importar la agrupación para que pueda acceder a los datos. Por ejemplo:

  ```
  # zpool import -o readonly=on tpool
  ```

  Para obtener más información sobre la importación de una agrupación con permiso de sólo lectura, consulte "Importación de una agrupación en modo de sólo lectura" en la página 113.
- Puede importar una agrupación a la que le falta un dispositivo de registro mediante el comando `zpool import -m`. Para obtener más información, consulte "Importación de una agrupación a la que le falta un dispositivo de registro" en la página 111.
- Si la agrupación no se puede recuperar con el método de recuperación de agrupación descrito anteriormente, deberá restaurar la agrupación y todos sus datos desde una copia de seguridad. Los procedimientos para ello son muy variados: dependen de la configuración de las agrupaciones y de la estrategia de las copias de seguridad. En primer lugar, guarde la configuración tal como se muestra en el comando `zpool status` para poder volver a crearla después de la destrucción de la agrupación. A continuación, utilice el comando `zpool destroy -f` para destruir la agrupación.

  Asimismo, conserve un archivo que contenga la disposición de los conjuntos de datos y guarde en lugar seguro las distintas propiedades que se han definido, ya que si en algún momento no se puede acceder al grupo, tampoco se podrá acceder a esta información. A partir de la configuración del grupo y la disposición del conjunto de datos, es posible reconstruir toda la configuración tras la destrucción del grupo. Los datos se pueden rellenar utilizando cualquier método de restauración o copia de seguridad.

Reparación de un sistema que no se puede iniciar

ZFS se ha concebido para mantenerse robusto y estable frente a los errores. Aun así, los errores de software o problemas imprevistos pueden desequilibrar el sistema al intentar acceder a una agrupación. Como parte del proceso de inicio se debe abrir cada agrupación, lo que significa que esta clase de errores harán que el sistema entre en un bucle de inicios de emergencia. Para solucionar esta situación, debe indicar a ZFS que no busque agrupaciones al inicio.

ZFS mantiene una caché interna de grupos disponibles junto con sus configuraciones en `/etc/zfs/zpool.cache`. La ubicación y el contenido de este archivo son personales y susceptibles de cambiarse. Si el sistema no se puede iniciar, inicie en `none` mediante la opción `-m =none`. Cuando el sistema se haya iniciado, vuelva a montar el sistema de archivos raíz como grabable y cambie el nombre o cambie la ubicación del archivo `/etc/zfs/zpool.cache`. Estas acciones hacen que ZFS no tenga en cuenta que en el sistema hay agrupaciones, lo cual impide que intente acceder a la agrupación dañada que causa el problema. A continuación, puede pasar a un estado normal del sistema mediante el comando `svcadm milestone all`. Se puede aplicar un proceso similar al iniciar desde un sistema de archivos raíz alternativo para efectuar reparaciones.

Cuando el sistema se haya iniciado, puede intentar importar la agrupación mediante el comando `zpool import`. Sin embargo, es probable que se cause el mismo error del inicio, puesto que el comando emplea el mismo mecanismo de acceso a grupos. Si en el sistema hay varias agrupaciones, haga lo siguiente:

- Cambie el nombre de `zpool.cache` o lleve el archivo a otra ubicación, tal como se ha indicado anteriormente.
- Determine qué agrupación podría tener problemas utilizando el comando `fmdump -eV`, para ver las agrupaciones que han notificado errores fatales.
- Importe una por una las agrupaciones que tienen problemas, como se describe en la salida de `fmdump`.

◆ ◆ ◆ CAPÍTULO 11

Archivado de instantáneas y recuperación de agrupaciones raíz

En este capítulo, se describe cómo archivar instantáneas que se pueden utilizar para migrar o restaurar un sistema Oracle Solaris 11 en caso de un fallo del sistema. Puede utilizar estos pasos para crear el núcleo de un plan básico de recuperación después de un desastre, o para migrar la configuración de un sistema a un nuevo dispositivo de inicio.

Este capítulo se divide en las secciones siguientes:

Descripción general del proceso de recuperación de ZFS

Como mínimo, se debe realizar una copia de seguridad de todos los datos del sistema de archivos de forma regular para reducir el tiempo de inactividad debido a fallos del sistema. Si se produce un fallo catastrófico en el sistema, puede restaurar las instantáneas de la agrupación raíz ZFS, en lugar de volver a instalar el sistema operativo y volver a crear la configuración del sistema. Luego, puede restaurar datos de agrupaciones que no sean raíz.

Cualquier sistema que ejecute Oracle Solaris 11 es candidato para copia de seguridad y archivado. El proceso general implica los siguientes pasos:

- Crear un archivo de instantánea de ZFS para los sistemas de archivos de agrupación raíz y para cualquier agrupación que no sea raíz que sea necesario migrar o recuperar.

  Debe volver a archivar las instantáneas de agrupación raíz después que se actualiza el sistema operativo.
- Guardar el archivo de instantánea en un soporte extraíble local, como una unidad USB, o enviar las instantáneas a un sistema remoto para una posible recuperación.
- Se sustituyen los discos fallidos u otros componentes del sistema.

- El sistema de destino se inicia desde el soporte de instalación de Oracle Solaris 11, se crean nuevas agrupaciones de almacenamiento y se recuperan los sistemas de archivos.
- Realizar la mínima configuración de inicio. Luego, el sistema se puede utilizar y ofrece todos los servicios que se ejecutaron en el momento de archivado.

Requisitos de recuperación de agrupaciones ZFS

- Los sistemas de archivado y de recuperación deben tener la misma arquitectura y deben cumplir los requisitos mínimos de Oracle Solaris 11 para las plataformas admitidas.
- Los discos de sustitución que contendrán la nueva agrupación de almacenamiento ZFS deben tener como mínimo la misma capacidad que los datos utilizados en las agrupaciones de archivado (ver a continuación).

