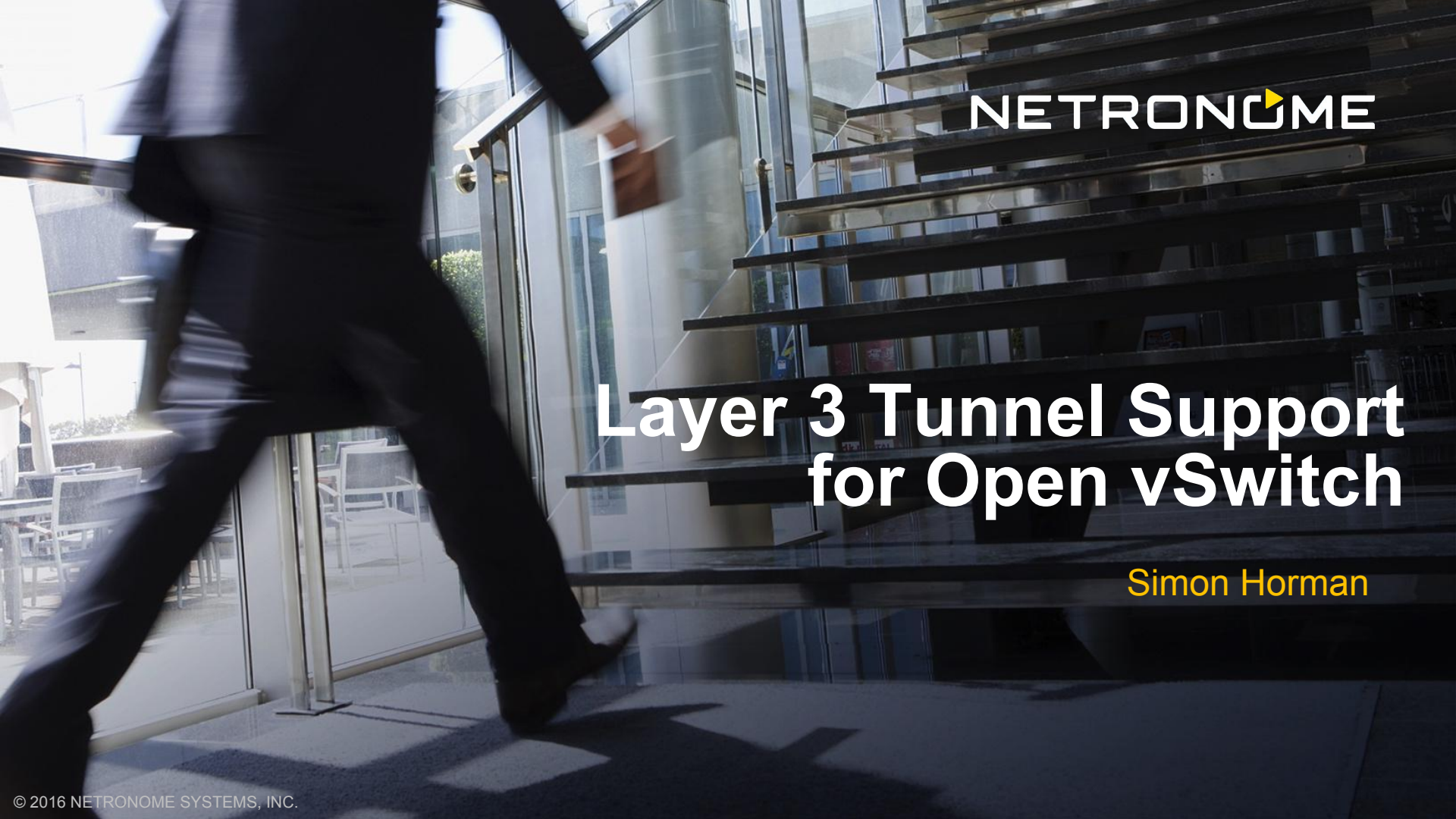


Layer 3 Tunnel Support for Open vSwitch

Simon Horman

Motivation

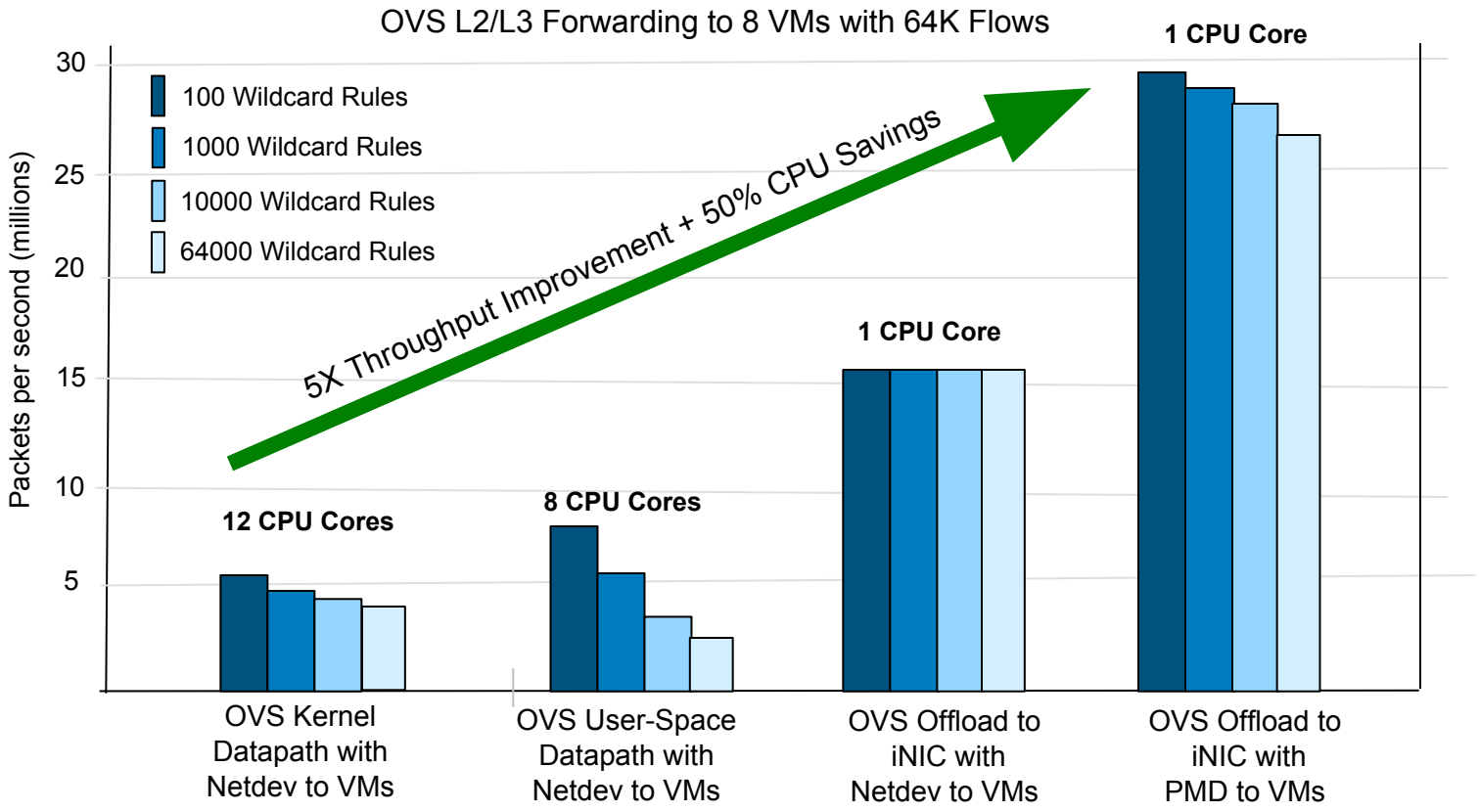
NETRONOME

Would like to:

- Allow rx and tx of packets over tunnels whose payload packet does not have an Ethernet header
- Add these features to upstream OvS then offload them

