

**NAME**

**reboot, halt, fastboot, fasthalt** - stopping and restarting the system

**SYNOPSIS**

**halt** [-lNnpq] [-k *kernel*]

**reboot** [-cdlNnpqr] [-k *kernel*]

**fasthalt** [-lNnpq] [-k *kernel*]

**fastboot** [-dlNnpq] [-k *kernel*]

**DESCRIPTION**

The **halt** and **reboot** utilities flush the file system cache to disk, send all running processes a SIGTERM (and subsequently a SIGKILL) and, respectively, halt or restart the system. The action is logged, including entering a shutdown record into the user accounting database.

The options are as follows:

- c** The system will turn off the power and then turn it back on if it can. If the power down action fails, the system will halt or reboot normally, depending on whether **halt** or **reboot** was called. At the present time, only the ipmi(4) driver implements the power cycle functionality and only on hardware with a BMC that supports power cycling. Unlike power off, the amount of hardware that supports power cycling is small.
- d** The system is requested to create a crash dump. This option is supported only when rebooting, and it has no effect unless a dump device has previously been specified with dumpon(8).
- k *kernel***  
Boot the specified *kernel* on the next system boot. If the kernel boots successfully, the *default* kernel will be booted on successive boots, this is a one-shot option. If the boot fails, the system will continue attempting to boot *kernel* until the boot process is interrupted and a valid kernel booted. This may change in the future.
- l** The halt or reboot is *not* logged to the system log. This option is intended for applications such as shutdown(8), that call **reboot** or **halt** and log this themselves.
- N** The file system cache is not flushed during the initial process clean-up, however the kernel level reboot(2) is still processed with a sync. This option can be useful for performing a "best-effort" reboot when devices might be unavailable. This can happen when devices have been disconnected, such as with iscsi(4).
- n** The file system cache is not flushed. This option should probably not be used.

- p** The system will turn off the power if it can. If the power down action fails, the system will halt or reboot normally, depending on whether **halt** or **reboot** was called.
- q** The system is halted or restarted quickly and ungracefully, and only the flushing of the file system cache is performed (if the **-n** option is not specified). This option should probably not be used.
- r** The system kills all processes, unmounts all filesystems, mounts the new root filesystem, and begins the usual startup sequence. After changing `vfs.root.mountfrom` with `kenv(1)`, **reboot -r** can be used to change the root filesystem while preserving kernel state. This requires the `tmpfs(5)` kernel module to be loaded because `init(8)` needs a place to store itself after the old root is unmounted, but before the new root is in place.

The **fasthalt** and **fastboot** utilities are nothing more than aliases for the **halt** and **reboot** utilities.

Normally, the `shutdown(8)` utility is used when the system needs to be halted or restarted, giving users advance warning of their impending doom and cleanly terminating specific programs.

## EXAMPLES

Replace current root filesystem with UFS mounted from `/dev/ada0s1a`:

```
kenv vfs.root.mountfrom=ufs:/dev/ada0s1a
reboot -r
```

This mechanism can also be used with NFS, with a caveat that it only works with NFSv4, and requires a numeric IPv4 address:

```
kenv vfs.root.mountfrom=nfs:192.168.1.1:/share/name
reboot -r
```

## SEE ALSO

`kenv(1)`, `getutxent(3)`, `ipmi(4)`, `boot(8)`, `dumpon(8)`, `nextboot(8)`, `savecore(8)`, `shutdown(8)`, `sync(8)`

## HISTORY

A **reboot** utility appeared in 4.0BSD.