

# LINUX 84: Etherate

Layer 2 Ethernet Testing

<https://github.com/jwbensley/Etherate>

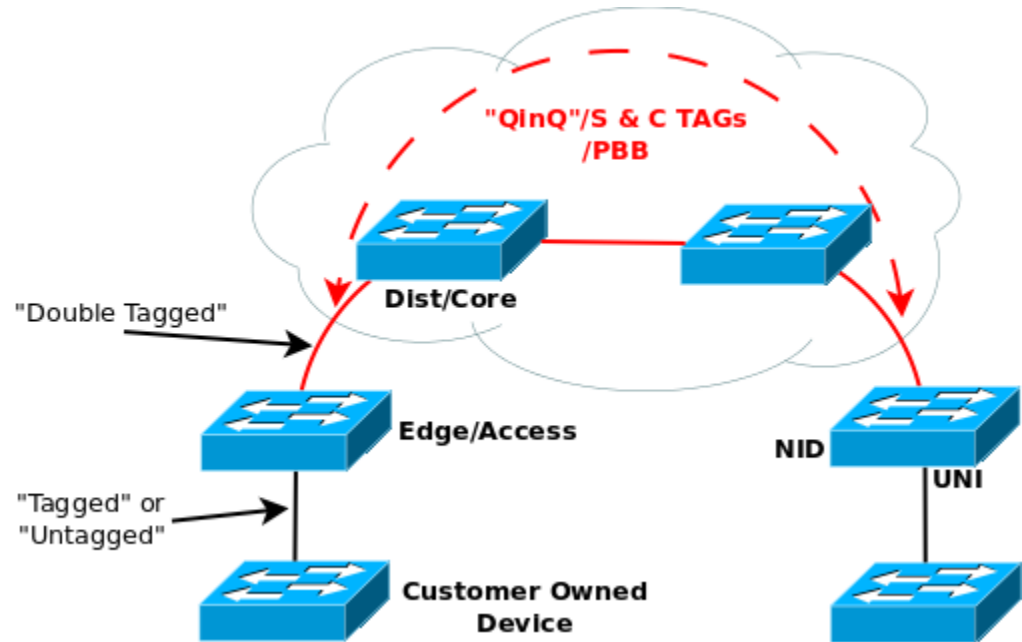
James Bensley,  
Vostron Ltd,  
jwbensley@gmail.com



**VOSTRON**  
mission critical networks

# Background To The Problem

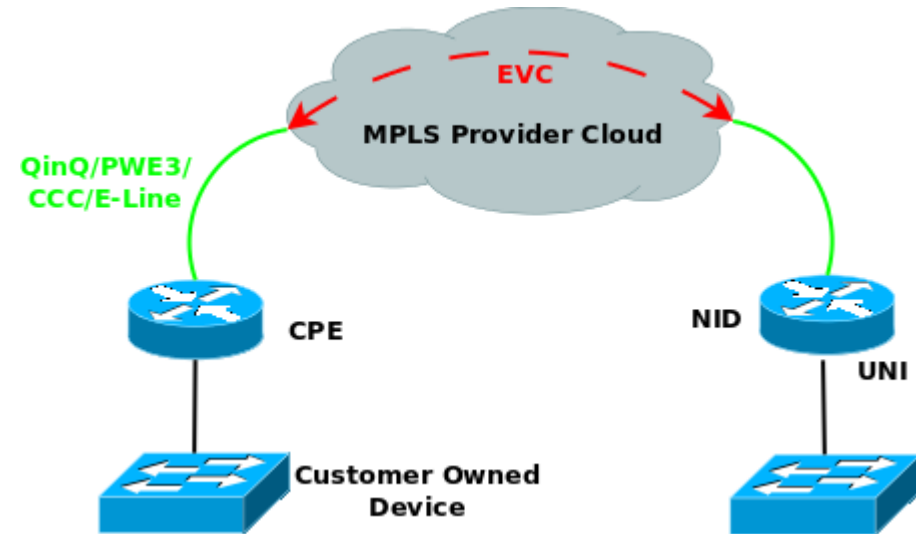
- “Old school” Layer 2 interconnect model - *Brag about my age and lack of FR exposure*
- Layer 2 Ethernet end-to-end, long haul Ethernet is “difficult” (read: ridiculous) at best - *huge L2 domains, STP is a catastrophe waiting to happen, seen customers affecting the core!*



Wrong s-tags/c-tags, loops, exchange of BPDUs, many problems...

