Virtual Extensible LAN Tunnel EndPoint with Internet Network Telemetry support – using P4

Abstract:

Virtual Extensible LAN (VXLAN) is an encapsulation method to extend layer 2 traffic over a layer 3 or IP-based network. It is designed to provide the same Ethernet layer 2 network services as a virtual LAN (VLAN) does today, but with greater extensibility and flexibility. Compared to VLAN, VXLAN can extend layer 2 segments across the data center network by using MAC-in-UDP encapsulation scheme, as shown in Figure 1.

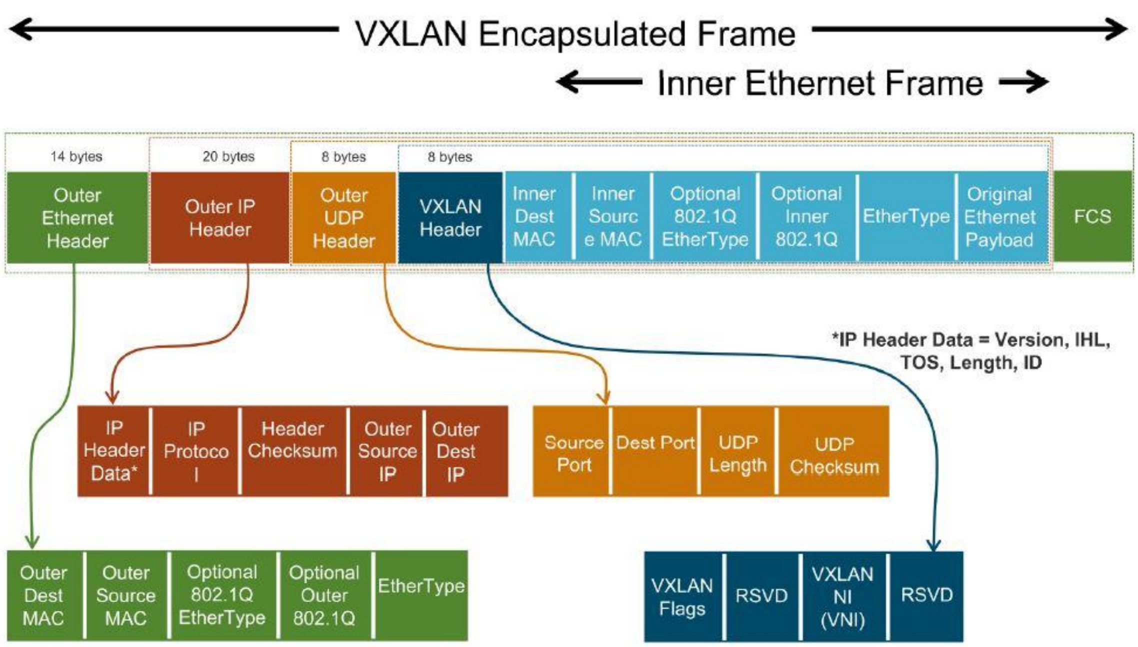


Figure 1

VXLAN uses VXLAN tunnel endpoint (VTEP) devices to map tenants' end devices to VXLAN segments and to perform VXLAN encapsulation and de-encapsulation. Each VTEP function has two interfaces: One is a switch interface on the local LAN segment to support local endpoint communication through bridging, and the other is an IP interface to the transport IP network.

The IP interface has a unique IP address that identifies the VTEP device on the transport IP network known as the infrastructure VLAN. The VTEP device uses this IP address to encapsulate Ethernet frames and transmits the encapsulated packets to the transport network through the IP interface. A VTEP device also discovers the remote VTEPs for its VXLAN segments and learns remote MAC Address-to-VTEP mappings through its IP interface. The functional components of VTEPs and the logical topology that is created for layer 2 connectivity across the transport IP network are shown in Figure 2.

