

NAME

prlimit – get and set process resource limits

SYNOPSIS

prlimit [options] [--resource[=*limits*]] [--pid *PID*]

prlimit [options] [--resource[=*limits*]] *command* [*argument...*]

DESCRIPTION

Given a process ID and one or more resources, **prlimit** tries to retrieve and/or modify the limits.

When *command* is given, **prlimit** will run this command with the given arguments.

The *limits* parameter is composed of a soft and a hard value, separated by a colon (:), in order to modify the existing values. If no *limits* are given, **prlimit** will display the current values. If one of the values is not given, then the existing one will be used. To specify the unlimited or infinity limit (**RLIM_INFINITY**), the `-1` or 'unlimited' string can be passed.

Because of the nature of limits, the soft limit must be lower or equal to the high limit (also called the ceiling). To see all available resource limits, refer to the **RESOURCE OPTIONS** section.

- *soft:_hard_* Specify both limits.
- *soft:* Specify only the soft limit.
- *:hard* Specify only the hard limit.
- *value* Specify both limits to the same value.

GENERAL OPTIONS

--noheadings

Do not print a header line.

-o, --output *list*

Define the output columns to use. If no output arrangement is specified, then a default set is used. Use

--help to get a list of all supported columns.

-p, --pid

Specify the process id; if none is given, the running process will be used.

--raw

Use the raw output format.

--verbose

Verbose mode.

-h, --help

Display help text and exit.

-V, --version

Print version and exit.

RESOURCE OPTIONS

-c, --core[=*limits*]

Maximum size of a core file.

-d, --data[=*limits*]

Maximum data size.

- e, --nice[=*limits*]**
Maximum nice priority allowed to raise.
- f, --fsize[=*limits*]**
Maximum file size.
- i, --sigpending[=*limits*]**
Maximum number of pending signals.
- l, --memlock[=*limits*]**
Maximum locked-in-memory address space.
- m, --rss[=*limits*]**
Maximum Resident Set Size (RSS).
- n, --nofile[=*limits*]**
Maximum number of open files.
- q, --msgqueue[=*limits*]**
Maximum number of bytes in POSIX message queues.
- r, --rtprio[=*limits*]**
Maximum real-time priority.
- s, --stack[=*limits*]**
Maximum size of the stack.
- t, --cpu[=*limits*]**
CPU time, in seconds.
- u, --nproc[=*limits*]**
Maximum number of processes.
- v, --as[=*limits*]**
Address space limit.
- x, --locks[=*limits*]**
Maximum number of file locks held.
- y, --rttime[=*limits*]**
Timeout for real-time tasks.

NOTES

The **prlimit(2)** system call is supported since Linux 2.6.36, older kernels will break this program.

EXAMPLES

prlimit --pid 13134

Display limit values for all current resources.

prlimit --pid 13134 --rss --nofile=1024:4095

Display the limits of the RSS, and set the soft and hard limits for the number of open files to 1024 and 4095, respectively.

prlimit --pid 13134 --nproc=512:

Modify only the soft limit for the number of processes.

prlimit --pid \$\$ --nproc=unlimited

Set for the current process both the soft and ceiling values for the number of processes to unlimited.

prlimit --cpu=10 sort -u hugefile

Set both the soft and hard CPU time limit to ten seconds and run **sort(1)**.

AUTHORS

[Davidlohr Bueso](#) <dave@gnu.org> – In memory of Dennis M. Ritchie.

SEE ALSO

ulimit(1p), **prlimit(2)**

REPORTING BUGS

For bug reports, use the issue tracker at <https://github.com/util-linux/util-linux/issues>.

AVAILABILITY

The **prlimit** command is part of the util-linux package which can be downloaded from [Linux Kernel Archive](#) <<https://www.kernel.org/pub/linux/utils/util-linux/>>.