# Background To The Problem

- Queue the “evolution of MPLS montage sequence” ...
- Typical present day L2 on-net Ethernet circuit diagram
- This isn't perfect either

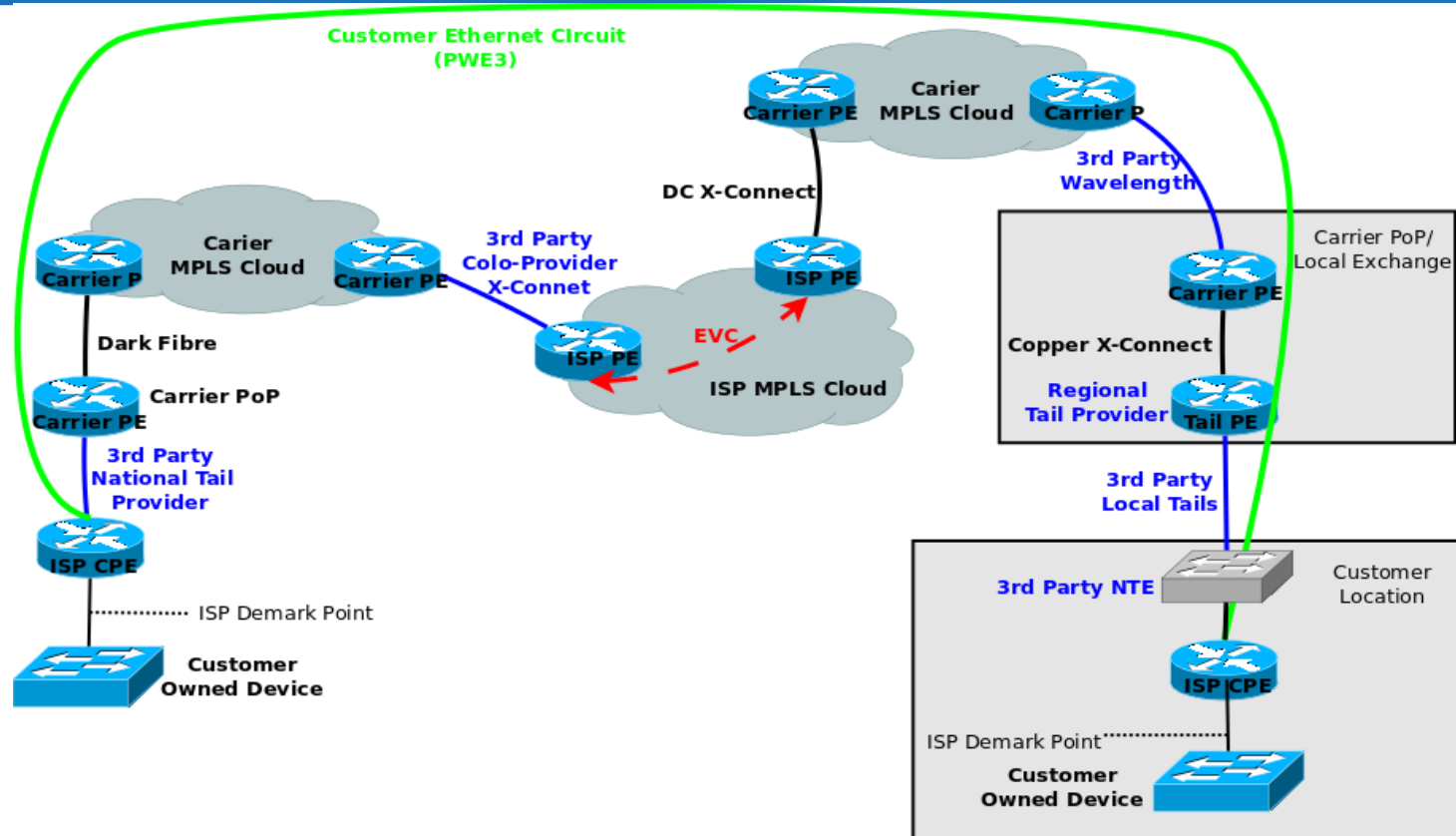


# Background To The Problem

**Real Life Example** (could be metro Ethernet circuit, PWE3, VPLS, copper cross connect, and so on):

- Sample network for customer with two offices that require 100Mbps layer 2 Ethernet connection between them
- ISP is taking entire solution from a single carrier, could be worse if ISP contracts with multiple carriers
- The following isn't the worst possible scenario, just a bad one!

