# **The reboot() system call**

reboot - reboot or enable/disable Ctrl-Alt-Del (also abbreviated

as CAD)

The **reboot**() call reboots the system, or enables/disables the reboot keystroke.

**Syntax:**

**#include<unistd.h>**  
 **#include <linux/reboot.h>**

**int reboot(int**magic**, int**magic2**, int**flag**, void \***arg**);**

Under glibc some of the constants involved have gotten symbolic names RB\_\*, and the library call is a 1-argument wrapper around the 3-argument system call:

**#include<unistd.h>**  
 **#include <sys/reboot.h>**

**int reboot(int***flag***);**

This system call will fail (with EINVAL) unless,

* magic equals LINUX\_REBOOT\_MAGIC1 (that is, 0xfee1dead) and
* magic2 equals LINUX\_REBOOT\_MAGIC2 (that is, 672274793). However, since 2.1.17 also LINUX\_REBOOT\_MAGIC2A (that is, 85072278) and since 2.1.97 also LINUX\_REBOOT\_MAGIC2B (that is, 369367448) and since 2.5.71 also LINUX\_REBOOT\_MAGIC2C (that is, 537993216) are permitted as value for magic2. (The hexadecimal values of these constants are meaningful.)

arg – pointer to additional argument flag

The flag 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)](https://man7.org/linux/man-pages/man2/sync.2.html), 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)](https://man7.org/linux/man-pages/man2/kexec_load.2.html).

This option is available only if the kernel was configured

with **CONFIG\_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)](https://man7.org/linux/man-pages/man2/sync.2.html), 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)](https://man7.org/linux/man-pages/man2/sync.2.html), 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)](https://man7.org/linux/man-pages/man2/sync.2.html), 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 use this function.

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.

**Behaviour 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/flag 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)](https://man7.org/linux/man-pages/man7/pid_namespaces.7.html).

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)](https://man7.org/linux/man-pages/man2/wait.2.html) 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)](https://man7.org/linux/man-pages/man2/wait.2.html) in the

parent process reports that the child was killed with a

**SIGINT** signal.

# **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](https://man7.org/linux/man-pages/man3/errno.3.html) is set to indicate the error.

# **ERRORS**

* **EFAULT** Problem with getting user-space data under

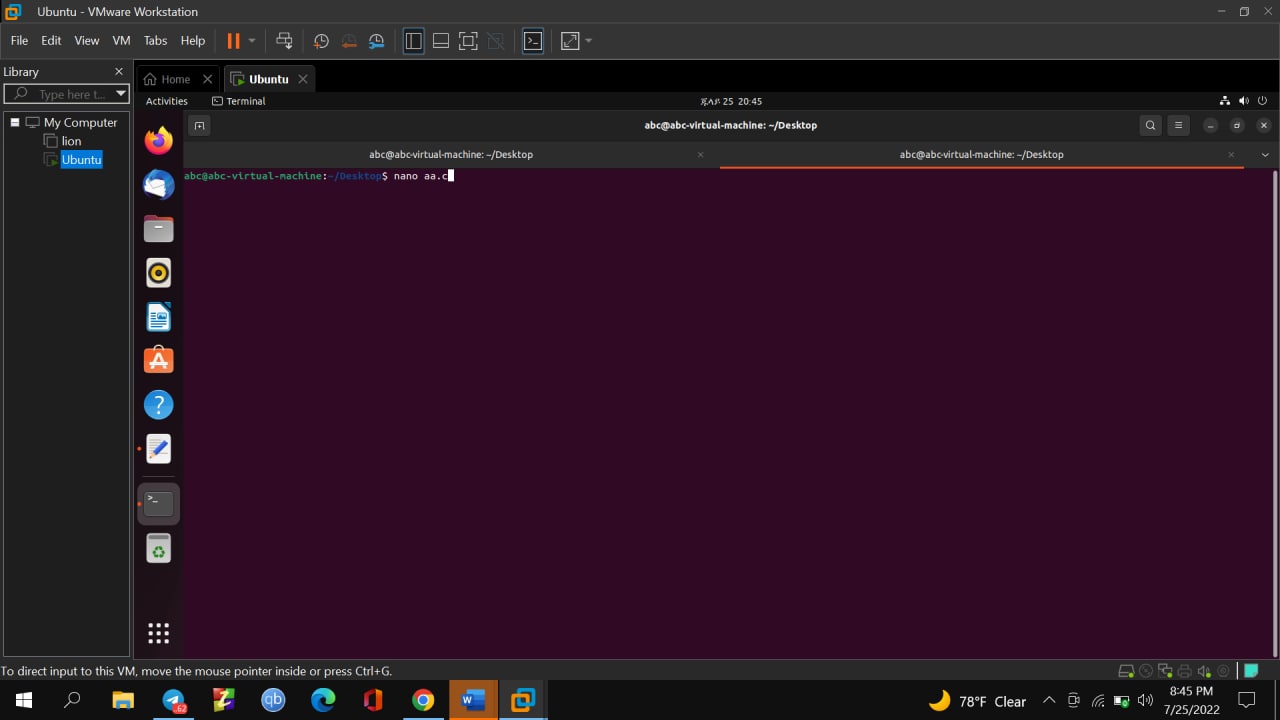
**LINUX\_REBOOT\_CMD\_RESTART2**.