  En Oracle Solaris 11, un disco de agrupación raíz debe tener una etiqueta SMI (VTOC). En Oracle Solaris 11.1, los discos de agrupación raíz de un sistema basado en x86 pueden tener una etiqueta SMI (VTOC) o EFI (GPT). Para obtener información acerca de la compatibilidad de inicio para discos con etiqueta EFI (GPT), consulte "Compatibilidad de inicio para discos con etiqueta EFI (GPT)" en la página 17.
- El acceso root es necesario en el sistema que contiene las instantáneas de archivado y en el sistema de recuperación. Si está utilizando `ssh` para acceder al sistema remoto, deberá configurarlo para el acceso privilegiado.

Creación de un archivo de instantánea ZFS para la recuperación

Antes de crear la instantánea de agrupación raíz ZFS, considere guardar la siguiente información:

- Capture las propiedades de la agrupación raíz.

```
sysA# zpool get all rpool
```

- Identifique el tamaño y la capacidad actual del disco de la agrupación raíz.

```
sysA# zpool list
NAME   SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
rpool   74G  5.42G  68.6G   7%  1.00x  ONLINE  -
```

- Identifique los componentes de la agrupación raíz.

```
sysA# zfs list -r rpool
NAME                      USED  AVAIL  REFER  MOUNTPOINT
rpool                    13.8G  53.1G  73.5K  /rpool
rpool/ROOT               3.54G  53.1G    31K  legacy
rpool/ROOT/solaris       3.54G  53.1G  3.37G  /
rpool/ROOT/solaris/var    165M  53.1G   163M  /var
rpool/VARSHARE           37.5K  53.1G  37.5K  /var/share
```

```
rpool/dump               8.19G  53.4G  7.94G  -
rpool/export               63K  53.1G    32K  /export
rpool/export/home          31K  53.1G    31K  /export/home
rpool/swap               2.06G  53.2G  2.00G  -
```

▼ Cómo crear un archivo de instantánea ZFS

En los siguientes pasos, se describe cómo crear una instantánea recursiva de la agrupación raíz que incluya todos los sistemas de archivos de la agrupación raíz. Otras agrupaciones que no son raíz se pueden archivar de este mismo modo.

Considere los siguientes puntos:

- Para una recuperación completa del sistema, envíe las instantáneas a una agrupación en un sistema remoto.
- Cree un recurso compartido NFS desde el sistema remoto y también configure `ssh` para permitir el acceso privilegiado, si es necesario.
- La instantánea de agrupación raíz recursiva se envía como un archivo de instantánea de gran tamaño a un sistema remoto, pero usted puede enviar las instantáneas recursivas para que se almacenen como instantáneas individuales en un sistema remoto.

En los siguientes pasos, la instantánea recursiva se denomina `rpool@snap1`. El sistema local que se debe recuperar es `sysA` y el sistema remoto es `sysB`. Tenga en cuenta que `rpool` es el nombre predeterminado de la agrupación raíz y puede ser diferente en su sistema.

1 Conviértase en un administrador.

2 Cree una instantánea recursiva de la agrupación raíz.

```
sysA# zfs snapshot -r rpool@rpool.snap1
```

3 Reduzca el archivo de instantánea mediante la eliminación de las instantáneas de intercambio y volcado, si lo desea.

```
sysA# zfs destroy rpool/dump@rpool.snap1
sysA# zfs destroy rpool/swap@rpool.snap1
```

El volumen de intercambio no contiene datos relevantes para una migración o recuperación del sistema. No elimine la instantánea de volumen de volcado si desea conservar los volcados por caída.

4 **Envíe la instantánea de agrupación raíz recursiva a otra agrupación en otro sistema.**

a. **Comparta un sistema de archivos en un sistema remoto para recibir la instantánea o las instantáneas:**

En los siguientes pasos, el sistema de archivos `/tank/snaps` se comparte para almacenar la instantánea raíz recursiva.

```
sysB# zfs set share.nfs=on tank/snaps
sysB# zfs set share.nfs.sec.default.root=sysA tank/snaps
```

b. **Envíe la instantánea de agrupación raíz recursiva a un sistema remoto.**

Envíe la instantánea recursiva al sistema de archivos remoto que se compartió en el paso anterior.

```
sysA# zfs send -Rv rpool@rpool.snap1 | gzip > /net/sysB/tank/snaps/
rpool.snap1.gz
sending from @ to rpool@rpool.snap1
sending from @ to rpool/VARSHARE@rpool.snap1
sending from @ to rpool/export@rpool.snap1
sending from @ to rpool/export/home@rpool.snap1
sending from @ to rpool/ROOT@rpool.snap1
sending from @ to rpool/ROOT/solaris@install
sending from @install to rpool/ROOT/solaris@rpool.snap1
sending from @ to rpool/ROOT/solaris/var@install
sending from @install to rpool/ROOT/solaris/var@rpool.snap1
```

Volver a crear la agrupación raíz y recuperar instantáneas de la agrupación raíz

Si necesita volver a crear la agrupación raíz y recuperar las instantáneas de la agrupación raíz, los pasos generales son los siguientes:

- Prepare uno o varios discos de reemplazo de la agrupación raíz y vuelva a crear la agrupación raíz.
- Restaure las instantáneas del sistema de archivos de la agrupación raíz.
- Seleccione y active el entorno de inicio deseado.
- Inicie el sistema.

▼ Cómo volver a crear la agrupación raíz en el sistema de recuperación

Revise las siguientes consideraciones al recuperar la agrupación raíz.

- Si un disco de la agrupación raíz no redundante falla, necesitará iniciar el sistema desde el soporte de instalación o un servidor de instalación para volver a instalar el sistema operativo o restaurar las instantáneas de la agrupación raíz que ha archivado anteriormente.

Para obtener información sobre cómo reemplazar un disco en el sistema, consulte la documentación del hardware.

- Si un disco de la agrupación raíz reflejada falla, puede sustituir el disco fallado mientras el sistema sigue funcionando. Para obtener información sobre el reemplazo de un disco fallido en una agrupación raíz reflejada, consulte "Cómo reemplazar un disco en una agrupación raíz ZFS (SPARC o x86/VTOC)" en la página 128.

