



Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'bpftool.8' command

\$ man bpftool.8

BPFTOOL(8)

BPFTOOL(8)

NAME

BPFTOOL - tool for inspection and simple manipulation of eBPF programs
and maps

SYNOPSIS

bpftool [OPTIONS] OBJECT { COMMAND | help }

bpftool batch file FILE

bpftool version

OBJECT := { map | program | link | cgroup | perf | net | feature |

btf | gen | struct_ops | iter }

OPTIONS := { { -V | --version } | { -j | --json } [{ -p | --pretty

}] | { -d | --debug } | { -l | --legacy } }

MAP-COMMANDS := { show | list | create | dump | update | lookup |

getnext | delete | pin | event_pipe | help }

PROG-COMMANDS := { show | list | dump jited | dump xlated | pin |

load | attach | detach | help }

LINK-COMMANDS := { show | list | pin | detach | help }

CGROUP-COMMANDS := { show | list | attach | detach | help }

PERF-COMMANDS := { show | list | help }

NET-COMMANDS := { show | list | help }

FEATURE-COMMANDS := { probe | help }

BTF-COMMANDS := { show | list | dump | help }

GEN-COMMANDS := { object | skeleton | min_core_btf | help }

```
STRUCT-OPS-COMMANDS := { show | list | dump | register | unregister  
| help }
```

```
ITER-COMMANDS := { pin | help }
```

DESCRIPTION

bpftool allows for inspection and simple modification of BPF objects on the system.

Note that format of the output of all tools is not guaranteed to be stable and should not be depended upon.

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 features that were included when bpftool was compiled. Optional features include linking against libbfd to provide the disassembler for JIT-ted programs (`bpftool prog dump jited`) and usage of BPF skeletons (some features like `bpftool prog profile` 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 requirements. 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.

-m, --mapcompat

Allow loading maps with unknown map definitions.

-n, --nomount

Do not automatically attempt to mount any virtual file system

(such as tracefs or BPF virtual file system) when necessary.

SEE ALSO

bpf(2), bpf-helpers(7), 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-perf(8),
bpftool-prog(8), bpftool-struct_ops(8)

BPFTOOL(8)