NAME

truncate — trim or punch holes in files

SYNOPSIS

truncate **-s** *size* [**-co**] *file...*
truncate **-r** *ref* [**-c**] *file...*

DESCRIPTION

Sets the size of *file*s according to a *size* expression or to match *ref*. If the new size is smaller, data beyond it is lost; otherwise, it reads zero (and, if supported, may be a hole).

Unless **-c** is specified, files are created, otherwise nonexistent files are ignored.

With **-s**, *size* is in the case-insensitive format:
[**+-<>/%**][*base*][**KMGTPEZY**][**B** (with at least one of {*base*, **KMGTPEZY**, **B**})]
Where *base* is an optionally-floating-point number of bytes, defaulting to **1**, which is then optionally multiplied by the relevant unit. **B** sets the unit multiplier to **1000** (from **1024**). *size* is equal to $base \cdot unit^{mult}$, if any, or *base*.

If **-o** is specified, *size* is multiplied by each *file*'s filesystem's block size (*st_blksize* stat(2) field).

By default, *file*s are simply assigned *size*, but if any of **+-<>/%** is specified at the start:

+	*fs* + *sz*	*size* is added to the *file*'s size
−	*fs* - *sz*	*size* is subtracted from the *file*'s size
<	**min**(*fs*, *sz*)	smaller of *size* and the *file*'s size is used
>	**max**(*fs*, *sz*)	larger of *size* and the *file*'s size is used
/	*fs* - (*fs* % *sz*)	largest multiple of *size* not larger than the *file*'s size (rounded down)
%	**min**(*fs* + (*sz* - (*fs* % *sz*)), *fs*)	smallest multiple of *size* not smaller than the *file*'s size (rounded up)

The result is then curried with **max**(*0*).

With **-r**, *file*s are truncated to the size of *ref*.

File sizes are determined as their final seekable position. This is the same as the stated size for regular files, but it also means, that *ref* can trivially be a file or a block device, but not a FIFO, and directories will produce meaningless, potentially huge, results, Regular files need only be stat(2)table, but others must be able to be read.

OPTIONS

-c, **--no-create** Don't create *file*s and ignore nonexistent ones. The default is to create *file*s with a=rw − *umask* if they don't already exist.

-s, **--size**=*size* Set *file*s' sizes to *size*, see above.
-o, **--io-blocks** Multiply *size* by the *file*'s I/O block size.

-r, **--reference**=*ref* Set *file*s' sizes to equal *ref*'s.

EXIT STATUS

1 if a file's size couldn't be determined when needed, some *file*s couldn't be opened (with the exception of ENOENT with **-c**) or truncated.

EXAMPLES

Create a multiple-of-16GB file at least as large as the swap area:

```
$ truncate -r /dev/zvol/babzoot/swap swap
truncate: /dev/zvol/babzoot/swap: Permission denied
# truncate -r /dev/zvol/babzoot/swap swap
# ls -lb swap
-rw-r--r-- 1 root staff 4294967296 Jul  3 22:07 swap
# truncate -s %16G swap
# ls -lb swap
-rw-r--r-- 1 root staff 17179869184 Jul  3 22:07 swap
```

Create files 1100 bytes longer than twice the block size:

```
$ truncate -os 2 ~/long /pub/er
$ truncate -s +1.1kB ~/long /pub/er
$ ls -lb ~/long /pub/er
-rw-r--r-- 1 cicada users   9292 Jul  3 22:01 /home/cicada/long
-rw-r--r-- 1 cicada users 263244 Jul  3 22:01 /pub/er
```

`/pub` is a `zfs`(4) dataset with **recordsize**=*128K*, whereas `/home` is a `ext4`(5) filesystem created with **mke2fs -b** *4096*.

SEE ALSO

`ftruncate`(2), `lseek`(2), `stat`(2)

HISTORY

Originates from the GNU system, also available in FreeBSD 4.2.

The GNU system disallows *size* with **B** but without a multiplier, as well as lower-case **B**, and only supports integer *bases*.

FreeBSD doesn't support **/%**, **B**, nor **-o**.