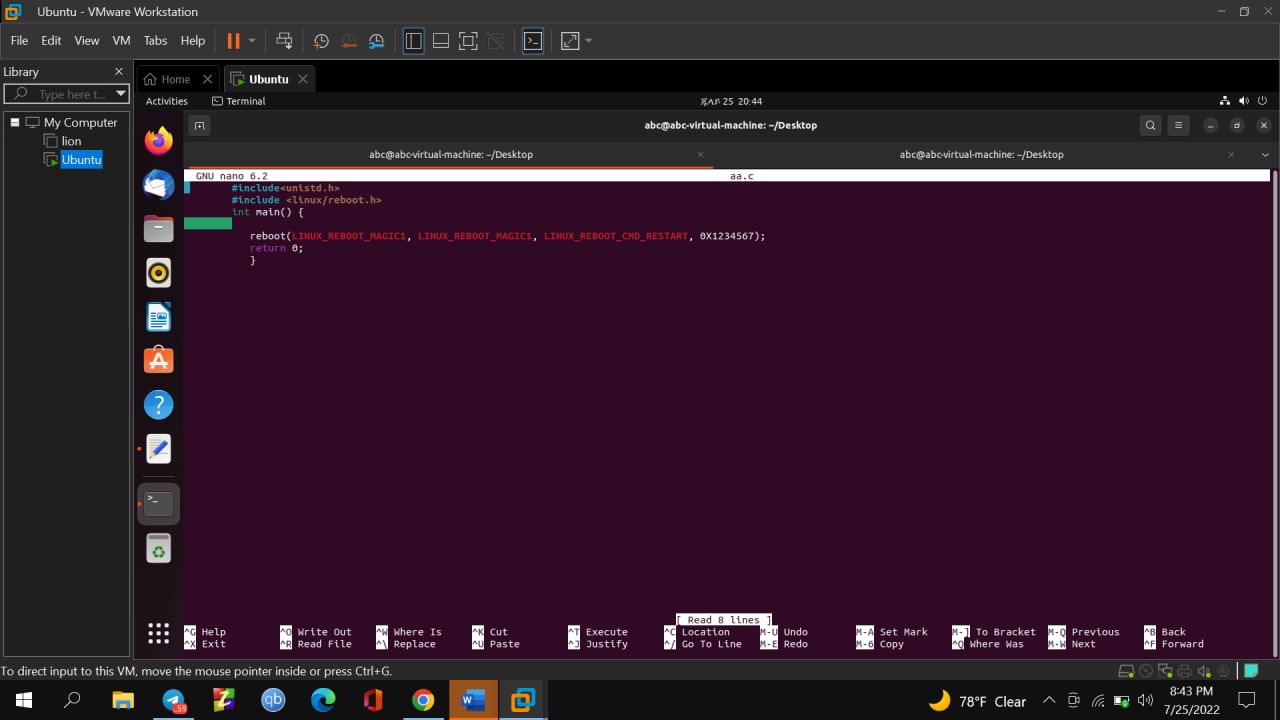
* **EINVAL** Bad magic numbers or cmd.
* **EPERM** The calling process has insufficient privilege to call

