Understanding MPLS Hashing

And how it affects your brand new equipment

Jeff Wheeler
jsw@inconcepts.biz

Job Snijders
job.snijders@atrato.com

NANOG57

Who are we?

Jeff Wheeler

Consultant
- design, implement, support
- whole life-cycle of network

IRC: AMAG

Hobby: Live Sound

Who are we?

Job Snijders

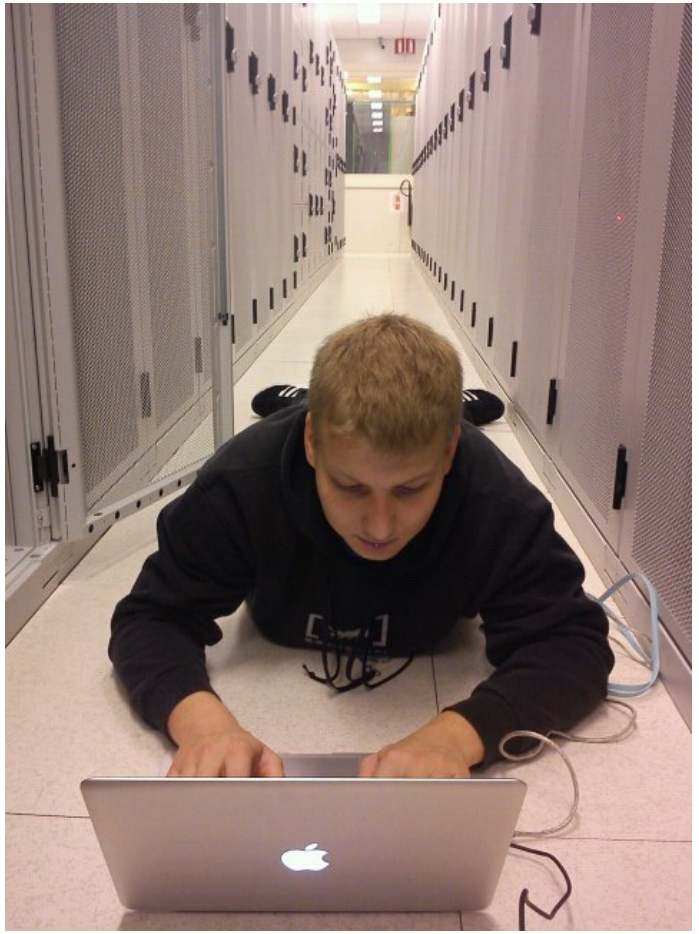
Network engineer @
AS 5580 (Atrato IP Networks)

Founder of NLNOG RING

Twitter: @JobSnijders

Hobbies: IP Routing, LISP, MPLS, IPv6,

Shoe size: 45/EU



Problem statement

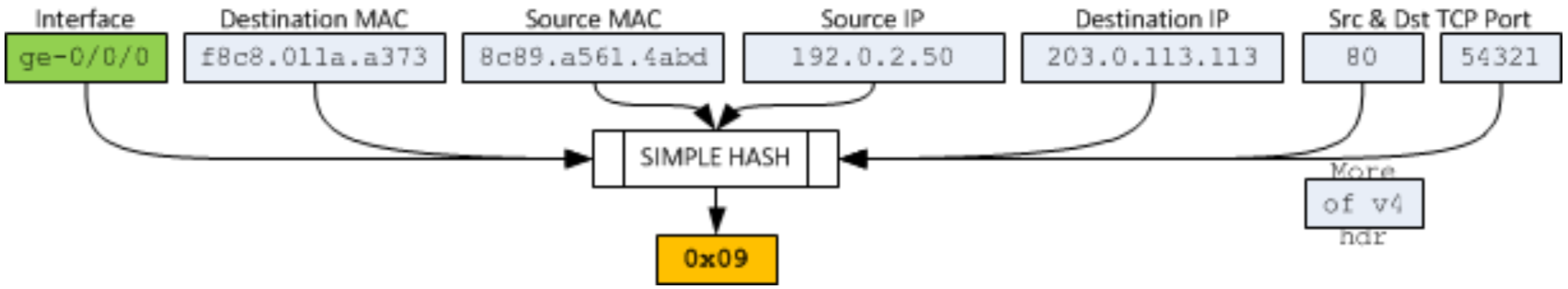
Mega ridiculous!

