Userspace OVS with HW Offload and AF_XDP

Friday, 28 August 2020 09:00 (45 minutes)

OVS has two major datapaths: 1) the Linux kernel datapath, which shipped with Linux distributions and 2) the userspace datapath, which usually coupled with DPDK library as packet I/O interface, and called OVS-DPDK. Recent OVS also supports two offload mechanisms: the TC-flower for the kernel datapath, and the DPDK rte_flow for the userspace datapath. The tc-flower API with kernel datapath seems to be more feature-rich, with the support for connection tracking. However, the userspace datapath is in general faster than the kernel datapath, due to more packet processing optimizations.

With the introduction of AF_XDP to OVS, the userspace datapath can process packets at high rate without requiring DPDK library. AF_XDP socket creates a fast packet channel to the OVS userspace datapath and shows similar performance compared to using DPDK. In this case, the AF_XDP socket with OVS userspace datapath enables a couple of new ideas. First, unlike OVS-DPDK, with AF_XDP, the userspace datapath can enable TC-flower offload, because the device driver is still running in the kernel. Second, when considering flows which can’t be offloaded to the hardware, ex: L7 processing, these flows can be redirected to OVS userspace datapath using AF_XDP socket, which is faster than processing in kernel. And finally, users can implement new features using a custom XDP program attached to the device, when flows can’t be offloaded due to lack of hardware support.

In summary, with this architecture, we hope that a flow can be processed in the following sequences:
1) In hardware with tc-flower API. This shows best performance with the latest hardware. And if not capable,
2) In XDP. This shows second to the hardware performance, with the flexibility for new features and with eBPF verifier’s safety guarantee. And if not capable.
3) In OVS userspace datapath. This shows the best software switching performance.

Moving forward, we hope to unify the two extreme deployment scenarios; the high performance NFV cases using OVS-DPDK, and the enterprise hypervisor use cases using OVS kernel module, by just using the OVS userspace datapath with AF_XDP. Currently we are exploring the feasibility of this design and limitations. We hope that by presenting this idea, we can get feedback from the community.

I agree to abide by the anti-harassment policy

I agree

Primary author:  TU, William (VMware)
Presenter:  TU, William (VMware)
Session Classification:  Networking and BPF Summit
Track Classification:  Networking & BPF Summit