1 Identifique y reemplace el disco de la agrupación raíz o el componente del sistema fallados.

En general, este disco es el dispositivo de inicio predeterminado, o puede seleccionar otro disco y restablecer el dispositivo de inicio predeterminado.

2 Inicie el sistema desde el soporte de instalación de Oracle Solaris 11 mediante la selección de una de las siguientes opciones.

- Soporte de instalación DVD o USB (SPARC o x86): inserte el soporte y seleccione el dispositivo correspondiente como dispositivo de inicio.

  Si se utiliza un soporte basado en texto, seleccione la opción Shell del menú del instalador de texto.
- Live Media (sólo x86): durante el procedimiento de recuperación, se puede utilizar la sesión de escritorio GNOME.
- Instalador automatizado o una copia local del soporte AI (SPARC o x86): seleccione la opción Shell del menú del instalador de texto. En un sistema SPARC, inicie el soporte AI (ya sea de forma local o a través de la red) y seleccione la opción Shell:

```
ok boot net:dhcp
.
.
.Welcome to the Oracle Solaris 11 installation menu

        1  Install Oracle Solaris
        2  Install Additional Drivers
        3  Shell
        4  Terminal type (currently xterm)
        5  Reboot

Please enter a number [1]: 3
```

3 SPARC o x86 (VTOC): prepare el disco de agrupación raíz.

a. Confirme que el disco de reemplazo de la agrupación raíz se puede ver desde la utilidad `format`.

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
       0. c2t0d0 <FUJITSU-MAY2073RCSUN72G-0401 cyl 14087 alt 2 hd 24 sec 424>
          /pci@780/pci@0/pci@9/scsi@0/sd@0,0
       1. c2t1d0 <FUJITSU-MAY2073RCSUN72G-0401 cyl 14087 alt 2 hd 24 sec 424>
          /pci@780/pci@0/pci@9/scsi@0/sd@1,0
```

```
       2. c2t2d0 <SEAGATE-ST973402SSUN72G-0400-68.37GB>
          /pci@780/pci@0/pci@9/scsi@0/sd@2,0
       3. c2t3d0 <SEAGATE-ST973401LSUN72G-0556-68.37GB>
          /pci@780/pci@0/pci@9/scsi@0/sd@3,0
Specify disk (enter its number): 0
```

b. SPARC o x86 (VTOC): confirme que el disco de agrupación raíz tiene una etiqueta SMI (VTOC) y un segmento 0 con la mayor parte de espacio en disco.

Revise la tabla de particiones para confirmar que el disco de la agrupación raíz tiene una etiqueta SMI y un segmento 0.

```
selecting c2t0d0
[disk formatted]
format> partition
partition> print
```

c. SPARC o x86 (VTOC): vuelva a etiquetar el disco con una etiqueta SMI (VTOC), si es necesario.

Utilice los siguientes comandos de método abreviado para volver a etiquetar el disco. Tenga en cuenta que estos comandos no proporcionan ninguna comprobación de errores; por lo tanto, asegúrese de estar volviendo a etiquetar el disco correcto.

- SPARC:

  ```
  sysA# format -L vtoc -d c2t0d0
  ```

  Confirme que el segmento 0 tenga espacio en disco asignado de modo adecuado. La partición predeterminada se aplica en el comando anterior, que puede ser demasiado pequeño para el segmento 0 de la agrupación raíz. Para obtener información acerca de la modificación de la tabla de particiones predeterminada, consulte "cómo reemplazar un disco de agrupación raíz ZFS (EFI [GPT])" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.

- x86:

  ```
  sysA# fdisk -B /dev/rdsk/c2t0d0p0
  sysA# format -L vtoc -d c2t0d0
  ```

  Confirme que el segmento 0 tenga espacio en disco asignado de modo adecuado. La partición predeterminada se aplica en el comando anterior, que puede ser demasiado pequeño para el segmento 0 de la agrupación raíz. Para obtener información acerca de la modificación de la tabla de particiones predeterminada, consulte "cómo reemplazar un disco de agrupación raíz ZFS (EFI [GPT])" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.

4 Vuelva a crear la agrupación raíz.

En un sistema SPARC o x86 (VTOC):

```
sysA# zpool create rpool c2t0d0s0
```

En un sistema basado en x86 con un disco de agrupación raíz con etiqueta EFI (GPT), utilice una sintaxis similar a la siguiente:

```
sysA# zpool create -B rpool c2t0d0
```

5 Monte el sistema de archivos que contiene las instantáneas desde el sistema remoto.

```
sysA# mount -F nfs sysB:/tank/snaps /mnt
```

6 Restaure las instantáneas de la agrupación raíz.

```
sysA# gzcat /mnt/rpool.snap1.qz | zfs receive -Fv rpool
receiving full stream of rpool@rpool.snap1 into rpool@rpool.snap1
received 92.7KB stream in 1 seconds (92.7KB/sec)
receiving full stream of rpool/export@rpool.snap1 into rpool/export@rpool.snap1
received 47.9KB stream in 1 seconds (47.9KB/sec)
.
.
.
```

7 Vuelva a crear los dispositivos de intercambio y volcado, si es necesario.

Por ejemplo:

```
sysA# zfs create -V 4G rpool/swap
sysA# zfs create -V 4G rpool/dump
```

Para obtener información sobre el cambio de tamaño de los volúmenes de intercambio y volcado, consulte "Planificación para espacio de intercambio" de *Administración de Oracle Solaris 11.1: dispositivos y sistemas de archivos*.

8 Monte el entorno de inicio.

El siguiente paso requiere el montaje del entorno de inicio para que se puedan instalar los bloques de inicio.

```
sysA# beadm mount solaris /tmp/mnt
```

9 Instale los bloques de inicio en el nuevo disco.

Utilice el siguiente comando en un sistema basado en SPARC o x86.

```
sysA# bootadm install-bootloader -P rpool
```

10 Si los mismos dispositivos no se va a utilizar o los dispositivos se configurarán de manera diferente en el sistema original, borre la información del dispositivo existente. A continuación, indique al sistema para reconfigurar la información del nuevo dispositivo.

