

# Considerations Arising from PCE-CC Proposals

Adrian Farrel ([adrian@olddog.co.uk](mailto:adrian@olddog.co.uk))

IETF-94 : Yokohama : November 2015

# PCE Centralised Controller (PCE-CC)

- A couple of I-Ds over the last year
  - The Use Cases for Using PCE as the Central Controller(PCECC) of LSPs
    - draft-zhao-pce-central-controller-user-cases
  - PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of LSPs
    - draft-zhao-pce-pcep-extension-for-pce-controller
- Examining the role of a PCE as a centralised control in an SDN-like architecture

# Two “New” Functions Emerge

1. Using PCE to compute a path and then using PCEP to touch each node in the network to establish the end-to-end LSP. The underlying forwarding might be MPLS swapping or MPLS pop-and-go (segment routing), and PCEP is essentially being used as an equivalent to OpenFlow or Netconf.
  2. Using PCEP to install a packet classification rule for LSPs. This seems to be a big missing component in the case of delegated/initiated LSPs where the PCC/LER has no idea what it is supposed to use the LSP for.
- These are related but significantly different

# SBI : What Can We Do Already?

- A TE-LSP is a series of “cross connects” and “resource reservations”
  - Each is a mapping from {input interface, input label} to {output interface, output label}
- PCEP allows an active PCE to install a TE-LSP in the network
  - The “cross-connects” are indicated by the ERO
  - An ERO can include label information (GMPLS)
- LSPs can be short
  - A single hop LSP can be just one “cross-connect”
- PCEP is already an SBI

# SBI : Work Might We Do?

- The ERO approach is a little ugly
  - It might trigger the signalling component to attempt to do work
  - We haven't worked much on "upstream interface for head-end LER" in GMPLS or PCEP
- We could add to PCEP specifically for this function
  - Not a lot of work

# SBI : How Excited Should We Be?

- There seem to be a number of existing SBIs
  - NETCONF
  - OpenFlow
  - ...
- Why develop a new one?
  - Arguments include:
    - We already have to implement PCEP
    - We already have a PCE
    - It doesn't necessitate any changes to PCE or PCEP
- Other applications might include
  - DetNet
  - 6tisch

# Traffic Classification for LSPs

- When a TE-LSP is set up, the head end needs to know how to use it
  - What traffic to send on the LSP
  - Whether it is a virtual link
  - Whether to advertise it in the IGP
  - What bits of this information to signal to the tail end
- PCEP allows an Active PCE to set up or modify LSPs
  - But we have no way to tell the head end how to use the LSP
  - This is because of history
    - It used to be the LER that made the request of the PCE, so it knew why it wanted the LSP
- This function is presumably necessary
  - But it is missing

# TC : How Do We Handle It Today?

- There are several possibilities
  - No-one uses Active PCE
    - The problem doesn't arise
  - Active PCE is used only in controlled environments
    - Head end always knows what the LSP is for
  - Active PCE is used in conjunction with config
    - The LSP is set up using PCEP
    - Some other mechanism tells the head end what to do
  - Active PCE is used in conjunction with BGP Flowspec
    - Possibly not what BGP Flowspec was designed for
      - But it works
- Note that the last two of these seem a waste
  - Why separate the functions?
  - Could use one protocol for everything

# TC : What Might We Do?

- It would not be hard to add some Objects and TLVs to PCEP
- Describe:
  - How to use the LSP
  - How to advertise the LSP
  - Extra signaling information
- We already have ways of describing flowspecs
  - Can re-use encodings (e.g., from BGP Flowspec)

# Suggestions for the WG

- Decide whether either case is related to ACTN
  - Some suggestions made at IETF-93
  - Doesn't seem related to me
  - Maybe both functions could be applied in ACTN
- Keep the two functions separate
  - They seem to have different motivations
  - The solution work is quite different
- Determine implementer/deployer support for each function
- Do not develop standards unless there is support
- Work SBI as an Applicability Statement
  - Develop protocol extensions only to fill gaps
- Work TC as extensions to Stateful PCEP
  - Doesn't seem to be relevant for Stateless PCE