MPLS P Routers are not aware of
what type of traffic is flowing through them
(by design)

and thus they have to ***gamble***
when hashing and load balancing.

What is hashing and why do I care?

When a router or switch load-shares traffic onto two or more links (LAG, ECMP) it must find some fields in the packet to decide which links to use for each packet

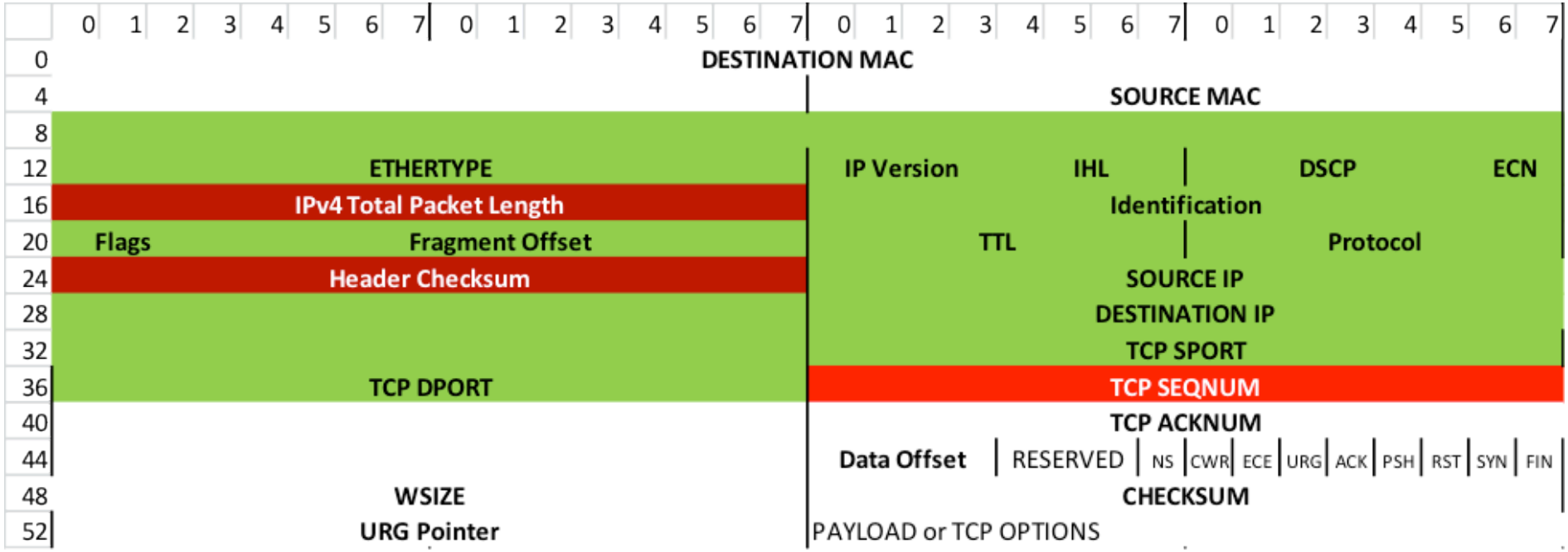


Some vendors allow you to customize what fields may be used:

```
forwarding-options hash-key family mpls {
    label-1;
    label-2;
    ether-pseudowire;
    ip;
}
```

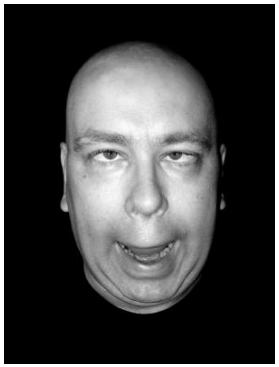
```
Brocade(config)# load-balance speculate-mpls-ip
```

IPv6 hash applied to ethernet frame w/ IPv4 TCP



Why would this happen?

- Vendors naively speculate what actual payload is
- They might look at first octet:
  - > if “4”, the packet must be an IPv4 packet
  - > if “6”, the packet must be an IPv6 packet



BROCADE

CISCO SYSTEMS

juniper NETWORKS

So… Sup' with 4 and 6 MACs?