Importance of Offloading

NETRONOME



Scope

NETRONOME

Datapaths:

- Linux Kernel
- User-Space with and without DPDK

Encapsulation Protocol:

- GRE (non-TEB) (rfc2794):
  - IP protocols over GRE
  - MPLS in GRE (rfc4023)

Background: Tunnel vPorts

NETRONOME

- Encapsulation and decapsulation is handled by output to/input from tunnel vports
- Not currently exposed in Open-Flow

Kernel Datapath Tunnel vPorts

NETRONOME

Kernel Datapath:

- On rx tunnel vport decapsulates packet passing the result and metadata to the datapath
- On tx tunnel vport encapsulates packet based on metadata

User-Space Tunnel vPorts

NETRONOME

Native Tunnelling:

- Tunnel ingress and egress on separate OvS bridge
- Internal rules match ingress and egress packets for tunnel vPorts and apply push and pop tunnel actions accordingly
- Like the Kernel Datapath tunnel metadata is:
  - Available in flow key after decapsulation
  - Used as parameters for encapsulation

Layer 3 Tunneling: Basic Concepts

NETRONOME

- Layer 2 and 3 vPorts
- push_eth and pop_eth datapath actions
- Datapath Attributes and packet type

Layer 2 and 3 vPorts

NETRONOME

- Layer 2 or 3 is a mode of vports
- Default is layer 2: behaviour of all vports until now

pop_eth and push_eth Actions

NETRONOME

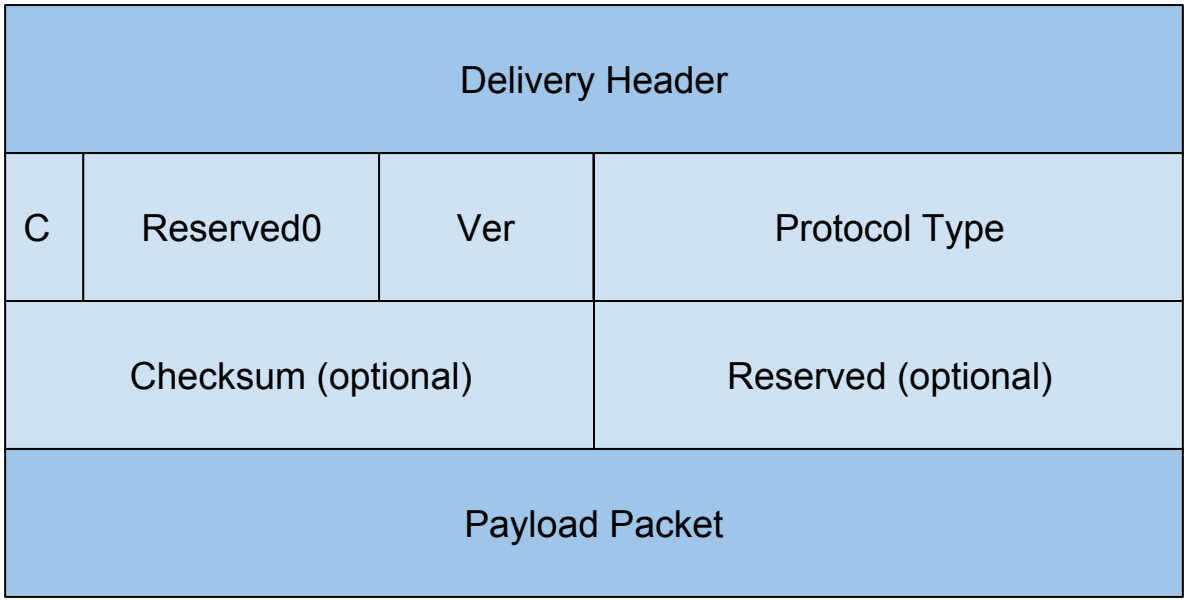
- Add or remove an ethernet header to/from start of packet
- Packets with a VLAN not currently permitted
- MPLS is treated as L2.5 and left alone
- Not currently exposed to OpenFlow:
  - Automatically included in actions of datapath flow

