

NAME

ionice – set or get process I/O scheduling class and priority

SYNOPSIS

```
ionice [-c class] [-n level] [-t] -p PID...
ionice [-c class] [-n level] [-t] -P PGID...
ionice [-c class] [-n level] [-t] -u UID...
ionice [-c class] [-n level] [-t] command [argument...]
```

DESCRIPTION

This program sets or gets the I/O scheduling class and priority for a program. If no arguments or just **-p** is given, **ionice** will query the current I/O scheduling class and priority for that process.

When *command* is given, **ionice** will run this command with the given arguments. If no *class* is specified, then *command* will be executed with the "best-effort" scheduling class. The default priority level is 4.

As of this writing, a process can be in one of three scheduling classes:

Idle A program running with idle I/O priority will only get disk time when no other program has asked for disk I/O for a defined grace period. The impact of an idle I/O process on normal system activity should be zero. This scheduling class does not take a priority argument. Presently, this scheduling class is permitted for an ordinary user (since kernel 2.6.25).

Best-effort

This is the effective scheduling class for any process that has not asked for a specific I/O priority. This class takes a priority argument from 0-7, with a lower number being higher priority. Programs running at the same best-effort priority are served in a round-robin fashion.

Note that before kernel 2.6.26 a process that has not asked for an I/O priority formally uses "**none**" as scheduling class, but the I/O scheduler will treat such processes as if it were in the best-effort class. The priority within the best-effort class will be dynamically derived from the CPU nice level of the process: $io_priority = (cpu_nice + 20) / 5$.

For kernels after 2.6.26 with the CFQ I/O scheduler, a process that has not asked for an I/O priority inherits its CPU scheduling class. The I/O priority is derived from the CPU nice level of the process (same as before kernel 2.6.26).

Realtime

The RT scheduling class is given first access to the disk, regardless of what else is going on in the system. Thus the RT class needs to be used with some care, as it can starve other processes. As with the best-effort class, 8 priority levels are defined denoting how big a time slice a given process will receive on each scheduling window. This scheduling class is not permitted for an ordinary (i.e., non-root) user.

OPTIONS

- c, --class *class***
Specify the name or number of the scheduling class to use; 0 for none, 1 for realtime, 2 for best-effort, 3 for idle.
- n, --classdata *level***
Specify the scheduling class data. This only has an effect if the class accepts an argument. For realtime and best-effort, 0-7 are valid data (priority levels), and 0 represents the highest priority level.
- p, --pid *PID*...**
Specify the process IDs of running processes for which to get or set the scheduling parameters.

- P, --pgid *PGID*...**
Specify the process group IDs of running processes for which to get or set the scheduling parameters.
- t, --ignore**
Ignore failure to set the requested priority. If *command* was specified, run it even in case it was not possible to set the desired scheduling priority, which can happen due to insufficient privileges or an old kernel version.
- h, --help**
Display help text and exit.
- u, --uid *UID*...**
Specify the user IDs of running processes for which to get or set the scheduling parameters.
- V, --version**
Display version information and exit.

EXAMPLES

```
# ionice -c 3 -p 89
```

Sets process with PID 89 as an idle I/O process.

```
# ionice -c 2 -n 0 bash
```

Runs 'bash' as a best-effort program with highest priority.

```
# ionice -p 89 91
```

Prints the class and priority of the processes with PID 89 and 91.

NOTES

Linux supports I/O scheduling priorities and classes since 2.6.13 with the CFQ I/O scheduler.

AUTHORS

Jens Axboe <jens@axboe.dk>

Karel Zak <kzak@redhat.com>

SEE ALSO

ioprio_set(2)

AVAILABILITY

The `ionice` command is part of the `util-linux` package and is available from <https://www.kernel.org/pub/linux/utils/util-linux/>.