Photo from 2012:
Here factory workers are producing Line-cards with MAC addresses starting with a 4 or a 6

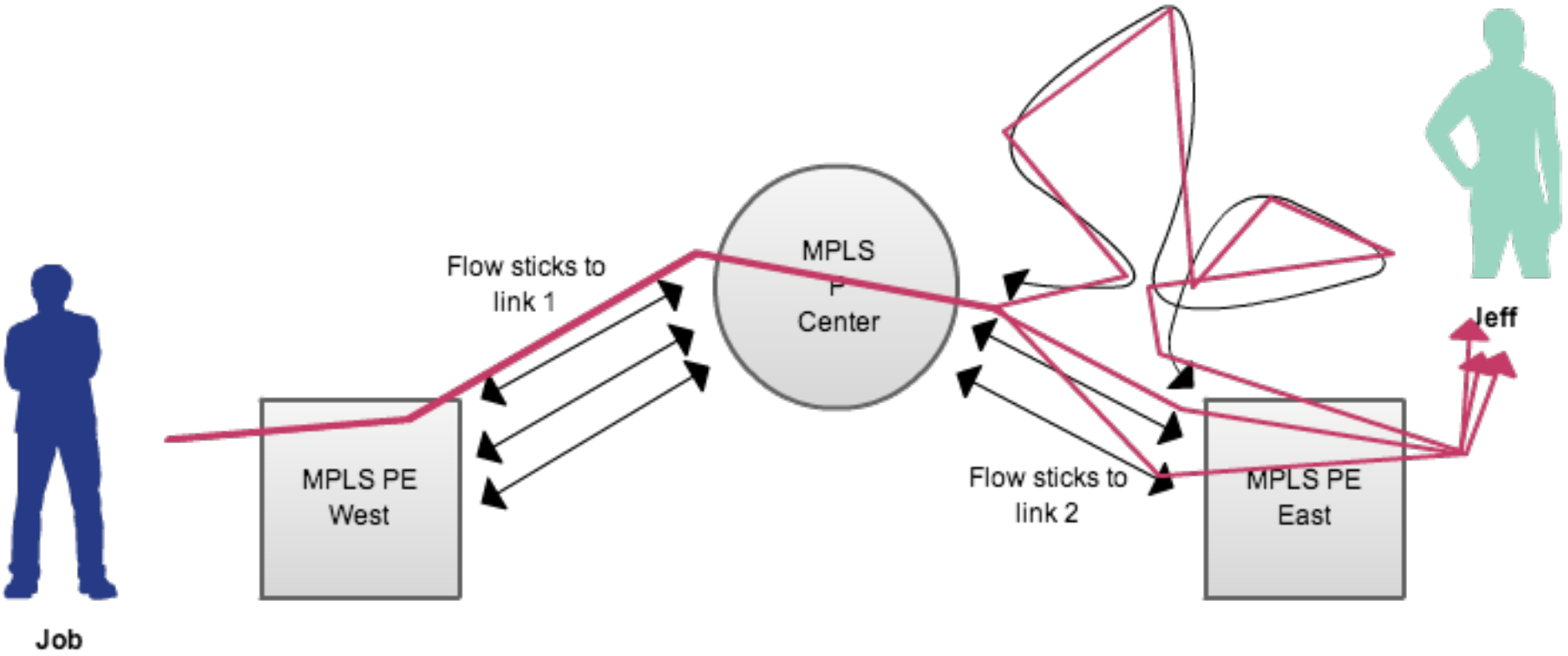
(Juniper, Cisco, Apple, Samsung, HTC, etc)