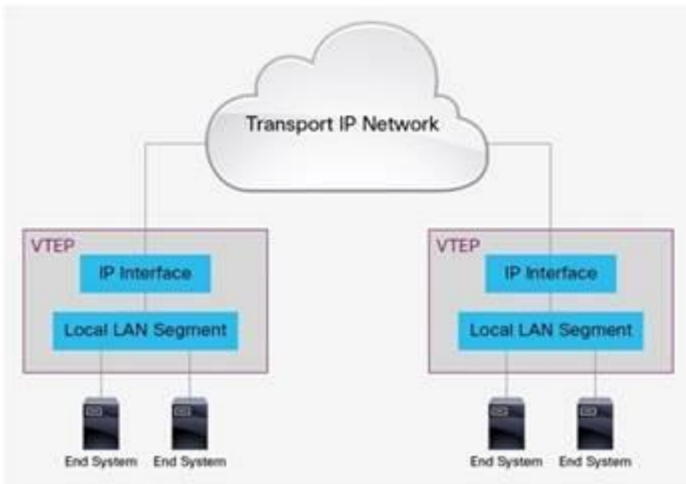


Figure 2

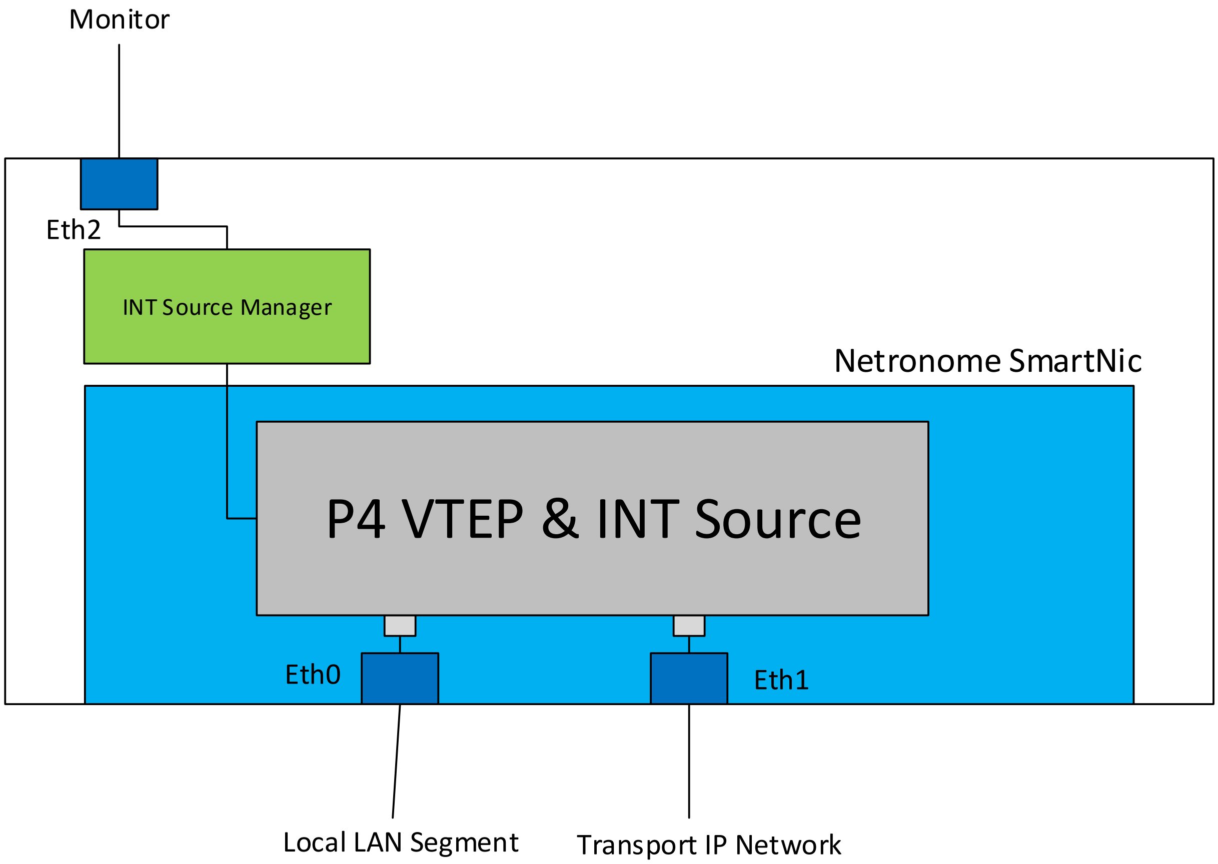
In-band Network Telemetry (INT) is a new framework designed to allow the collection and reporting of network state, by the data plane, without requiring intervention of additional control plane protocols. Data packets are instrumented with INT header fields that contain “Telemetry Instructions” to INT-capable network device. The instructions tell the devices what Telemetry data to collect and the collected data is written into the forwarded data packets.

INT data that can be collected includes: switch IDs, Input/Output Port IDs, Hop Latency, Queue Occupancy, In/Out Timestamps and more. Such telemetry data provides immense value in networks in general. It enables real-time debugging of network issues as well as “self healing” networks.

Programming Protocol-independent Packet Processor (P4) is a high-level language that enables to program the data plane and so due its flexibility, it enables to support emerging new protocols such as INT.

Goals:

- INT Reference - Refer to http://p4.org/wp-content/uploads/fixed/INT/INT-current-spec.pdf
- Build Netronome SmartNIC environment to be able to deploy P4 code. Refer to: "Development Tools User's Guide ,Appendix E. P4 CLI Tools Example" , and perform the following labs:
    - The Basic NIC, including load balancing and meters, lab is available at https://github.com/open-nfpsw/p4_basic_lb_metering_nic
    - The VNF latency measurement lab using in-band telemetry is at: https://github.com/open-nfpsw/vnf_telemetry_lab
    - Meters example is at https://github.com/open-nfpsw/meters_lab
- Implement P4 program into the Netronome SmartNic that will implement VTEP
- Add INT Source functionality to the P4 program that will add INT metadata header bits {for example: switchID, hop latency } to any ingress IP unicast packets from the local LAN. The INT Bits will be determined by the monitor program.



Requirements:

Internet Networking Course, Phyton