

NAME

zfs-create - create ZFS dataset

SYNOPSIS

zfs create [-Pnpuv] [-o *property=value*]<?> *filesystem*

zfs create [-ps] [-b *blocksize*] [-o *property=value*]<?> -V *size volume*

DESCRIPTION

zfs create [-Pnpuv] [-o *property=value*]<?> *filesystem*

Creates a new ZFS file system. The file system is automatically mounted according to the **mountpoint** property inherited from the parent, unless the **-u** option is used.

-o *property=value*

Sets the specified property as if the command **zfs set *property=value*** was invoked at the same time the dataset was created. Any editable ZFS property can also be set at creation time. Multiple **-o** options can be specified. An error results if the same property is specified in multiple **-o** options.

-p Creates all the non-existing parent datasets. Datasets created in this manner are automatically mounted according to the **mountpoint** property inherited from their parent. Any property specified on the command line using the **-o** option is ignored. If the target filesystem already exists, the operation completes successfully.

-n Do a dry-run ("No-op") creation. No datasets will be created. This is useful in conjunction with the **-v** or **-P** flags to validate properties that are passed via **-o** options and those implied by other options. The actual dataset creation can still fail due to insufficient privileges or available capacity.

-P

Print machine-parsable verbose information about the created dataset. Each line of output contains a key and one or two values, all separated by tabs. The **create_ancestors** and **create** keys have *filesystem* as their only value. The **create_ancestors** key only appears if the **-p** option is used. The **property** key has two values, a property name that property's value. The **property** key may appear zero or more times, once for each property that will be set local to *filesystem* due to the use of the **-o** option.

-u Do not mount the newly created file system.

-v Print verbose information about the created dataset.

zfs create [-ps] [-b *blocksize*] [-o *property=value*]<?> -V *size volume*

Creates a volume of the given size. The volume is exported as a block device in */dev/zvol/path*, where

path is the name of the volume in the ZFS namespace. The size represents the logical size as exported by the device. By default, a reservation of equal size is created.

size is automatically rounded up to the nearest multiple of the **blocksize**.

-b *blocksize*

Equivalent to **-o volblocksize=blocksize**. If this option is specified in conjunction with **-o volblocksize**, the resulting behavior is undefined.

-o *property=value*

Sets the specified property as if the **zfs set property=value** command was invoked at the same time the dataset was created. Any editable ZFS property can also be set at creation time. Multiple **-o** options can be specified. An error results if the same property is specified in multiple **-o** options.

-p Creates all the non-existing parent datasets. Datasets created in this manner are automatically mounted according to the **mountpoint** property inherited from their parent. Any property specified on the command line using the **-o** option is ignored. If the target filesystem already exists, the operation completes successfully.

-s Creates a sparse volume with no reservation. See **volsize** in the *Native Properties* section of `zfsprops(7)` for more information about sparse volumes.

-n Do a dry-run ("No-op") creation. No datasets will be created. This is useful in conjunction with the **-v** or **-P** flags to validate properties that are passed via **-o** options and those implied by other options. The actual dataset creation can still fail due to insufficient privileges or available capacity.

-P

Print machine-parsable verbose information about the created dataset. Each line of output contains a key and one or two values, all separated by tabs. The **create_ancestors** and **create** keys have *volume* as their only value. The **create_ancestors** key only appears if the **-p** option is used. The **property** key has two values, a property name that property's value. The **property** key may appear zero or more times, once for each property that will be set local to *volume* due to the use of the **-b** or **-o** options, as well as **refreservation** if the volume is not sparse.

-v Print verbose information about the created dataset.

ZFS for Swap

Swapping to a ZFS volume is prone to deadlock and not recommended. See OpenZFS FAQ.

Swapping to a file on a ZFS filesystem is not supported.

EXAMPLES

Example 1: Creating a ZFS File System Hierarchy

The following commands create a file system named *pool/home* and a file system named *pool/home/bob*. The mount point */export/home* is set for the parent file system, and is automatically inherited by the child file system.

```
# zfs create pool/home
# zfs set mountpoint=/export/home pool/home
# zfs create pool/home/bob
```

Example 2: Promoting a ZFS Clone

The following commands illustrate how to test out changes to a file system, and then replace the original file system with the changed one, using clones, clone promotion, and renaming:

```
# zfs create pool/project/production
  populate /pool/project/production with data
# zfs snapshot pool/project/production@today
# zfs clone pool/project/production@today pool/project/beta
  make changes to /pool/project/beta and test them
# zfs promote pool/project/beta
# zfs rename pool/project/production pool/project/legacy
# zfs rename pool/project/beta pool/project/production
  once the legacy version is no longer needed, it can be destroyed
# zfs destroy pool/project/legacy
```

SEE ALSO

zfs-destroy(8), *zfs-list*(8), *zpool-create*(8)