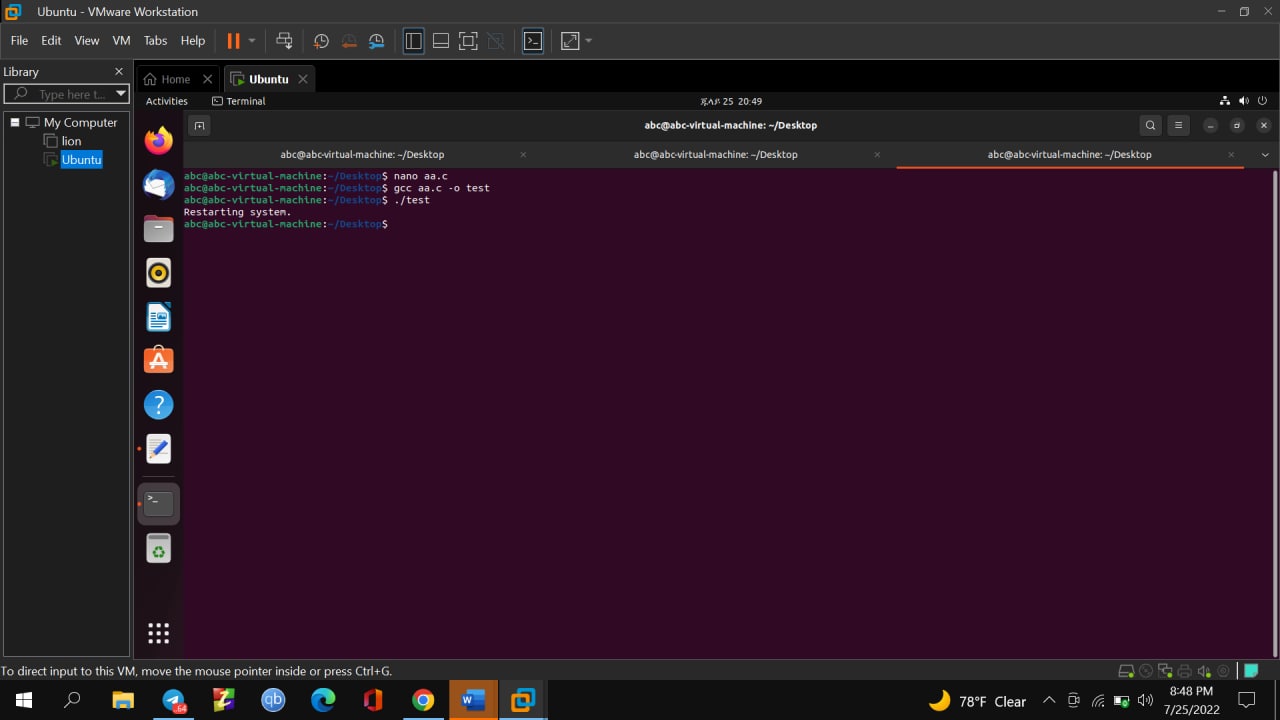
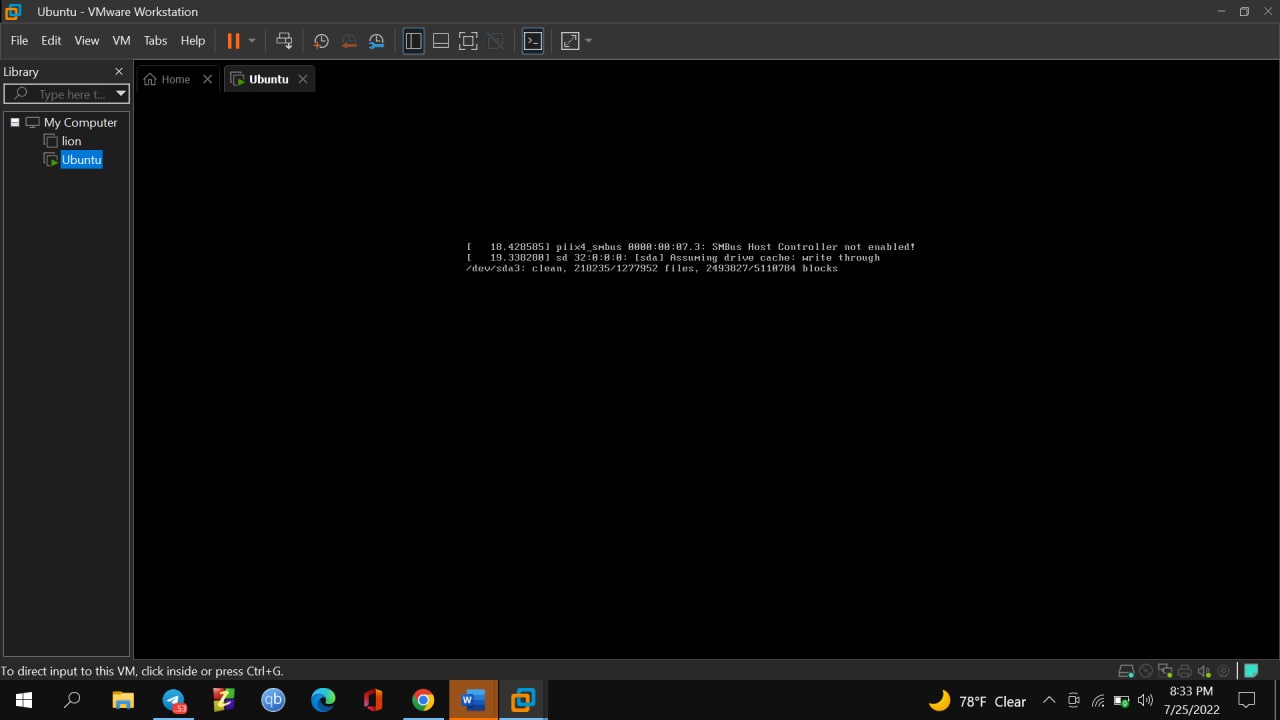
**reboot**(); the caller must have the **CAP\_SYS\_BOOT** inside its