Datapath Attributes and Packet Type

NETRONOME

- Presence of ETHERTYPE and ETHERNET attributes indicates L2 packet
- Presence of ETHERTYPE but not ETHERNET attribute indicates L3 packet
- ETHERTYPE corresponds to Protocol Type in GRE header

GRE Header



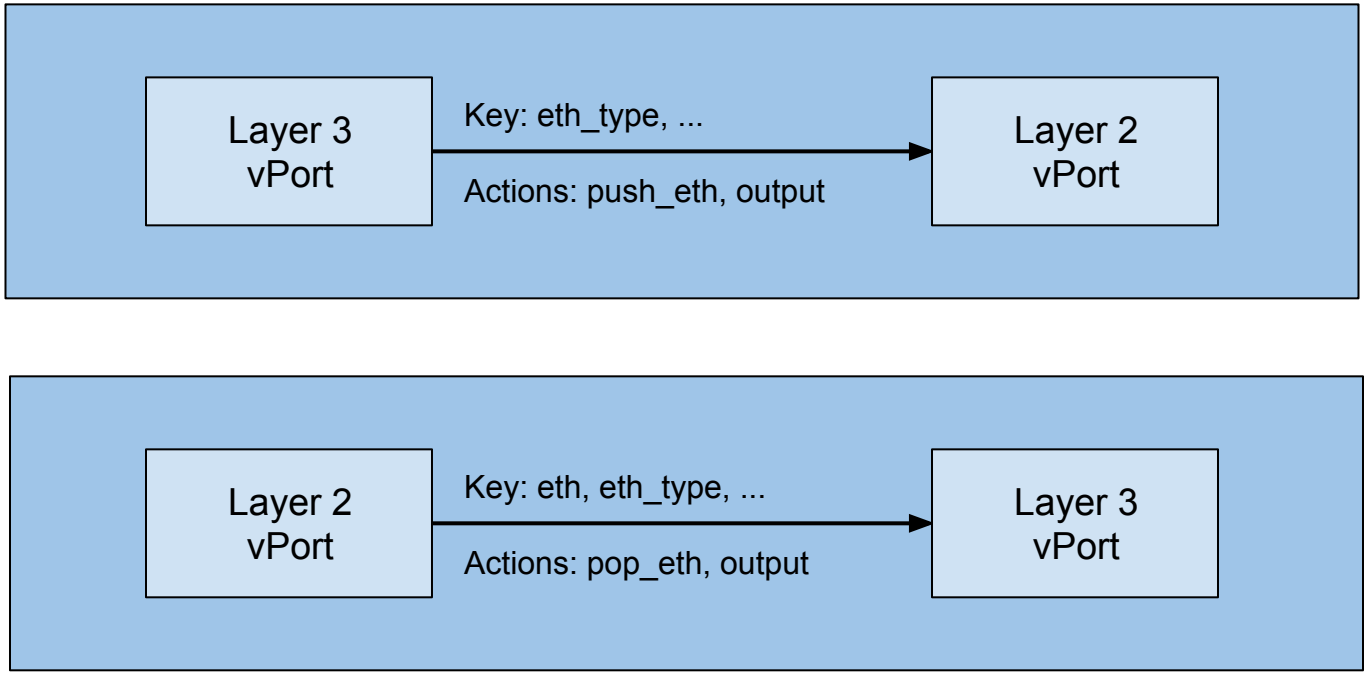
C: Checksum Present

Operation

NETRONOME

- OvS User-Space (ovs-vswitchd) is aware of which vports are Layer 2 and which are Layer 3
- It is aware of the input port for each flow
- And thus when translating from OpenFlow to datapath flows it can add push_eth and pop_eth actions before output actions as necessary

Packet Flow



vPort Implementations

NETRONOME

- User-Space (non-Datapath)
- Kernel Datapath
- User-Space Datapath

User-Space (non-Datapath) vPorts

NETRONOME

- vPorts have new layer3 flag to distinguish layer mode
- vPorts of the same type (e.g. GRE) but different layer mode share the same datapath vport