Qcuik teh fxo bworn over jumped angry the dog

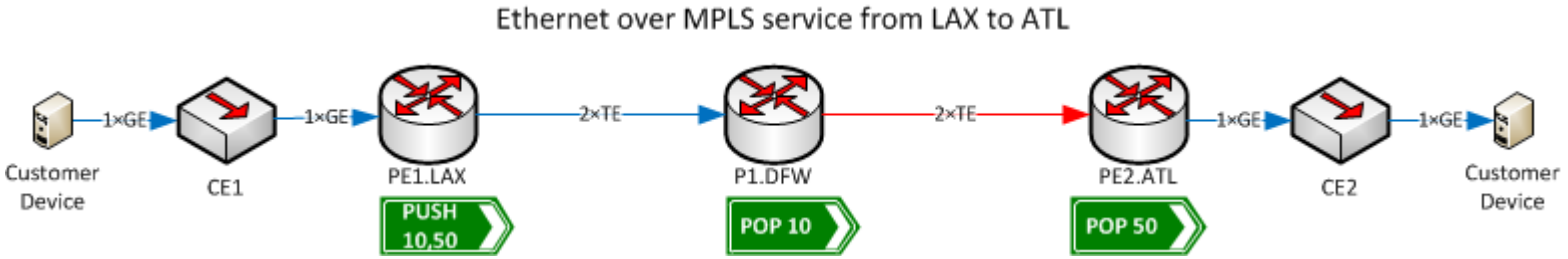
(The quick brown fox jumped over the angry dog)

- TCP implementations have a hard time dealing with out-of-order packets
- Causes TCP re-transmissions and slow-starting
- Even when there is no packet loss

Load balancing overview: out of order



Example pseudo-wire service



- Two MPLS labels are added to the customer's packets when they leave PE1.LAX for P1.DFW (**10, 50**)

- 10 **Top label** 10, tells P1.DFW where to send the packet. It does not have ***any information about the contents of the packet*** or the meaning of label 50.
- 50 **Bottom label** 50, will still be on the packet when it arrives at PE2.ATL. This tells it what VPN or pseudo-wire the packet belongs to.

Example Cases

Case #1 Remote Peering

A network wants to connect to AMS-IX or LINX through a Pseudo-wire delivered by a Service Provider (so called remote peering)

Expected traffic:

ethernet frames containing IPv4 TCP (and maybe a little UDP)

Case #1 Remote Peering

Now what happens when the participant sends traffic to a MAC address starting with a 6?

Case #1 Remote Peering

Case #1 Remote Peering

Is this real?

6-MAC prevalence - 16 October 2012

AMS-IX: 2.89%

DECIX: 2.44%

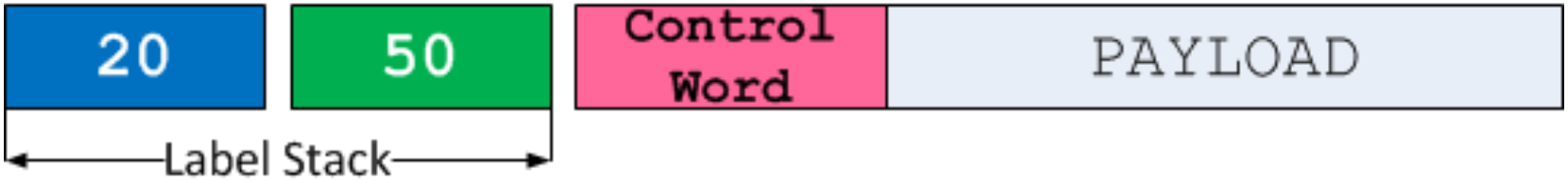
LINX: 1.45%

Case #2 Enterprise VPLS or PW

- Customer buys VPLS or point-to-point pseudowire service from your network
- They have mysterious throughput problems (reordering) with traffic destined to machines with MAC 4X: or 6X:
- Engineers should put ‘check MAC’ on debug list.

