

NAME

zpool-trim - initiate TRIM of free space in ZFS storage pool

SYNOPSIS

zpool trim [-**dw**] [-**r** *rate*] [-**c**|-**s**] *pool* [*device*]<?>

DESCRIPTION

Initiates an immediate on-demand TRIM operation for all of the free space in a pool. This operation informs the underlying storage devices of all blocks in the pool which are no longer allocated and allows thinly provisioned devices to reclaim the space.

A manual on-demand TRIM operation can be initiated irrespective of the **autotrim** pool property setting. See the documentation for the **autotrim** property above for the types of vdev devices which can be trimmed.

-d, --secure

Causes a secure TRIM to be initiated. When performing a secure TRIM, the device guarantees that data stored on the trimmed blocks has been erased. This requires support from the device and is not supported by all SSDs.

-r, --rate *rate*

Controls the rate at which the TRIM operation progresses. Without this option TRIM is executed as quickly as possible. The rate, expressed in bytes per second, is applied on a per-vdev basis and may be set differently for each leaf vdev.

-c, --cancel

Cancel trimming on the specified devices, or all eligible devices if none are specified. If one or more target devices are invalid or are not currently being trimmed, the command will fail and no cancellation will occur on any device.

-s, --suspend

Suspend trimming on the specified devices, or all eligible devices if none are specified. If one or more target devices are invalid or are not currently being trimmed, the command will fail and no suspension will occur on any device. Trimming can then be resumed by running **zpool trim** with no flags on the relevant target devices.

-w, --wait

Wait until the devices are done being trimmed before returning.

PERIODIC TRIM

On machines using systemd, trim timers can be enabled on a per-pool basis. **weekly** and **monthly** timer units are provided.

```
systemctl enable zfs-trim-weekly@rpool.timer --now
```

```
systemctl enable zfs-trim-monthly@otherpool.timer --now
```

SEE ALSO

systemd.timer(5), zpoolprops(7), zpool-initialize(8), zpool-wait(8)