eBPF Offload Starting Guide

STEP 1 - UPGRADE YOUR KERNEL

Because of the tight integration with Linux networking stack to use the eBPF offload you need to install a very recent version of Linux kernel.

The minimal requirement is Linux 4.9-rc1 (to be released on Oct 17th 2016). Most Linux distributions have vanilla kernel builds available in a special repo:

Fedora: https://fedoraproject.org/wiki/Kernel_Vanilla_Repositories

Ubuntu: https://wiki.ubuntu.com/Kernel/MainlineBuilds

Although to ensure all up to date features are available, we recommend building the kernel directly from the networking -next tree source. The git repo is available at:

Browse link: http://git.kernel.org/cgit/linux/kernel/git/davem/net-next.git/

Clone links: https://git.kernel.org/pub/scm/linux/kernel/git/davem/net-next.git

git://git.kernel.org/pub/scm/linux/kernel/git/davem/net-next.git

Refer to the guides for your distribution on how to build a kernel from sources:

Fedora: https://fedoraproject.org/wiki/Building_a_custom_kernel

Ubuntu: https://wiki.ubuntu.com/KernelTeam/GitKernelBuild

Although distributions often do sometimes overcomplicate things. If you are confident enough following the most basic build/installation guides may prove more useful:

https://kernelnewbies.org/KernelBuild

Please feel free to ask questions on the open-nfp Google Group if you have problems.

STEP 2 - INSTALL THE FIRMWARE

For the eBPF capable firmware file please email: nick.viljoen@netronome.com

Please place the firmware file in /lib/firmware/netronome/nfp6000_net.cat.

STEP 3 - INSTALL THE DRIVERS

You can use the the Linux upstream drivers for the NFP or get the lastest code from our GitHub:

https://github.com/Netronome/nfp-drv-kmods/

After the driver is loaded you should see a line in your kernel logs a line which lists all capabilities of the firmware:

```
nfp_net 0000:02:00.0 eth7: CAP: 0xb5606ff PROMISC L2BCFILT L2MCFILT RXCSUM TXCSUM RXVLAN
TXVLAN GATHER TSO RSS L2SWITCH AUTOMASK IRQMOD VXLAN NVGRE BPF
```

Make sure that BPF is listed here.

STEP 4 - UPDATE USER SPACE TOOLS

Check if your version of ethtool support setting enabling TC offloads:

```
ethtool -k <ifc> | grep hw-tc-offload
```

If nothing is printed you need to update ethtool to a recent version.

Now all you have to do is enable hardware offloads on the interface:

```
ethtool -K <ifc> hw-tc-offload on
```

If you now load programs to TC ingress (or XDP) the hardware offload will be attempted.

Please note the these instruction are highly initial-this process will be changing significantly and being updated as this is a project that is still in its infancy.

Netronome Resources

These will give the reader some understanding of the internals and also the usage of the Netronome eBPF solution.

Jakub Kicinski and Dinan Gunawardena: P4, eBPF and Linux TC Offload

Video: https://attendee.gotowebinar.com/recording/204752531684872963

Slides: http://open-nfp.org/media/pdfs/Open_NFP_P4_EBPF_Linux_TC_Offload_FINAL.pdf

Netdev will go here

Useful eBPF resources

There are a wide variety of useful eBPF resources online, some relating to the architecture (such as the documentation), others relating to the hooks (TC or XDP are the most relevant hooks for offloadable use cases), eBPF and P4 are also covered as well as usecases.

The Documentation: https://www.kernel.org/doc/Documentation/networking/filter.txt

Summarised Documentation: https://github.com/iovisor/bpf-docs/blob/master/eBPF.md

General Information on eBPF

These documents should give the reader a good overview of the premises of eBPF

Jonathan Corbet: Extending extended BPF

LWN Article: https://lwn.net/Articles/603983/

Brendan Gregg: Using Linux BPF Superpowers

Video: https://atscaleconference.com/videos/linux-4-x-performance-using-bpf-superpowers/

Slides: http://fr.slideshare.net/brendangregg/linux-bpf-superpowers

Brendan Blanco: IOVisor@SCALE14x

Slides: https://www.socallinuxexpo.org/sites/default/files/presentations/Room%20211%20-%20IOVisor%20-%20SCaLE%2014x.pdf

Daniel Borkmann: On getting tc classifier fully programmable with cls_bpf

Video: https://www.youtube.com/watch?v=KHXxSN5vwHY

Slides: http://www.netdevconf.org/1.1/proceedings/slides/borkmann-tc-classifier-cls-bpf.pdf

Paper: http://www.netdevconf.org/1.1/proceedings/papers/On-getting-tc-classifier-fully-programmable-with-cls-bpf.pdf

Traffic Control (TC)

The traditional low level kernel hook to which eBPF can be attached

Martin Brown: Traffic Control HOWTO (2006)

Documentation: http://linux-ip.net/articles/Traffic-Control-HOWTO/

Jamal Hadi Salim: Linux Traffic Control Classifier-Action Subsystem Architecture

Video: https://www.youtube.com/watch?v=cyeJYjZHv5M

Slides: http://people.netfilter.org/pablo/netdev0.1/slides/hadi-salim-TC-act-arch-slides.pdf

Paper: https://people.netfilter.org/pablo/netdev0.1/papers/Linux-Traffic-Control-Classifier-Action-Subsystem-Architecture.pdf

Werner Almesberger: Linux Network Traffic Control- Implementation Overview (1999-OLD)

Paper: https://www.almesberger.net/cv/papers/tcio8.pdf

eXpress Data Path (XDP)

A new high performance hook from Tom Herbert and Alexei Starovoitov which sits in driver space

IoVisor XDP Overview: https://www.iovisor.org/technology/xdp

Tom Herbert and Alexei Starovoitov:Initial Presentation about the eXpress Data Path

Slides: https://github.com/iovisor/bpf-docs/raw/master/Express_Data_Path.pdf

Jesper Dangaard Brouer: XDP Documentation

Documentation: https://prototype-kernel.readthedocs.io/en/latest/networking/XDP/index.html

eBPF and P4

Example of how to compile P4 to eBPF

John Fastabend: P4 on the Edge

Slides: https://schd.ws/hosted_files/2016p4workshop/1d/Intel%20Fastabend-P4%20on%20the%20Edge.pdf

Audio: https://ovsorbit.benpfaff.org/#e11

eBPF Application examples

Use case examples from Cisco, Facebook and academia

Cilium-Container Networking: Thomas Graf

Slides: http://www.slideshare.net/ThomasGraf5/cilium-fast-ipv6-container-networking-with-bpf-and-xdp

Audio: https://ovsorbit.benpfaff.org

ILA-Network Address translation: Tom Herbert (the TC and XDP based eBPF patches for this have been posted by Alexei on Netdev the mailing list)

Documentation: https://tools.ietf.org/html/draft-herbert-nvo3-ila-00

Slides: https://www.ietf.org/proceedings/92/slides/slides-92-nvo3-1.pdf

Patches: https://www.spinics.net/lists/netdev/msg397014.html

InKev: In-Kernel Distributed Network Virtualization for DCN (Sigcomm 2016)

Paper: https://ccronline.sigcomm.org/wp-content/uploads/2016/07/sigcomm-ccr-paper4.pdf