



Linux Ubuntu 26.04 Manual Pages on command 'perf-lock.1'

\$ man perf-lock.1

PERF-LOCK(1) perf Manual PERF-LOCK(1)

NAME

perf-lock - Analyze lock events

SYNOPSIS

perf lock {record|report|script|info|contention}

DESCRIPTION

You can analyze various lock behaviours and statistics with this perf lock command.

'perf lock record <command>' records lock events between start and end <command>. And this command produces the file "perf.data" which contains tracing results of lock events.

'perf lock report' reports statistical data.

'perf lock script' shows raw lock events.

'perf lock info' shows metadata like threads or addresses of lock instances.

'perf lock contention' shows contention statistics.

COMMON OPTIONS

-i, --input=<file>

Input file name. (default: perf.data unless stdin is a fifo)

--output=<file>

Output file name for perf lock contention and report.

-v, --verbose

Be more verbose (show symbol address, etc).

-q, --quiet

Do not show any warnings or messages. (Suppress -v)

-D, --dump-raw-trace

Dump raw trace in ASCII.

-f, --force

Don't complain, do it.

--vmlinux=<file>

vmlinux pathname

--kallsyms=<file>

kallsyms pathname

REPORT OPTIONS

-k, --key=<value>

Sorting key. Possible values: acquired (default), contended, avg_wait, wait_total, wait_max, wait_min.

-F, --field=<value>

Output fields. By default it shows all the fields but users can customize that using this. Possible values: acquired, contended, avg_wait, wait_total, wait_max, wait_min.

-c, --combine-locks

Merge lock instances in the same class (based on name).

-t, --threads

The -t option is to show per-thread lock stat like below:

```
$ perf lock report -t -F acquired,contended,avg_wait
```

Name	acquired	contended	avg wait (ns)
------	----------	-----------	---------------

perf	240569	9	5784
swapper	106610	19	543
:15789	17370	2	14538
ContainerMgr	8981	6	874
sleep	5275	1	11281
ContainerThread	4416	4	944
RootPressureThr	3215	5	1215
rcu_preempt	2954	0	0
ContainerMgr	2560	0	0
unnamed	1873	0	0

EventManager_De	1845	1	636
-----------------	------	---	-----

futex-default-S	1609	0	0
-----------------	------	---	---

-E, --entries=<value>

Display this many entries.

INFO OPTIONS

-t, --threads

dump only the thread list in perf.data

-m, --map

dump only the map of lock instances (address:name table)

CONTENTION OPTIONS

-k, --key=<value>

Sorting key. Possible values: contended, wait_total (default), wait_max, wait_min, avg_wait.

-F, --field=<value>

Output fields. By default it shows all but the wait_min fields and users can customize that using this. Possible values: contended, wait_total, wait_max, wait_min, avg_wait.

-t, --threads

Show per-thread lock contention stat

-b, --use-bpf

Use BPF program to collect lock contention stats instead of using the input data.

-a, --all-cpus

System-wide collection from all CPUs.

-C, --cpu=<value>

Collect samples only on the list of CPUs provided. Multiple CPUs can be provided as a comma-separated list with no space: 0,1. Ranges of CPUs are specified with -: 0-2.

Default is to monitor all CPUs.

-p, --pid=<value>

Record events on existing process ID (comma separated list).

--tid=<value>

Record events on existing thread ID (comma separated list).

-M, --map-nr-entries=<value>

Maximum number of BPF map entries (default: 16384). This will be aligned to a power of

2.

`--max-stack=<value>`

Maximum stack depth when collecting lock contention (default: 8).

`--stack-skip=<value>`

Number of stack depth to skip when finding a lock caller (default: 3).

`-E, --entries=<value>`

Display this many entries.

`-I, --lock-addr`

Show lock contention stat by address

`-o, --lock-owner`

Show lock contention stat by owners. This option can be combined with `-t`, which shows owner's per thread lock stats, or `-v`, which shows owner's stacktrace. Requires `--use-bpf`.

`-Y, --type-filter=<value>`

Show lock contention only for given lock types (comma separated list). Available values are: semaphore, spinlock, rwlock, rwlock:R, rwlock:W, rwsem, rwsem:R, rwsem:W, rtmutex, rwlock-rt, rwlock-rt:R, rwlock-rt:W, percpu-rwmem, pcpu-sem, pcpu-sem:R, pcpu-sem:W, mutex

Note that RW-variant of locks have :R and :W suffix. Names without the suffix are shortcuts for the both variants. Ex) rwsem = rwsem:R + rwsem:W.

`-L, --lock-filter=<value>`

Show lock contention only for given lock addresses or names (comma separated list).

`-S, --callstack-filter=<value>`

Show lock contention only if the callstack contains the given string. Note that it matches the substring so `rq` would match both `raw_spin_rq_lock` and `irq_enter_rcu`.

`-x, --field-separator=<SEP>`

Show results using a CSV-style output to make it easy to import directly into spreadsheets. Columns are separated by the string specified in `SEP`.

`--lock-cgroup`

Show lock contention stat by cgroup. Requires `--use-bpf`.

`-G, --cgroup-filter=<value>`

Show lock contention only in the given cgroups (comma separated list).

`-J, --inject-delay=<time@function[1]>`

Add delays to the given lock. It's added to the contention-end part so that the (new)

owner of the lock will be delayed. But by slowing down the owner, the waiters will also be delayed as well. This is working only with -b/--use-bpf.

The 'time' is specified in nsec but it can have a unit suffix. Available units are "ms", "us" and "ns". Currently it accepts up to 10ms of delays for safety reasons.

Note that it will busy-wait after it gets the lock. Delaying locks can have significant consequences including potential kernel crashes. Please use it at your own risk.

SEE ALSO

[perf\(1\)](#)

NOTES

1. [time@function](#)

[mailto:time@function](#)

perf

05/25/2026

PERF-LOCK(1)