NAME
reboot − reboot or enable/disable Ctrl-Alt-Del

LIBRARY
Standard C library (*libc*, *−lc*)

SYNOPSIS
/* Since Linux 2.1.30 there are symbolic names **LINUX_REBOOT_***
   for the constants and a fourth argument to the call: */

#include <linux/reboot.h> /* Definition of **LINUX_REBOOT_*** constants */
#include <sys/syscall.h>   /* Definition of **SYS_*** constants */
#include <unistd.h>

int syscall(SYS_reboot, int *magic***, int** *magic2***, int** *cmd***, void ****arg***);**

/* Under glibc and most alternative libc's (including uclibc, dietlibc,
   musl and a few others), some of the constants involved have gotten
   symbolic names **RB_***, and the library call is a 1-argument
   wrapper around the system call: */

#include <sys/reboot.h>   /* Definition of **RB_*** constants */
#include <unistd.h>

int reboot(int *cmd***);**

DESCRIPTION
The **reboot**() call reboots the system, or enables/disables the reboot keystroke (abbreviated CAD, since the default is Ctrl-Alt-Delete; it can be changed using **loadkeys**(1)).

This system call fails (with the error **EINVAL**) unless *magic* equals **LINUX_REBOOT_MAGIC1** (that is, 0xfee1dead) and *magic2* equals **LINUX_REBOOT_MAGIC2** (that is, 0x28121969). However, since Linux 2.1.17 also **LINUX_REBOOT_MAGIC2A** (that is, 0x05121996) and since Linux 2.1.97 also **LINUX_REBOOT_MAGIC2B** (that is, 0x16041998) and since Linux 2.5.71 also **LINUX_RE-BOOT_MAGIC2C** (that is, 0x20112000) are permitted as values for *magic2*. (The hexadecimal values of these constants are meaningful.)

The *cmd* argument can have the following values:

LINUX_REBOOT_CMD_CAD_OFF
>   (**RB_DISABLE_CAD**, 0). CAD is disabled. This means that the CAD keystroke will cause a **SIGINT** signal to be sent to init (process 1), whereupon this process may decide upon a proper action (maybe: kill all processes, sync, reboot).

LINUX_REBOOT_CMD_CAD_ON
>   (**RB_ENABLE_CAD**, 0x89abcdef). CAD is enabled. This means that the CAD keystroke will immediately cause the action associated with **LINUX_REBOOT_CMD_RESTART**.

LINUX_REBOOT_CMD_HALT
>   (**RB_HALT_SYSTEM**, 0xcdef0123; since Linux 1.1.76). The message "System halted." is printed, and the system is halted. Control is given to the ROM monitor, if there is one. If not preceded by a **sync**(2), data will be lost.

LINUX_REBOOT_CMD_KEXEC
>   (**RB_KEXEC**, 0x45584543, since Linux 2.6.13). Execute a kernel that has been loaded earlier with **kexec_load**(2). This option is available only if the kernel was configured with **CON-FIG_KEXEC**.

LINUX_REBOOT_CMD_POWER_OFF
>   (**RB_POWER_OFF**, 0x4321fedc; since Linux 2.1.30). The message "Power down." is printed, the system is stopped, and all power is removed from the system, if possible. If not preceded by a **sync**(2), data will be lost.

LINUX_REBOOT_CMD_RESTART
  (**RB_AUTOBOOT**, 0x1234567). The message "Restarting system." is printed, and a default restart is performed immediately. If not preceded by a **sync**(2), data will be lost.

LINUX_REBOOT_CMD_RESTART2
  (0xa1b2c3d4; since Linux 2.1.30). The message "Restarting system with command '%s'" is printed, and a restart (using the command string given in *arg*) is performed immediately. If not preceded by a **sync**(2), data will be lost.

LINUX_REBOOT_CMD_SW_SUSPEND
  (**RB_SW_SUSPEND**, 0xd000fce1; since Linux 2.5.18). The system is suspended (hibernated) to disk. This option is available only if the kernel was configured with **CONFIG_HIBERNATION**.

Only the superuser may call **reboot**().

The precise effect of the above actions depends on the architecture. For the i386 architecture, the additional argument does not do anything at present (2.1.122), but the type of reboot can be determined by kernel command-line arguments ("reboot=...") to be either warm or cold, and either hard or through the BIOS.

Behavior inside PID namespaces
  Since Linux 3.4, if **reboot**() is called from a PID namespace other than the initial PID namespace with one of the *cmd* values listed below, it performs a "reboot" of that namespace: the "init" process of the PID namespace is immediately terminated, with the effects described in **pid_namespaces**(7).

  The values that can be supplied in *cmd* when calling **reboot**() in this case are as follows:

  LINUX_REBOOT_CMD_RESTART, **LINUX_REBOOT_CMD_RESTART2**
    The "init" process is terminated, and **wait**(2) in the parent process reports that the child was killed with a **SIGHUP** signal.

  LINUX_REBOOT_CMD_POWER_OFF, **LINUX_REBOOT_CMD_HALT**
    The "init" process is terminated, and **wait**(2) in the parent process reports that the child was killed with a **SIGINT** signal.

  For the other *cmd* values, **reboot**() returns −1 and *errno* is set to **EINVAL**.

RETURN VALUE
  For the values of *cmd* that stop or restart the system, a successful call to **reboot**() does not return. For the other *cmd* values, zero is returned on success. In all cases, −1 is returned on failure, and *errno* is set to indicate the error.

ERRORS
  EFAULT
    Problem with getting user-space data under **LINUX_REBOOT_CMD_RESTART2**.

  EINVAL
    Bad magic numbers or *cmd*.

  EPERM
    The calling process has insufficient privilege to call **reboot**(); the caller must have the **CAP_SYS_BOOT** inside its user namespace.

STANDARDS
  reboot() is Linux-specific, and should not be used in programs intended to be portable.

SEE ALSO
  systemctl(1), **systemd**(1), **kexec_load**(2), **sync**(2), **bootparam**(7), **capabilities**(7), **ctrlaltdel**(8), **halt**(8), **shutdown**(8)