```
# devfsadm -Cn -r /tmp/mnt
# touch /tmp/mnt/reconfigure
```

11 Desmonte el entorno de inicio.

```
#beadm unmount solaris
```

12 Si es necesario, active el entorno de inicio.

Por ejemplo:

```
sysA# beadm list
BE        Active Mountpoint Space  Policy Created
--        ------ ---------- -----  ------ -------
solaris-1 -      -          46.95M static 2012-07-20 10:25
solaris   -      -           3.83G static 2012-07-19 13:44
# beadm activate solaris
```

13 Compruebe que puede iniciar el sistema correctamente desde el disco de reemplazo de la agrupación raíz.

Si es necesario, restablezca el dispositivo de inicio predeterminado:

- SPARC: configure el sistema para que se inicie automáticamente desde el disco nuevo, mediante el comando `eeprom` o el comando `setenv` desde la PROM de inicio.
- x86: vuelva a configurar el BIOS del sistema.

◆ ◆ ◆ CAPÍTULO 12

Prácticas de ZFS recomendadas por Oracle Solaris

Este capítulo describe las prácticas recomendadas para crear, supervisar y mantener las agrupaciones de almacenamiento y los sistemas de archivos de ZFS.

Este capítulo se divide en las secciones siguientes:

Prácticas recomendadas de agrupaciones de almacenamiento

Las siguientes secciones proporcionan las prácticas recomendadas para crear y supervisar agrupaciones de almacenamiento ZFS. Para obtener más información sobre la resolución de problemas de agrupaciones de almacenamiento, consulte el Capítulo 10, "Recuperación de agrupaciones y solución de problemas de Oracle Solaris ZFS".

Prácticas generales del sistema

- Mantener el sistema actualizado con los parches y las versiones más reciente de Solaris.
- Confirme que el controlador acepta comandos de vaciado de caché para saber que los datos se graban de forma segura, lo cual es importante antes de cambiar los dispositivos de la agrupación o dividir una agrupación de almacenamiento reflejada. Por lo general, esto no representa un problema en hardware de Oracle/Sun, pero es una buena práctica confirmar que está activada la configuración de vaciado de caché del hardware.
- Determinar los requisitos de memoria en virtud de la carga de trabajo real del sistema.
  - Con una huella de memoria de aplicación conocida, por ejemplo, para una aplicación de base de datos, puede limitar el tamaño de la ARC de modo de que la aplicación no necesite reclamar su memoria necesaria de la caché de ZFS.
  - Tenga en cuenta los requisitos de memoria para la eliminación de datos duplicados.

- Identifique el uso de la memoria de ZFS con el siguiente comando:

```
# mdb -k
> ::memstat
Page Summary                Pages                MB  %Tot
------------     ----------------  ----------------  ----
Kernel                     388117              1516   19%
ZFS File Data               81321               317    4%
Anon                        29928               116    1%
Exec and libs                1359                 5    0%
Page cache                   4890                19    0%
Free (cachelist)             6030                23    0%
Free (freelist)           1581183              6176   76%

Total                     2092828              8175
Physical                  2092827              8175
> $q
```

- Considere el uso de la memoria ECC para proteger contra los daños de memoria. Los daños silenciosos de la memoria pueden dañar los datos.

- Realizar copias de seguridad de forma regular. Aunque una agrupación creada con redundancia de ZFS puede ayudar a reducir el tiempo de inactividad debido a fallos de hardware, no es inmune a fallos de hardware, fallos de energía o cables desconectados. Asegúrese de que se realicen copias de seguridad de los datos de forma regular. Si los datos son importantes, se les debe realizar una copia de seguridad. A continuación, se enumeran diferentes formas de proporcionar copias de los datos:
  - Instantáneas de ZFS regulares o cotidianas.
  - Copias de seguridad semanales de los datos de la agrupación ZFS. Puede utilizar el comando `zpool split` para crear un duplicado exacto de la agrupación de almacenamiento ZFS reflejada.
  - Copias de seguridad mensuales utilizando un producto de copia de seguridad de nivel empresarial.
- RAID de hardware.
  - Considere el uso del modo JBOD para matrices de almacenamiento en lugar de RAID de hardware, para que ZFS pueda gestionar el almacenamiento y la redundancia.
  - Utilice RAID de hardware o redundancia de ZFS, o ambos.
  - El uso de redundancia de ZFS tiene muchas ventajas. Para los entornos de producción, configure ZFS para que pueda reparar las incoherencias de datos. Utilice redundancia de ZFS, como RAID-Z, RAID-Z-2, RAID-Z-3 y reflejo, independientemente del nivel de RAID implementado en el dispositivo de almacenamiento subyacente. Con la redundancia, las fallas en el dispositivo de almacenamiento subyacente o en sus conexiones con el host pueden ser detectadas y reparadas por ZFS.

  También consulte "Prácticas de creación de agrupaciones en matrices de almacenamiento locales o conectadas a la red" en la página 343.
- Los volcados por caída consumen más espacio en disco, generalmente entre 1/2 y 3/4 de tamaño en el rango de memoria física.

Prácticas de creación de agrupaciones de almacenamiento ZFS

Las siguientes secciones proporcionan prácticas de agrupaciones generales y más específicas.

Prácticas generales de agrupaciones de almacenamiento

- Utilizar discos enteros para activar la memoria caché de escritura de disco y para proporcionar mantenimiento más sencillo. Crear agrupaciones en segmentos agrega complejidad a la gestión y recuperación de discos.
- Utilizar la redundancia de ZFS para que ZFS pueda reparar las incoherencias de datos.
  - El siguiente mensaje aparece cuando se crea una agrupación no redundante:

    ```
    # zpool create tank c4t1d0 c4t3d0
    'tank' successfully created, but with no redundancy; failure
    of one device will cause loss of the pool
    ```
  - Para agrupaciones reflejadas, utilice pares de discos reflejados.
  - Para agrupaciones RAID-Z, agrupe de 3 a 9 discos por VDEV
  - No combine componentes RAID-Z y reflejados dentro de la misma agrupación. Estas agrupaciones son más difíciles de gestionar y el rendimiento puede verse afectado.
- Utilizar reservas activas para reducir el tiempo de inactividad debido a fallos de hardware.
- Utilizar discos de tamaño similar para que la E/S esté equilibrada entre dispositivos.
  - Los LUN más pequeños se pueden ampliar en LUN más grandes.
  - Para mantener tamaños óptimos de metaslabs, no expanda LUN de tamaños extremadamente distintos, como de 128 MB a 2 TB.
- Considerar la posibilidad de crear una agrupación raíz pequeña y agrupaciones de datos más grandes para admitir una recuperación del sistema más rápida.

Prácticas de creación de agrupaciones raíz

- **SPARC (SMI [VTOC])**: cree agrupaciones raíz con segmentos mediante el uso del identificador `s*`. No utilice el identificador `p*`. En general, la agrupación raíz ZFS de un sistema se crea cuando se instala el sistema. Si se crea una segunda agrupación raíz o se vuelve a crear una agrupación raíz, utilizar una sintaxis similar a la siguiente:

  ```
  # zpool create rpool c0t1d0s0
  ```

  O bien, crear una agrupación raíz reflejada. Por ejemplo:

  ```
  # zpool create rpool mirror c0t1d0s0 c0t2d0s0
  ```

- **x86 (EFI [GPT])**: cree agrupaciones raíz con discos enteros mediante el uso del identificador `d*`. No utilice el identificador `p*`. En general, la agrupación raíz ZFS de un sistema se crea cuando se instala el sistema. Si se crea una segunda agrupación raíz o se vuelve a crear una agrupación raíz, utilizar una sintaxis similar a la siguiente:

  ```
  # zpool create rpool c0t1d0
  ```

  O bien, crear una agrupación raíz reflejada. Por ejemplo:

  ```
  # zpool create rpool mirror c0t1d0 c0t2d0
  ```

- La agrupación raíz debe crearse como configuración reflejada o una configuración de un solo disco. No se admiten configuraciones RAID-Z ni distribuidas. No se pueden agregar discos adicionales para crear varios dispositivos virtuales reflejados de nivel superior mediante el comando `zpool add`, pero se puede ampliar un dispositivo virtual reflejado mediante el comando `zpool attach`.
- Una agrupación raíz no puede tener un dispositivo de registro independiente.
- Se pueden establecer las propiedades de agrupaciones durante una instalación de AI, pero el algoritmo de compresión `gzip` no se admite en las agrupaciones raíz.
- No cambie el nombre de la agrupación raíz tras su creación en una instalación inicial. El cambio de nombre de la agrupación raíz puede impedir el inicio del sistema.
- No cree una agrupación raíz en una unidad USB para un sistema de producción porque los discos de agrupación raíz son fundamentales para la operación continua, en especial, en un entorno empresarial. Considere la posibilidad de utilizar discos internos del sistema para la agrupación raíz o, al menos, utilice discos de la misma calidad que utilizaría para datos no raíz. Además, es posible que una unidad USB no sea suficientemente grande para admitir un tamaño de volumen de volcado que sea equivalente a la memoria física o que tenga al menos la mitad de su tamaño.

Prácticas de creación de agrupaciones (de datos) que no son raíz

- Crear agrupaciones que no son raíz con discos enteros mediante el identificador `d*`. No utilizar el identificador `p*`.
  - ZFS tiene un funcionamiento óptimo sin ningún software de administración de volumen adicional.
  - Para tener un mejor rendimiento, utilice discos individuales o, al menos, LUN formados con pocos discos. Al proporcionar ZFS con más visibilidad de la configuración de LUN, ZFS puede tomar mejores decisiones de programación de E/S.
  - Cree configuraciones de agrupaciones redundantes en varios controladores para reducir el tiempo de inactividad debido a fallos del controlador.
  - **Agrupaciones de almacenamiento reflejadas**: consuma más espacio en el disco pero, en general, obtenga un mejor rendimiento con lecturas aleatorias pequeñas.

    ```
    # zpool create tank mirror c1d0 c2d0 mirror c3d0 c4d0
    ```

- **Agrupaciones de almacenamiento RAID-Z**: se pueden crear con 3 estrategias de paridad, donde la paridad es igual a 1 (`raidz`), 2 (`raidz2`) o 3 (`raidz3`). Una configuración de RAID-Z maximiza el espacio en el disco y suele funcionar bien cuando los datos se escriben y se leen en grandes cantidades (128 K o más).
  - Considere una configuración RAID-Z de paridad única (`raidz`) con 2 VDEV de 3 discos (2+1) cada uno.

    ```
    # zpool create rzpool raidz1 c1t0d0 c2t0d0 c3t0d0 raidz1 c1t1d0 c2t1d0 c3t1d0
    ```
  - Una configuración RAIDZ-2 ofrece mejor disponibilidad de datos y se desempeña de manera similar a RAID-Z. RAIDZ-2 tiene un tiempo medio significativamente mejor para la pérdida de datos (MTTDL) que los reflejos RAID-Z o bidireccionales. Cree una configuración de RAID-Z de paridad doble (`raidz2`) en 6 discos (4+2).

    ```
    # zpool create rzpool raidz2 c0t1d0 c1t1d0 c4t1d0 c5t1d0 c6t1d0 c7t1d0
    raidz2 c0t2d0 c1t2d0 c4t2d0 c5t2d0 c6t2d0 c7t2d
    ```
  - La configuración RAIDZ-3 maximiza el espacio en disco y ofrece una excelente disponibilidad porque puede resistir 3 fallos de disco. Cree una configuración RAID-Z de paridad triple (`raidz3`) en 9 discos (6+3).

    ```
    # zpool create rzpool raidz3 c0t0d0 c1t0d0 c2t0d0 c3t0d0 c4t0d0
    c5t0d0 c6t0d0 c7t0d0 c8t0d0
    ```

Prácticas de creación de agrupaciones en matrices de almacenamiento locales o conectadas a la red

Tenga en cuenta las siguientes prácticas de agrupación de almacenamiento al crear una agrupación de almacenamiento ZFS en una matriz conectada de forma local o remota.

- Si crea una agrupación en dispositivos SAN y la conexión de red es lenta, es posible que los dispositivos de la agrupación tengan el estado `UNAVAIL` durante un tiempo. Debe evaluar si la conexión de red es adecuada para proporcionar los datos de forma continua. Además, tenga en cuenta que si utiliza dispositivos SAN para la agrupación raíz, es posible que no estén disponibles en cuanto se inicia el sistema y es posible que los dispositivos de la agrupación raíz también tengan el estado `UNAVAIL`.
- Confirme con el proveedor de matrices que la matriz de discos no vacía la caché después de que ZFS emite una solicitud de vaciado de la caché de escritura.
- Utilice discos enteros, no segmentos de discos, como dispositivos de la agrupación de almacenamiento, de modo que Oracle Solaris ZFS active las cachés de discos locales pequeños, que se vacían en momentos adecuados.
- Para obtener el mejor rendimiento, cree un LUN para cada disco físico en la matriz. Si se utiliza un solo LUN grande, es posible que ZFS ponga en cola muy pocas operaciones de E/S de lectura para un rendimiento óptimo del almacenamiento. Por el contrario, si se utilizan muchos LUN pequeños, es posible que el almacenamiento se llene con una gran cantidad de operaciones de E/S de lectura pendientes.

- Para Oracle Solaris ZFS, no se recomienda utilizar una matriz de almacenamiento que utiliza software de aprovisionamiento dinámico (o delgado) para implementar la asignación de espacio virtual. Cuando Oracle Solaris ZFS escribe los datos modificados en el espacio libre, escribe en todo el LUN. El proceso de escritura de Oracle Solaris ZFS asigna todo el espacio virtual desde el punto de vista de la matriz de almacenamiento, lo cual invalida el beneficio del aprovisionamiento dinámico.

  Tenga en cuenta que el software de aprovisionamiento dinámico puede ser innecesario al utilizar ZFS:

  - Puede ampliar un LUN en una agrupación de almacenamiento ZFS existente y utilizará el nuevo espacio.
  - También funciona un comportamiento similar cuando un LUN más pequeño se reemplaza con uno de mayor tamaño.
  - Si evalúa las necesidades de almacenamiento para su agrupación y crea la agrupación con LUN más pequeños iguales a las necesidades de almacenamiento, más adelante puede ampliar los LUN a un tamaño mayor si se necesita más espacio.
- Si es posible que la matriz presente dispositivos individuales (modo JBOD), considere la posibilidad de crear agrupaciones de almacenamiento ZFS redundantes (reflejadas o RAID-Z) en este tipo de matriz de modo que ZFS pueda comunicar y corregir las inconsistencias de datos.

Prácticas de creación de agrupaciones para una base de datos Oracle

Tenga en cuenta las siguientes prácticas de agrupación de almacenamiento al crear una base de datos Oracle.

- Utilizar una agrupación reflejada o un RAID de hardware para agrupaciones.
- Las agrupaciones RAID-Z por lo general no se recomiendan para lectura aleatorias cargas de trabajo aleatorias.
- Crear una pequeña agrupación independiente con un dispositivo de registro independiente para los registros de rehacer de la base de datos.
- Crear una pequeña agrupación independiente para el registro de archivo.

Para obtener más información, consulte la siguiente documentación técnica:

`http://blogs.oracle.com/storage/entry/new_white_paper_configuring_oracle`

Uso de agrupaciones de almacenamiento ZFS en VirtualBox

- De manera predeterminada, Virtual Box está configurada para omitir comandos de vaciado de caché desde el almacenamiento subyacente. Esto significa que, en caso de un bloqueo del sistema o un error de hardware, se pueden perder los datos.
- Ejecute el siguiente comando para activar el vaciado de caché en Virtual Box:

```
VBoxManage setextradata <VM_NAME> "VBoxInternal/Devices/<type>/0/LUN#<n>/Config/IgnoreFlush" 0
```

- <VM_NAME> es el nombre de la máquina virtual
- <type> es el tipo de controlador, que puede ser `piix3ide` (si utiliza el controlador virtual IDE habitual) o `ahci`, si utiliza un controlador SATA
- <n> es el número de disco

Prácticas de agrupaciones de almacenamiento para rendimiento

- Mantener la capacidad de la agrupación por debajo del 90% para obtener el mejor rendimiento.
- Se recomiendan las agrupaciones reflejadas en lugar de las agrupaciones RAID-Z para cargas de trabajo de lectura/escritura aleatoria.
- Separar dispositivos de registro.
  - Recomendado para mejorar el rendimiento de la escritura síncrona.
  - Con una alta carga de escritura síncrona, se evita la fragmentación de escribir muchos los bloques de registros en la agrupación principal.
- Se recomiendan los dispositivos de caché independientes para mejorar el rendimiento de lectura.
- Limpieza/creación: una agrupación RAID-Z muy grande con muchos dispositivos tendrá tiempos de limpieza y creación más largos.
- Rendimiento de la agrupación lento: utilice el comando `zpool status` para descartar cualquier problema de hardware que esté causando problemas de rendimiento en la agrupación. Si no aparece ningún problema en el comando `zpool status`, utilice el comando `fmdump` para mostrar los fallos de hardware, o utilice el comando `fmdump -eV` para revisar los errores de hardware que todavía no han provocado un fallo.

Prácticas de supervisión y mantenimiento de agrupaciones de almacenamiento ZFS

- Asegúrese de que la capacidad de agrupación esté por debajo del 90% para obtener el mejor rendimiento.

  El rendimiento de la agrupación se puede degradar cuando una agrupación está muy llena y los sistemas de archivos se actualizan con frecuencia, como en un servidor de correo muy ocupado. Las agrupaciones llenas pueden ocasionar una penalización del rendimiento, pero no otros problemas. Si la carga de trabajo principal es de archivos inmutables, mantenga la agrupación en el rango de uso entre un 95 y 96%. Incluso si el contenido más estático está en el rango entre 95 y 96%, se pueden ver perjudicados los rendimientos de escritura, lectura y creación.

  - Supervise el espacio de la agrupación y del sistema de archivos para asegurarse de que no estén llenos.
  - Evalúe la posibilidad de usar reservas y cuotas ZFS a fin de garantizar que el espacio del sistema de archivos no supere el 90% de la capacidad de la agrupación.
- Supervise el estado de la agrupación.
  - Agrupaciones redundantes: supervise la agrupación con `zpool status` y `fmdump` semanalmente.
  - Agrupaciones no redundantes: supervise la agrupación con `zpool status` y `fmdump` bisemanalmente.
- Ejecute `zpool scrub` de forma regular para identificar problemas de integridad de los datos.
  - Si tiene unidades de calidad de consumidor, trate de programar una limpieza semanal.
  - Si tiene unidades de calidad de centro de datos, trate de programar una limpieza mensual.
  - También debería realizar una limpieza antes de reemplazar dispositivos o reducir temporalmente la redundancia de una agrupación para asegurarse de que todos los dispositivos se encuentren en funcionamiento.
- Supervise las fallas de la agrupación o del dispositivo. Use `zpool status` como se describe a continuación. También use `fmdump` o `fmdump -eV` para ver si se produjo alguna falla o error de dispositivo.
  - Agrupaciones redundantes: supervise el estado de la agrupación con `zpool status` y `fmdump` semanalmente.
  - Agrupaciones no redundantes: supervise el estado de la agrupación con `zpool status` y `fmdump` bisemanalmente
- El dispositivo de la agrupación está `UNAVAIL` u `OFFLINE` - Si el dispositivo de una agrupación no está disponible, compruebe que el dispositivo se muestre en la salida del comando `format`. Si el dispositivo no se muestra en la salida de `format`, no estará visible para ZFS.

  Si el dispositivo de una agrupación está `UNAVAIL` u `OFFLINE`, en general, esto significa que el dispositivo ha fallado o que el cable se ha desconectado, o algún otro problema de hardware, como un cable o controlador incorrectos que han provocado que el dispositivo sea inaccesible.
- Considerar la configuración del servicio `smtp-notify` para que notifique cuando un componente de hardware se diagnostique como defectuoso. Para obtener más información, consulte sección Parámetros de notificación de `smf(5)` and `smtp-notify(1M)`.

  De manera predeterminada, algunas notificaciones se configuran de forma automática para ser enviadas al usuario root. Si agrega un alias para la cuenta de usuario como root en el archivo `/etc/aliases`, recibirá notificaciones por correo electrónico, similares a la siguiente:

```
From noaccess@tardis.space.com Fri Jun 29 16:58:59 2012
Date: Fri, 29 Jun 2012 16:58:58 -0600 (MDT)
From: No Access User <noaccess@tardis.space.com>
```

```
Message-Id: <201206292258.q5TMwwFL002753@tardis.space.com>
Subject: Fault Management Event: tardis:ZFS-8000-8A
To: root@tardis.space.com
Content-Length: 771

SUNW-MSG-ID: ZFS-8000-8A, TYPE: Fault, VER: 1, SEVERITY: Critical
EVENT-TIME: Fri Jun 29 16:58:58 MDT 2012
PLATFORM: ORCL,SPARC-T3-4, CSN: 1120BDRCCD, HOSTNAME: tardis
SOURCE: zfs-diagnosis, REV: 1.0
EVENT-ID: 76c2d1d1-4631-4220-dbbc-a3574b1ee807
DESC: A file or directory in pool 'pond' could not be read due to corrupt data.
AUTO-RESPONSE: No automated response will occur.
IMPACT: The file or directory is unavailable.
REC-ACTION: Use 'fmadm faulty' to provide a more detailed view of this event.
Run 'zpool status -xv' and examine the list of damaged files to determine what
has been affected. Please refer to the associated reference document at
http://support.oracle.com/msg/ZFS-8000-8A for the latest service procedures
and policies regarding this diagnosis.
```

- Supervisar el espacio de la agrupación de almacenamiento - Utilice el comando `zpool list` y el comando `zfs list` para identificar la cantidad de disco que consumen los datos del sistema de archivos. Las instantáneas de ZFS pueden consumir espacio en disco y, si no están enumeradas por el comando `zfs list`, también pueden consumir espacio en disco en modo silencioso. Utilice el comando de instantánea `zfs list - t` para identificar el espacio en disco consumido por las instantáneas.

Prácticas recomendadas de sistemas de archivos

En las siguientes secciones, se describen las prácticas recomendadas de sistemas de archivos.

Prácticas de creación de sistemas de archivos

En las siguientes secciones, se describen las prácticas de creación de sistemas de archivos ZFS.

- Crear un sistema de archivos por usuario para los directorios de inicio.
- Considerar el uso de cuotas y reservas del sistema de archivos para gestionar y reservar espacio en disco para sistemas de archivos importantes.
- Considerar el uso de cuotas de grupo y usuario para gestionar el espacio en disco en un entorno con varios usuarios.
- Utilizar la herencia de propiedades de ZFS para aplicar las propiedades a varios sistemas de archivos descendientes.

Prácticas de creación de sistemas de archivos para una base de datos Oracle

Tenga en cuenta las siguientes prácticas de sistema de archivos al crear una base de datos Oracle.

- Hacer coincidir la propiedad `recordsize` de ZFS con `db_block_size` de Oracle.
- Crear los sistemas de archivos de índice y tabla de base de datos en la agrupación de base de datos principal, utilizando un `recordsize` de 8 KB y el valor predeterminado `primarycache`.
- Crear datos temporales y sistemas de archivos de espacio de tabla de deshacer en la agrupación de base de datos principal, utilizando los valores predeterminados `recordsize` y `primarycache`.
- Crear sistema de archivos de registro de archivo en la agrupación de archivo mediante la activación de la compresión y el valor predeterminado `recordsize`, y establecer `primarycache` en `metadata`.

Para obtener más información, consulte "Ajuste de ZFS para una base de datos Oracle" de *Manual de referencia de parámetros ajustables de Oracle Solaris 11.1*.

Prácticas de supervisión de sistema de archivos ZFS

Debe supervisar los sistemas de archivos ZFS para asegurarse de que están disponibles y para identificar problemas de consumo de espacio.

- Semanalmente, supervise la disponibilidad del espacio del sistema de archivos con los comandos `zpool list` y `zfs list` en lugar de utilizar los comandos `du` y `df`, ya que los comandos heredados no rinden cuenta del espacio que utilizan los sistemas de archivos descendiente ni las instantáneas.

  Para obtener más información, consulte "Resolución de problemas de espacio ZFS" en la página 297.
- Visualice el consumo de espacio del sistema de archivos mediante el comando `zfs list -o space`.
- El espacio del sistema de archivos puede ser consumido por las instantáneas sin tener conocimiento de ello. Puede visualizar toda la información del conjunto de datos mediante la siguiente sintaxis:

  ```
  # zfs list -t all
  ```
- Un sistema de archivos `/var` independiente se crea de forma automática cuando se instala un sistema, pero usted debe definir una cuota y una reserva en este sistema de archivos para asegurarse de que dicho sistema no consume espacio de la agrupación raíz sin tener conocimiento de ello.
- Además, puede utilizar el comando `fsstat` para visualizar la actividad de operación de archivo de sistemas de archivos ZFS. Se informa de la actividad por punto de montaje o por tipo de sistema de archivos. El ejemplo siguiente muestra la actividad general del sistema de archivos ZFS:

  ```
  # fsstat /
   new  name   name  attr  attr lookup rddir  read read  write write
  ```

```
 file remov  chng   get   set    ops   ops   ops bytes   ops bytes
  832   589   286  837K 3.23K  2.62M 20.8K 1.15M 1.75G 62.5K  348M /
```

- Copias de seguridad.
  - Mantenga las instantáneas del sistema de archivos.
  - Considere el uso de un software en el nivel de la empresa para la realización de copias de seguridad semanales y mensuales.
  - Almacene las instantáneas de agrupaciones raíz en un sistema remoto para la reconstrucción completa.

◆ ◆ ◆ APÉNDICE A

Descripciones de versiones de Oracle Solaris ZFS

Este apéndice describe versiones de ZFS disponibles, las características de cada versión y el sistema operativo Solaris pertinente.

Este apéndice contiene las secciones siguientes:

Información general de versiones de ZFS

El uso de una versión concreta de ZFS disponible en versiones de Solaris da acceso a nuevas funciones de sistema de archivos y agrupación de ZFS. Puede utilizar uno de los comandos `zpool upgrade` o `zfs upgrade` para identificar si una agrupación o un sistema de archivos es de una versión anterior a la suministrada con la versión de Solaris vigente. También puede usar estos comandos para actualizar sus versiones de sistema de archivos y agrupación.

Para obtener información sobre el uso de los comandos `zpool upgrade` y `zfs upgrade`, consulte "Actualización de sistemas de archivos ZFS" en la página 216 y "Actualización de agrupaciones de almacenamiento de ZFS" en la página 115.

Versiones de agrupación de ZFS

La siguiente tabla proporciona una lista de versiones de agrupaciones de ZFS disponibles en la versión Oracle Solaris.

Versión	Oracle Solaris 11	Descripción
1	snv_36	Versión ZFS inicial

Versión	Oracle Solaris 11	Descripción
2	snv_38	Bloques ditto (metadatos repetidos)
3	snv_42	Repuestos en marcha y RAID-Z de doble paridad
4	snv_62	`zpool history`
5	snv_62	Algoritmo de compresión `gzip`
6	snv_62	Propiedad de agrupación `bootfs`
7	snv_68	Dispositivos de registro con diferente función
8	snv_69	Administración delegada
9	snv_77	Propiedades `refquota` y `refreservation`
10	snv_78	Dispositivos de caché
11	snv_94	Rendimiento de limpieza mejorado
12	snv_96	Propiedades de instantáneas
13	snv_98	Propiedad `Snapused`
14	snv_103	Propiedad `aclinherit` `passthrough-x`
15	snv_114	Cálculo de espacio de agrupación y usuario
16	snv_116	Propiedad `stmf`
17	snv_120	RAID-Z de triple paridad
18	snv_121	Retenciones de instantánea
19	snv_125	Eliminación de dispositivo de registro
20	snv_128	Algoritmo de compresión (codificación de longitud cero) `zle`
21	snv_128	Eliminación de datos duplicados
22	snv_128	Propiedades recibidas
23	snv_135	ZIL limitado
24	snv_137	Atributos del sistema
25	snv_140	Estadísticas de limpieza mejorada
26	snv_141	Rendimiento mejorado de supresión de instantáneas
27	snv_145	Rendimiento mejorado de creación de instantáneas
28	snv_147	Reemplazos de múltiples de dispositivos virtuales
29	snv_148	Asignador híbrido de reflejo/RAID-Z

Versión	Oracle Solaris 11	Descripción
30	snv_149	Cifrado
31	snv_150	Rendimiento de "zfs list" mejorado
32	snv_151	Tamaño de bloque de 1 MB
33	snv_163	Compatibilidad con recurso compartido mejorada
34	S11.1	Uso compartido con herencia

Versiones de sistema de archivos ZFS

La siguiente tabla muestra las versiones de sistemas de archivos ZFS disponibles en la versión Oracle Solaris. Tenga en cuenta que las funciones que están disponibles en determinadas versiones del sistema de archivos requieren una versión específica de agrupación.

Versión	Oracle Solaris 11	Descripción
1	snv_36	Versión inicial de sistemas de archivos ZFS
2	snv_69	Entradas de directorio mejoradas
3	snv_77	Sin distinción de mayús-minús e identificador exclusivo de sistema de archivo (FUID)
4	snv_114	Propiedades `userquota` y `groupquota`
5	snv_137	Atributos del sistema
6	S11.1	Compatibilidad con sistemas de archivos de varios niveles

Índice

B

C

D

E

F

G

M

N

P

R

S

T

U

V

Z