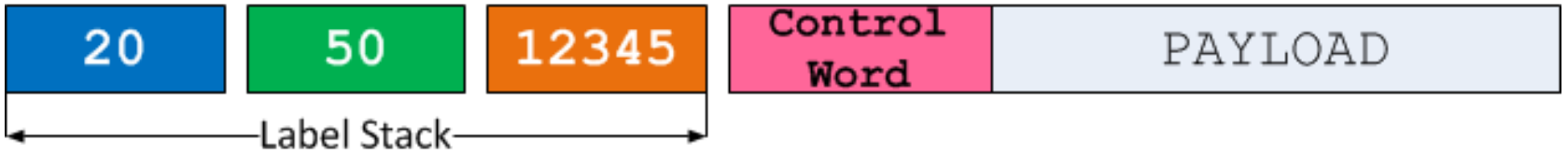
Solutions

Solutions: PW Control Word



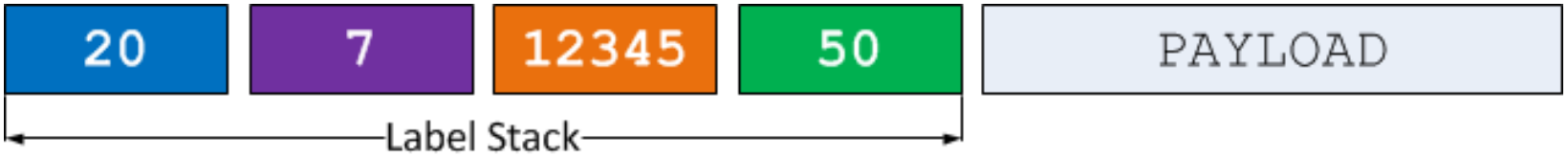
- RFC4385 (February 2006)
  - Ingress LSR (source PE) stuffs 4 bytes at the beginning of the payload
    - Signaled by LDP, BGP
    - First nibble is 0000b so P nodes won't think the payload is IPv4 or IPv6
- Both PEs must have support
- It **can** help if your backbone doesn't know about the feature.
- Control Word can carry a sequence number

Solutions: Flow-Aware Transport of PWs



- RFC6391 (November 2011)
  - Ingress LSR (source PE) imposes an extra label, called the "**Flow Label**," containing entropy information beneath the inner-most label
  - New **Flow Label** – the entropy information supplied by the Ingress LSR. Signaled by LDP
- Both PEs must support it.
- It **can** help if your backbone doesn't know about the feature.

Solutions: MPLS Entropy Labels



- RFC 6790 (November 2012)
  - New **Entropy Label Indicator** – a reserved label value (7) that tells LSRs the following label contains entropy
  - New **Entropy Label** – the entropy information supplied by the Ingress LSR.
- Both PEs must support it.
- Signaled by LDP, BGP, or RSVP
- It **can** help even when your backbone doesn't understand Entropy Labels

