



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

\$ man perf-kwork.1

PERF-KWORK(1) perf Manual PERF-KWORK(1)

NAME

perf-kwork - Tool to trace/measure kernel work properties (latencies)

SYNOPSIS

perf kwork {record|report|latency|timehist|top}

DESCRIPTION

There are several variants of perf kwork:

'perf kwork record <command>' to record the kernel work

of an arbitrary workload.

'perf kwork report' to report the per kwork runtime.

'perf kwork latency' to report the per kwork latencies.

'perf kwork timehist' provides an analysis of kernel work events.

'perf kwork top' to report the task cpu usage.

Example usage:

perf kwork record -- sleep 1

perf kwork report

perf kwork report -b

perf kwork latency

perf kwork latency -b

perf kwork timehist

perf kwork top

perf kwork top -b

By default it shows the individual work events such as irq, workqueue,

including the run time and delay (time between raise and actually entry):

Runtime start	Runtime end	Cpu	Kwork name	Runtime	Delaytime
		(TYPE)NAME:NUM	(msec)	(msec)	

1811186.976062	1811186.976327	[0000] (s)RCU:9		0.266	0.114
1811186.978452	1811186.978547	[0000] (s)SCHED:7		0.095	0.171
1811186.980327	1811186.980490	[0000] (s)SCHED:7		0.162	0.083
1811186.981221	1811186.981271	[0000] (s)SCHED:7		0.050	0.077
1811186.984267	1811186.984318	[0000] (s)SCHED:7		0.051	0.075
1811186.987252	1811186.987315	[0000] (s)SCHED:7		0.063	0.081
1811186.987785	1811186.987843	[0006] (s)RCU:9		0.058	0.645
1811186.988319	1811186.988383	[0000] (s)SCHED:7		0.064	0.143
1811186.989404	1811186.989607	[0002] (s)TIMER:1		0.203	0.111
1811186.989660	1811186.989732	[0002] (s)SCHED:7		0.072	0.310
1811186.991295	1811186.991407	[0002] eth0:10		0.112	
1811186.991639	1811186.991734	[0002] (s)NET_RX:3		0.095	0.277
1811186.989860	1811186.991826	[0002] (w)vmstat_shepherd		1.966	0.345

...

Times are in msec.usec.

OPTIONS

-D, --dump-raw-trace=

Display verbose dump of the sched data.

-f, --force

Don?t complain, do it.

-k, --kwork

List of kwork to profile (irq, softirq, workqueue, sched, etc)

-v, --verbose

Be more verbose. (show symbol address, etc)

OPTIONS for perf kwork report

.ft C

-b::

--use-bpf::

Use BPF to measure kwork runtime

-C::

--cpu::

Only show events for the given CPU(s) (comma separated list).

-i::

--input::

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

-n::

--name::

Only show events for the given name.

-s::

--sort::

Sort by key(s): runtime, max, count

-S::

--with-summary::

Show summary with statistics

--time::

Only analyze samples within given time window: <start>,<stop>. Times have the format seconds.microseconds. If start is not given (i.e., time string is 'x.y') then analysis starts at the beginning of the file. If stop time is not given (i.e, time string is 'x.y,') then analysis goes to end of file.

OPTIONS for 'perf kwork latency'

.ft

-b, --use-bpf

Use BPF to measure kwork latency

-C, --cpu

Only show events for the given CPU(s) (comma separated list).

-i, --input

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

-n, --name

Only show events for the given name.

-s, --sort

Sort by key(s): avg, max, count

--time

Only analyze samples within given time window: <start>,<stop>. Times have the format seconds.microseconds. If start is not given (i.e., time string is ,x.y) then analysis starts at the beginning of the file. If stop time is not given (i.e, time string is x.y,) then analysis goes to end of file.

OPTIONS FOR PERF KWORK TIMEHIST

-C, --cpu

Only show events for the given CPU(s) (comma separated list).

-g, --call-graph

Display call chains if present (default off).

-i, --input

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

-k, --vmlinux=<file>

Vmlinux pathname

-n, --name

Only show events for the given name.

--kallsyms=<file>

Kallsyms pathname

--max-stack

Maximum number of functions to display in backtrace, default 5.

--symfs=<directory>

Look for files with symbols relative to this directory.

--time

Only analyze samples within given time window: <start>,<stop>. Times have the format seconds.microseconds. If start is not given (i.e., time string is ,x.y) then analysis starts at the beginning of the file. If stop time is not given (i.e, time string is x.y,) then analysis goes to end of file.

OPTIONS for perf kwork top

.ft C

-b::

--use-bpf::

Use BPF to measure task cpu usage.

-C::

--cpu::

Only show events for the given CPU(s) (comma separated list).

-i::

--input::

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

-n::

--name::

Only show events for the given name.

-s::

--sort::

Sort by key(s): rate, runtime, tid

--time::

Only analyze samples within given time window: <start>,<stop>. Times have the format seconds.microseconds. If start is not given (i.e., time string is 'x.y') then analysis starts at the beginning of the file. If stop time is not given (i.e, time string is 'x.y,') then analysis goes to end of file.

SEE ALSO

.ft

perf-record(1)

perf

05/25/2026

PERF-KWORK(1)