# Background To The Problem



# The Problem Definition

- Many provider domains involved in one continuous link (MPoF) - *Tell the story of the Czech multi-links*
- Providers disagree on demark of responsibilities, inter-provider issues are resolved very slowly, if ever! - *Tell the story of the handover between two major carriers*
- Multiple SLAs of varying levels are signed, less are delivered - *Tell the story of the Germany tail fault resolution*
- Each provider is hopefully using OAM, some sort of reasonable monitoring and a reliable testing solution - *Tell the story of the UK carrier midday flaps, pro- > re -> no-active support*
- and more of course.....

# The Problem Definition

- Providers are hopefully performing end-to-end delivery tests on their respective sections of a circuit, who provides total end-to-end testing? - *The ISP would need to perform end to end testing (even if the carrier does, its usually a contractor)*
- How do we know all equipment in the network path will pass all the types of traffic it needs to (QinQ tags, different EtherTypes, MPLS labels etc) - *The ISP would need to interrogate the carriers configuration*
- How does the ISP check that carriers are honouring priority values? (How does the ISP also check it's own network?) - *The ISP would need to generate predefined test traffic to validate the results*

*\* The ISP doesn't HAVE to do any of this but I want to guarantee service level and operation to be confident in our own products?*

# Solution Criteria

- Armed with programming skills my laptop should be all the hardware I require for day to day tasks
- Build an open source client/server model CLI testing tool that can be used on commodity hardware



# My Solution: Etherate

- Generate layer 2 Ethernet traffic directly on the wire (almost)
- Change the traffic properties and test scenario to “rate an Ethernet” link (*see what I did there*)
- Be able to easily perform tests from a centralised PoP



# My Solution: Etherate

**I have no interest in developing a GUI!**

**I have no interest in porting to another OS!**

# My Solution: Etherate

- Generate traffic with varying properties;
  - Any EtherType
  - Any Source MAC and/or Destination MAC
  - Any VLAN ID
  - Any priority (PCP) value
  - Supports double tagging (inner and outer VLAN ID and PCP)
  - Toggle DFI bit
  - Toggle frame acknowledgement
  - Optional speed limit in Mbps or Frames/p/s
  - Optional frame payload size
  - Optional transfer limit in MBs/GBs
  - Optional test duration

# My Solution: Etherate

- Test result details;
  - One way delay and Round Trip Time (separate unidirectional measurements)
  - Uni & bidirectional speed testing
  - Frames p/s count, Mbps count, total MBs transferred
  - Dropped frames count (M.I.A) and number of non-test frames received
  - Maximum MTU size

# My Solution: Etherate

- Example tests;
  - Delay and RTT
  - Max speed -> can we achieve the CDR our carrier promised, does our customer link hit our QoS policing or shaping configuration?
  - Hardware/circuit performance -> Frames p/s count, backplane & ASIC testing, buffer/queues/shaper testing
  - NOC testing -> Do interface counters increment, send bad frames to generate errors, calibrate traffic monitoring & billing systems

# My Solution: Etherate

- Example tests (cont);
  - Are we passing QinQ tags, are we honouring PCP values?
  - Are we allowing/denying EtherTypes as required?
  - Verify MTU size end-to-end

# Future Development

- Add continual “OWAMP” tests (currently just at the start of a test)
- Add L2 storm control (broadcast and multicast) testing (no RX host involved)
- Add feature to load frame payload from a text file
- Add BPDU & keepalive generation shortcuts
- Report throughput if additional headers (IPv4/6/TCP/UDP) were present
- Add max MTU scanning/sweeping test

# Long Term Future Development

- Add RFC2544 & ITU Y.1564 compliance - *Still reading the documents*
- Add BERT - *Still researching bit patterns*
- Format output to go into a storable format like a DB, for stats gathering - *Other features need to be finished first to give a better idea of the data that would be collected*
- Add layer 1 / wire testing features (auto neg, duplex, speed, cable length, cross pairs etc) - *Still researching layer 1 coding possibilities*



**Q1 2014 ==  
Etherate Beta 0.5**

<https://github.com/jwbensley/Etherate>

**I'm looking for more networks to test on!**

James Bensley,  
Vostron Ltd,  
jwbensley@gmail.com



**VOSTRON**  
mission critical networks