Kernel Datapath vPorts

NETRONOME

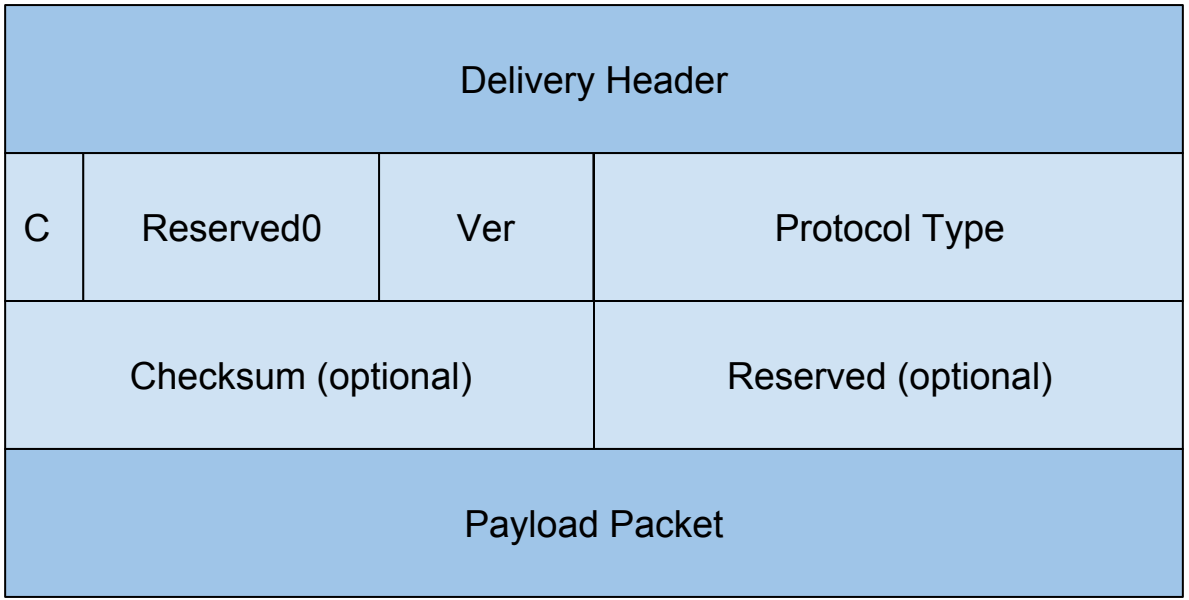
- Switch to using ipgre rather than gretap netdev in kernel
- ipgre (and ipvxlan) vports have recently been enhanced to allow rx/tx of TEB as well as non-TEB packets
- Thus facilitating a single datapath vport for use with both layer 2 and 3 user-space vports
- This design was motivated by a desire to avoid vport type explosion

User-Space Datapath vPorts

NETRONOME

- New user-space datapath only NEXT_BASE_LAYER flow key attribute
- Used to distinguish flows with layer 2 and 3 payload packets

GRE Header



C: Checksum Present

Configuration Example

NETRONOME

```
ovs-vsctl add-port br0 tun1 -- \
        set Interface tun1 type=gre \
                options:remote_ip=10.0.0.2 \
                options:key=flow \
                options:layer3=true
```

Future Work

NETRONOME

Encapsulation Protocols:

- MPLS in IP (rfc4023)
- MPLS in UDP (rfc7510)
- NSH (draft-ietf-sfc-nsh-05)
- VXLAN-GPE (draft-ietf-nvo3-vxlan-gpe-02)
- LISP (rfc6830)

Credits

NETRONOME

Many, including:

- Lorand Jakub, Thomas Morin: Original implementation
- Jiri Benc: Kernel Tunnel Enhancements

Availability

NETRONOME

Open vSwitch (User-Space):

https://github.com/horms/openvswitch l3-vpn

Kernel (Datapath):

https://github.com/horms/linux l3-vpn

Working towards upstream merge!

Questions

NETRONOME