```
Documentation for /proc/sys/kernel/*    kernel version 2.2.10
        (c) 1998, 1999,  Rik van Riel <riel@nl.linux.org>
        (c) 2009,        Shen Feng<shen@cn.fujitsu.com>

For general info and legal blurb, please look in README.

==============================================================

This file contains documentation for the sysctl files in
/proc/sys/kernel/ and is valid for Linux kernel version 2.2.

The files in this directory can be used to tune and monitor
miscellaneous and general things in the operation of the Linux
kernel. Since some of the files _can_ be used to screw up your
system, it is advisable to read both documentation and source
before actually making adjustments.

Currently, these files might (depending on your configuration)
show up in /proc/sys/kernel:

- acct
- acpi_video_flags
- auto_msgmni
- bootloader_type            [ X86 only ]
- bootloader_version         [ X86 only ]
- callhome                   [ S390 only ]
- cap_last_cap
- core_pattern
- core_pipe_limit
- core_uses_pid
- ctrl-alt-del
- dmesg_restrict
- domainname
- hostname
- hotplug
- hung_task_panic
- hung_task_check_count
- hung_task_timeout_secs
- hung_task_warnings
- kexec_load_disabled
- kptr_restrict
- kstack_depth_to_print       [ X86 only ]
- l2cr                        [ PPC only ]
- modprobe                    ==> Documentation/debugging-modules.txt
- modules_disabled
- msg_next_id                 [ sysv ipc ]
- msgmax
- msgmnb
- msgmni
- nmi_watchdog
- osrelease
- ostype
- overflowgid
- overflowuid
- panic
- panic_on_oops
- panic_on_stackoverflow
- panic_on_unrecovered_nmi
- panic_on_warn
- pid_max
- powersave-nap               [ PPC only ]
- printk
- printk_delay
- printk_ratelimit
- printk_ratelimit_burst
- randomize_va_space
- real-root-dev               ==> Documentation/initrd.txt
- reboot-cmd                  [ SPARC only ]
- rtsig-max
- rtsig-nr
- sem
- sem_next_id                 [ sysv ipc ]
- sg-big-buff                 [ generic SCSI device (sg) ]
- shm_next_id                 [ sysv ipc ]
- shm_rmid_forced
- shmall
- shmmax                      [ sysv ipc ]
- shmmni
- softlockup_all_cpu_backtrace
- soft_watchdog
- stop-a                      [ SPARC only ]
- sysrq                       ==> Documentation/sysrq.txt
- sysctl_writes_strict
- tainted
- threads-max
- unknown_nmi_panic
- watchdog
- watchdog_thresh
- version

==============================================================

acct:

highwater lowwater frequency

If BSD-style process accounting is enabled these values control
its behaviour. If free space on filesystem where the log lives
goes below <lowwater>% accounting suspends. If free space gets
above <highwater>% accounting resumes. <Frequency> determines
how often do we check the amount of free space (value is in
seconds). Default:
4 2 30
That is, suspend accounting if there left <= 2% free; resume it
if we got >=4%; consider information about amount of free space
valid for 30 seconds.

==============================================================

acpi_video_flags:

flags

See Doc*/kernel/power/video.txt, it allows mode of video boot to be
set during run time.

==============================================================

auto_msgmni:
```

This variable has no effect and may be removed in future kernel
releases. Reading it always returns 0.
Up to Linux 3.17, it enabled/disabled automatic recomputing of msgmni
upon memory add/remove or upon ipc namespace creation/removal.
Echoing "1" into this file enabled msgmni automatic recomputing.
Echoing "0" turned it off. auto_msgmni default value was 1.

==

bootloader_type:

x86 bootloader identification

This gives the bootloader type number as indicated by the bootloader,
shifted left by 4, and OR'd with the low four bits of the bootloader
version. The reason for this encoding is that this used to match the
type_of_loader field in the kernel header; the encoding is kept for
backwards compatibility. That is, if the full bootloader type number
is 0x15 and the full version number is 0x234, this file will contain
the value 340 = 0x154.

See the type_of_loader and ext_loader_type fields in
Documentation/x86/boot.txt for additional information.

==

bootloader_version:

x86 bootloader version

The complete bootloader version number. In the example above, this
file will contain the value 564 = 0x234.

See the type_of_loader and ext_loader_ver fields in
Documentation/x86/boot.txt for additional information.

==

callhome:

Controls the kernel's callhome behavior in case of a kernel panic.

The s390 hardware allows an operating system to send a notification
to a service organization (callhome) in case of an operating system panic.

