Rocky Enterprise Linux 9.2 Manual Pages on command 'bpftool-perf.8'

$ man bpftool-perf.8

```
BPFTOOL-PERF(8)                                  BPFTOOL-PERF(8)

NAME

    bpftool-perf - tool for inspection of perf related bpf prog attachments

SYNOPSIS

      bpftool [OPTIONS] perf COMMAND

      OPTIONS  := { { -j | --json } [{ -p | --pretty }] | { -d | --debug }

      | { -l | --legacy } }

      COMMANDS := { show | list | help }

PERF COMMANDS

    bpftool perf { show | list }

    bpftool perf help

DESCRIPTION

      bpftool perf { show | list }

          List all raw_tracepoint, tracepoint, kprobe attachment in the

          system.

          Output will start with process id and file descriptor in that

          process, followed by bpf program id, attachment  information,

          and  attachment  point.   The attachment point for raw_trace?
```

point/tracepoint is the trace probe name. The attachment point for k[ret]probe is either symbol name and offset, or a kernel virtual address. The attachment point for u[ret]probe is the file name and the file offset.

bpftool perf help

Print short help message.

OPTIONS

-h, --help

Print short help message (similar to bpftool help).

-V, --version

Print bpftool's version number (similar to bpftool version), the number of the libbpf version in use, and optional fea? tures that were included when bpftool was compiled. Optional features include linking against libbfd to provide the disas? sembler for JIT-ted programs (bpftool prog dump jited) and usage of BPF skeletons (some features like bpftool prog pro? file or showing pids associated to BPF objects may rely on it).

-j, --json

Generate JSON output. For commands that cannot produce JSON, this option has no effect.

-p, --pretty

Generate human-readable JSON output. Implies -j.

-d, --debug

Print all logs available, even debug-level information. This includes logs from libbpf as well as from the verifier, when attempting to load programs.

-l, --legacy

Use legacy libbpf mode which has more relaxed BPF program re? quirements. By default, bpftool has more strict requirements about section names, changes pinning logic and doesn't sup? port some of the older non-BTF map declarations.

See

https://github.com/libbpf/libbpf/wiki/Libbpf:-the-road-to-v1.0

for details.

EXAMPLES

```
# bpftool perf
   pid 21711  fd 5: prog_id 5  kprobe  func __x64_sys_write  offset 0
   pid 21765  fd 5: prog_id 7  kretprobe  func __x64_sys_nanosleep  offset 0
   pid 21767  fd 5: prog_id 8  tracepoint  sys_enter_nanosleep
   pid 21800  fd 5: prog_id 9  uprobe  filename /home/yhs/a.out  offset 1159
# bpftool -j perf
   [{"pid":21711,"fd":5,"prog_id":5,"fd_type":"kprobe","func":"__x64_sys_write","offset":0}, \
    {"pid":21765,"fd":5,"prog_id":7,"fd_type":"kretprobe","func":"__x64_sys_nanosleep","offset":0}, \
    {"pid":21767,"fd":5,"prog_id":8,"fd_type":"tracepoint","tracepoint":"sys_enter_nanosleep"}, \
    {"pid":21800,"fd":5,"prog_id":9,"fd_type":"uprobe","filename":"/home/yhs/a.out","offset":1159}]
```

SEE ALSO

bpf(2), bpf-helpers(7), bpftool(8), bpftool-btf(8),
bpftool-cgroup(8), bpftool-feature(8), bpftool-gen(8),
bpftool-iter(8), bpftool-link(8), bpftool-map(8), bpftool-net(8),
bpftool-prog(8), bpftool-struct_ops(8)