NAME
    modprobe − Add and remove modules from the Linux Kernel

SYNOPSIS
    modprobe [−**v**] [−**V**] [−**C** *config−file*] [−**n**] [−**i**] [−**q**] [−**b**] [*modulename*] [*module parameters...*]

    modprobe [−r] [−**v**] [−**n**] [−**i**] [*modulename...*]

    modprobe [−c]

    modprobe [−−dump−modversions] [*filename*]

DESCRIPTION
    modprobe intelligently adds or removes a module from the Linux kernel: note that for convenience, there
    is no difference between _ and − in module names (automatic underscore conversion is performed).
    modprobe looks in the module directory /lib/modules/'uname −r' for all the modules and other files, except
    for the optional configuration files in the /etc/modprobe.d directory (see **modprobe.d**(5)). **modprobe** will
    also use module options specified on the kernel command line in the form of <module>.<option> and
    blacklists in the form of modprobe.blacklist=<module>.

    Note that unlike in 2.4 series Linux kernels (which are not supported by this tool) this version of **modprobe**
    does not do anything to the module itself: the work of resolving symbols and understanding parameters is
    done inside the kernel. So module failure is sometimes accompanied by a kernel message: see **dmesg**(8).

    modprobe expects an up−to−date modules.dep.bin file as generated by the corresponding **depmod** utility
    shipped along with **modprobe** (see **depmod**(8)). This file lists what other modules each module needs (if
    any), and **modprobe** uses this to add or remove these dependencies automatically.

    If any arguments are given after the *modulename*, they are passed to the kernel (in addition to any options
    listed in the configuration file).

OPTIONS
    −a, **−−all**
        Insert all module names on the command line.

    −b, **−−use−blacklist**
        This option causes **modprobe** to apply the **blacklist** commands in the configuration files (if any) to
        module names as well. It is usually used by **udev**(7).

    −C, **−−config**
        This option overrides the default configuration directory (/etc/modprobe.d).

        This option is passed through **install** or **remove** commands to other **modprobe** commands in the
        MODPROBE_OPTIONS environment variable.

    −c, **−−showconfig**
        Dump out the effective configuration from the config directory and exit.

    −−dump−modversions
        Print out a list of module versioning information required by a module. This option is commonly used
        by distributions in order to package up a Linux kernel module using module versioning deps.

    −d, **−−dirname**
        Root directory for modules, / by default.

    −−first−time
        Normally, **modprobe** will succeed (and do nothing) if told to insert a module which is already present
        or to remove a module which isn't present. This is ideal for simple scripts; however, more complicated
        scripts often want to know whether **modprobe** really did something: this option makes modprobe fail
        in the case that it actually didn't do anything.

    −−force−vermagic
        Every module contains a small string containing important information, such as the kernel and
        compiler versions. If a module fails to load and the kernel complains that the "version magic" doesn't

match, you can use this option to remove it. Naturally, this check is there for your protection, so using this option is dangerous unless you know what you're doing.

This applies to any modules inserted: both the module (or alias) on the command line and any modules on which it depends.

−−force−modversion

When modules are compiled with CONFIG_MODVERSIONS set, a section detailing the versions of every interfaced used by (or supplied by) the module is created. If a module fails to load and the kernel complains that the module disagrees about a version of some interface, you can use "−−force−modversion" to remove the version information altogether. Naturally, this check is there for your protection, so using this option is dangerous unless you know what you're doing.

This applies any modules inserted: both the module (or alias) on the command line and any modules on which it depends.

−f, **−−force**

Try to strip any versioning information from the module which might otherwise stop it from loading: this is the same as using both **−−force−vermagic** and **−−force−modversion**. Naturally, these checks are there for your protection, so using this option is dangerous unless you know what you are doing.

This applies to any modules inserted: both the module (or alias) on the command line and any modules it on which it depends.

−i, **−−ignore−install**, **−−ignore−remove**

This option causes **modprobe** to ignore **install** and **remove** commands in the configuration file (if any) for the module specified on the command line (any dependent modules are still subject to commands set for them in the configuration file). Both **install** and **remove** commands will currently be ignored when this option is used regardless of whether the request was more specifically made with only one or other (and not both) of **−−ignore−install** or **−−ignore−remove**. See **modprobe.d**(5).

−n, **−−dry−run**, **−−show**

This option does everything but actually insert or delete the modules (or run the install or remove commands). Combined with **−v**, it is useful for debugging problems. For historical reasons both **−−dry−run** and **−−show** actually mean the same thing and are interchangeable.

−q, **−−quiet**

With this flag, **modprobe** won't print an error message if you try to remove or insert a module it can't find (and isn't an alias or **install/remove** command). However, it will still return with a non−zero exit status. The kernel uses this to opportunistically probe for modules which might exist using request_module.

−R, **−−resolve−alias**

Print all module names matching an alias. This can be useful for debugging module alias problems.

−r, **−−remove**

This option causes **modprobe** to remove rather than insert a module. If the modules it depends on are also unused, **modprobe** will try to remove them too. Unlike insertion, more than one module can be specified on the command line (it does not make sense to specify module parameters when removing modules).

There is usually no reason to remove modules, but some buggy modules require it. Your distribution kernel may not have been built to support removal of modules at all.

−S, **−−set−version**

Set the kernel version, rather than using **uname**(2) to decide on the kernel version (which dictates where to find the modules).

−−show−depends

List the dependencies of a module (or alias), including the module itself. This produces a (possibly

empty) set of module filenames, one per line, each starting with "insmod" and is typically used by distributions to determine which modules to include when generating initrd/initramfs images. **Install** commands which apply are shown prefixed by "install". It does not run any of the install commands. Note that **modinfo**(8) can be used to extract dependencies of a module from the module itself, but knows nothing of aliases or install commands.

−s, **−−syslog**
This option causes any error messages to go through the syslog mechanism (as LOG_DAEMON with level LOG_NOTICE) rather than to standard error. This is also automatically enabled when stderr is unavailable.

This option is passed through **install** or **remove** commands to other **modprobe** commands in the MODPROBE_OPTIONS environment variable.

−V, **−−version**
Show version of program and exit.

−v, **−−verbose**
Print messages about what the program is doing. Usually **modprobe** only prints messages if something goes wrong.

This option is passed through **install** or **remove** commands to other **modprobe** commands in the MODPROBE_OPTIONS environment variable.

ENVIRONMENT
The MODPROBE_OPTIONS environment variable can also be used to pass arguments to **modprobe**.

COPYRIGHT
This manual page originally Copyright 2002, Rusty Russell, IBM Corporation. Maintained by Jon Masters and others.

SEE ALSO
modprobe.d(5), **insmod**(8), **rmmod**(8), **lsmod**(8), **modinfo**(8) **depmod**(8)

AUTHORS
Jon Masters <jcm@jonmasters.org>
Developer

Robby Workman <rworkman@slackware.com>
Developer

Lucas De Marchi <lucas.de.marchi@gmail.com>
Developer