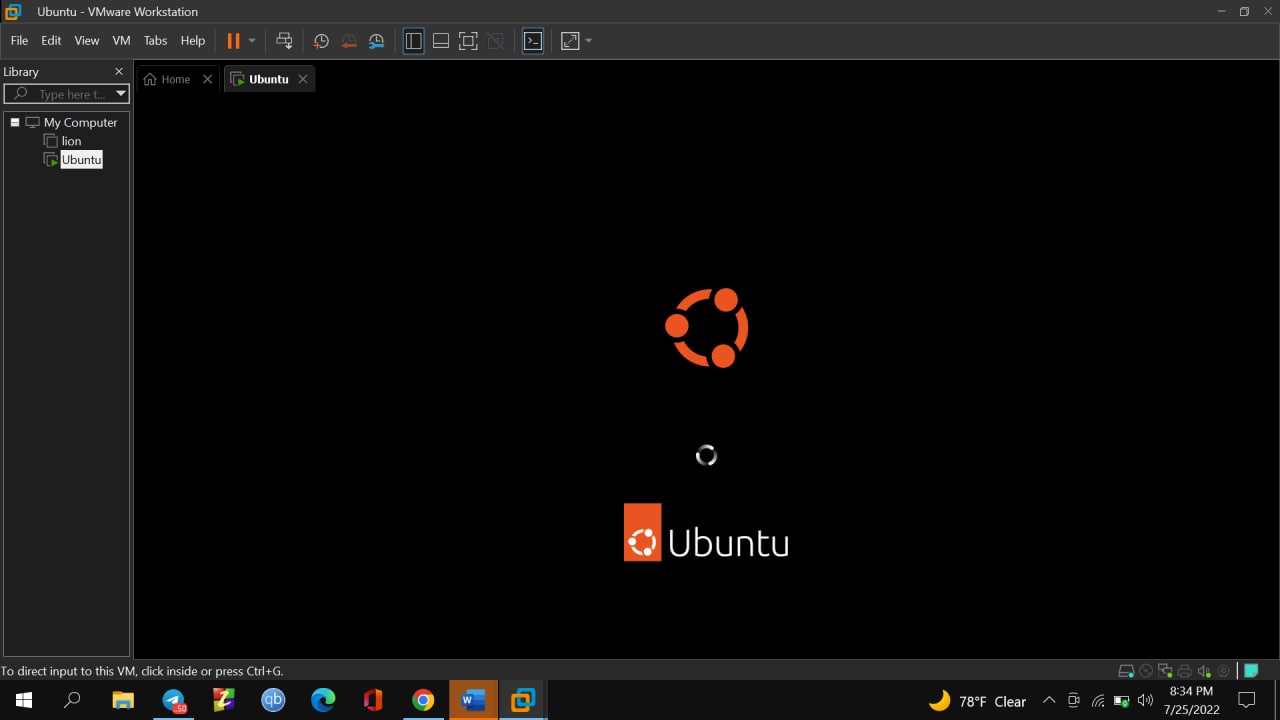
user namespace.

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



****

****

****

The above code is an implementation for one of the flags to restart the system.