Alternative Solutions

Alternative: Better Duck-Typing

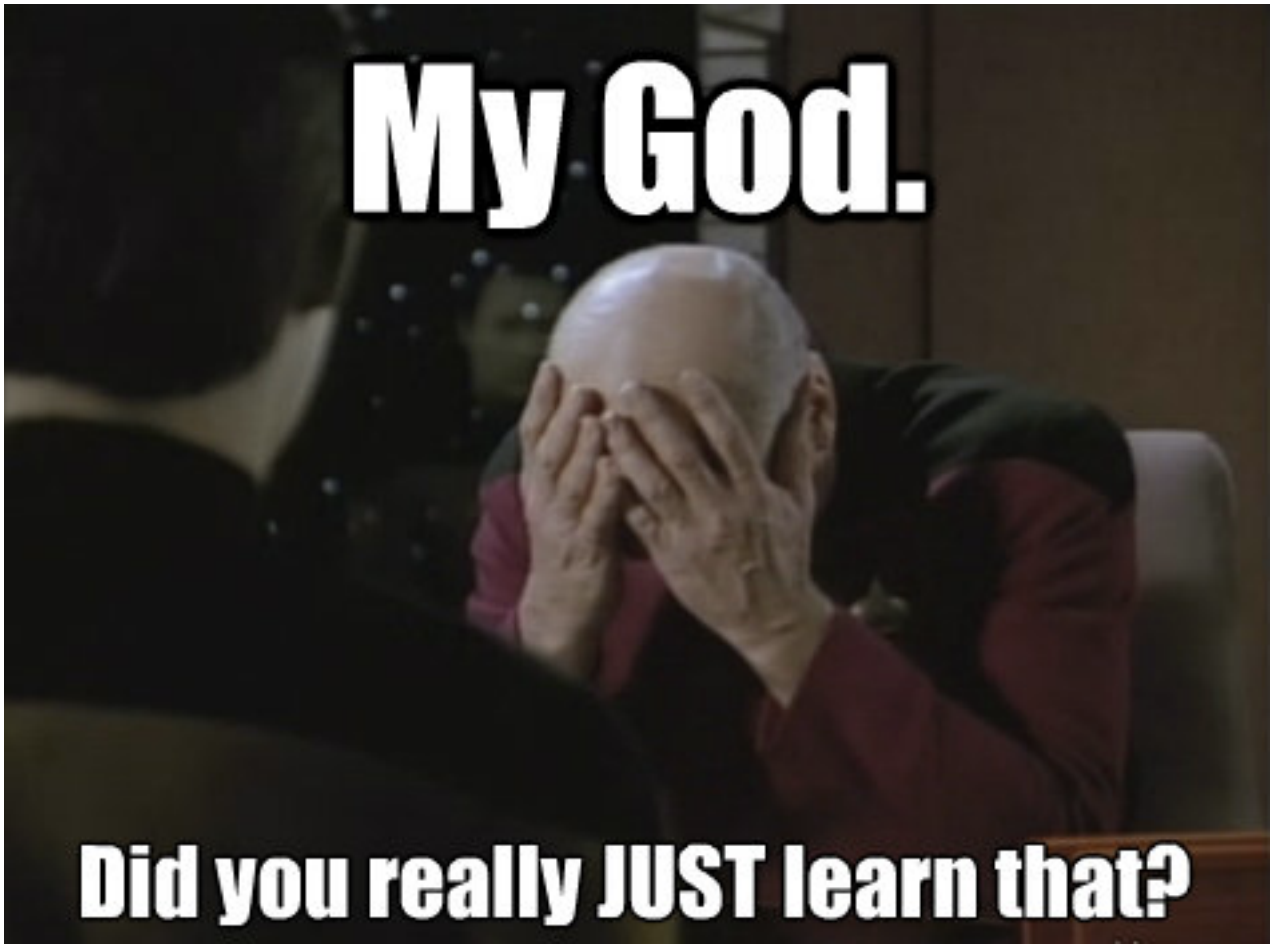
Essentially, improve reliability of the “guessing” mechanism that P nodes do

- Compare IPv4 or IPv6 length, checksum, etc.
- If they don’t match assume it is an Ethernet frame

Alternative: More MPLS Ethertypes

- Currently two Ethertypes used for MPLS
  - 0x8847 mpls-unicast
  - 0x8848 mpls-multicast
- Could create more to identify payload type
  - mpls-ipv4-unicast
  - mpls-ipv6-unicast
  - mpls-ethernetpw-unicast
- Treat existing types as “unspecified payload”

Alternative: Improve TCP



Time schedule

- better ducktyping:
  - Brocade: Q2 2013 in 5.5
  - Juniper: no idea & no need
  - Cisco: no idea & no need
- MPLS Ethertype:
  - No support, has to be defined in IETF first
- Fix TCP so it becomes reorder tolerant
  - LOL! As if…
  - Probably too costly

So, Why isn't AMS-IX suffering today?

AMS-IX:

- Brocade MLX MPLS network
- No MPLS encapsulated IP traffic
- Only MPLS encapsulated Ethernet traffic
- Can safely do 'no load-balance speculate-mpls-ip'

Why was this problem for Atrato?

Atrato:

- Brocade MLX MPLS network
- Transports IP traffic (transit / peering)
- Transports MPLS encapsulated Ethernet traffic
- Solution: Forced to ‘no load-balance speculate-mpls-ip’ and push back BGP-free core

Why isn’t LINX suffering today?

LINX:

- Juniper PTX core, MX based edge
- PTX only balances based on labels
- Large amount of any to any LSPs
- Juniper MX PE nodes set control-word

Funny thing Cisco ASR 9k did with 6 DMACs

On ingress it checked first octet for 4 or 6:

- If payload began with a 6 it would compare calculated packet-length with packet-length inside packet itself.
- Would obviously fail when considering an ethernet packet to be an IPv6 packet: thus drop packet to the floor.

Found in: 4.2(1)BASE, fixed in 4.3(0.32)I

Questions?!

Ps. Did you know that vendors have known this for a long time (was documented in BCP128 in 2007, Control-Word drafts mentioned the problem in 2004)