When the value in this file is 0 (which is the default behavior)
nothing happens in case of a kernel panic. If this value is set to "1"
the complete kernel oops message is send to the IBM customer service
organization in case the mainframe the Linux operating system is running
on has a service contract with IBM.

==

cap_last_cap

Highest valid capability of the running kernel. Exports
CAP_LAST_CAP from the kernel.

```
==============================================================

core_pattern:

core_pattern is used to specify a core dumpfile pattern name.
. max length 128 characters; default value is "core"
. core_pattern is used as a pattern template for the output filename;
  certain string patterns (beginning with '%') are substituted with
  their actual values.
. backward compatibility with core_uses_pid:
        If core_pattern does not include "%p" (default does not)
        and core_uses_pid is set, then .PID will be appended to
        the filename.
. corename format specifiers:
        %<NUL>  '%' is dropped
        %%      output one '%'
        %p      pid
        %P      global pid (init PID namespace)
        %i      tid
        %I      global tid (init PID namespace)
        %u      uid (in initial user namespace)
        %g      gid (in initial user namespace)
        %d      dump mode, matches PR_SET_DUMPABLE and
                /proc/sys/fs/suid_dumpable
        %s      signal number
        %t      UNIX time of dump
        %h      hostname
        %e      executable filename (may be shortened)
        %E      executable path
        %<OTHER> both are dropped
. If the first character of the pattern is a '|', the kernel will treat
  the rest of the pattern as a command to run.  The core dump will be
  written to the standard input of that program instead of to a file.

==============================================================

core_pipe_limit:

This sysctl is only applicable when core_pattern is configured to pipe
core files to a user space helper (when the first character of
core_pattern is a '|', see above).  When collecting cores via a pipe
to an application, it is occasionally useful for the collecting
application to gather data about the crashing process from its
/proc/pid directory.  In order to do this safely, the kernel must wait
for the collecting process to exit, so as not to remove the crashing
processes proc files prematurely.  This in turn creates the
possibility that a misbehaving userspace collecting process can block
the reaping of a crashed process simply by never exiting.  This sysctl
defends against that.  It defines how many concurrent crashing
processes may be piped to user space applications in parallel.  If
this value is exceeded, then those crashing processes above that value
are noted via the kernel log and their cores are skipped.  0 is a
special value, indicating that unlimited processes may be captured in
parallel, but that no waiting will take place (i.e. the collecting
process is not guaranteed access to /proc/<crashing pid>/).  This
value defaults to 0.

==============================================================

core_uses_pid:
```

```
The default coredump filename is "core".  By setting
core_uses_pid to 1, the coredump filename becomes core.PID.
If core_pattern does not include "%p" (default does not)
and core_uses_pid is set, then .PID will be appended to
the filename.

==============================================================

ctrl-alt-del:

When the value in this file is 0, ctrl-alt-del is trapped and
sent to the init(1) program to handle a graceful restart.
When, however, the value is > 0, Linux's reaction to a Vulcan
Nerve Pinch (tm) will be an immediate reboot, without even
syncing its dirty buffers.

Note: when a program (like dosemu) has the keyboard in 'raw'
mode, the ctrl-alt-del is intercepted by the program before it
ever reaches the kernel tty layer, and it's up to the program
to decide what to do with it.

==============================================================

dmesg_restrict:

This toggle indicates whether unprivileged users are prevented
from using dmesg(8) to view messages from the kernel's log buffer.
When dmesg_restrict is set to (0) there are no restrictions. When
dmesg_restrict is set set to (1), users must have CAP_SYSLOG to use
dmesg(8).

The kernel config option CONFIG_SECURITY_DMESG_RESTRICT sets the
default value of dmesg_restrict.

==============================================================

domainname & hostname:

These files can be used to set the NIS/YP domainname and the
hostname of your box in exactly the same way as the commands
domainname and hostname, i.e.:
# echo "darkstar" > /proc/sys/kernel/hostname
# echo "mydomain" > /proc/sys/kernel/domainname
has the same effect as
# hostname "darkstar"
# domainname "mydomain"

Note, however, that the classic darkstar.frop.org has the
hostname "darkstar" and DNS (Internet Domain Name Server)
domainname "frop.org", not to be confused with the NIS (Network
Information Service) or YP (Yellow Pages) domainname. These two
domain names are in general different. For a detailed discussion
see the hostname(1) man page.

==============================================================

hotplug:

Path for the hotplug policy agent.
```

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.