NVO3 dataplane encapsulation requirements discussion

Erik Nordmark, Arista Networks

Background

- Discussions in the NVO3 interim meetings and on the mailing list
- These slides try to expand on that a but
- Goal is to discuss this here and now

My Assumption 1

- Using MPLS for the dataplane encapsulation is not the design center; NVO3 will use an encapsulation which includes an end-to-end VNI identifier

My Assumption 2

- We want to focus on the dataplane **encapsulation** requirements
- Requirements on how the operational or implementation requirements of devices in the dataplane can be left out (at least for now)
  - Those seem to be independent of the encaps format

My Assumption 3

- Goal is to have a minimal set of encapsulation requirements; say 2-4 pages of text
- Get consensus on those
- Later compare protocols against the requirements

Perfection is achieved, not when there is nothing more to add, but when there is nothing left to take away.
Antoine de Saint-Exupery
French writer (1900 - 1944)

VNI ID

- The encapsulation MUST support an end-to-end VNI ID field. This field MUST be large enough to scale to 100's of thousands of virtual networks
  - At least 24 bits? At least 32 bits?
  - Do we envision needing to grow this later?
    - Note that NVO3 over NVO3 might be one way to handle unanticipated growth

NVO3 QoS field

- NO need for a QoS/CoS field in the NVO3 encaps
  - We have an outer IP DSCP, plus an inner 802.1Q priority and/or IP DSCP, which is sufficient
- [Current draft-ietf-nvo-dataplane-requirements has this as a MAY]

ECMP

- MUST/SHOULD ? facilitate ECMP in unmodified IP routers in the underlay
  - One way to do this is to use UDP encaps with a UDP source port containing the hash of the encapsulated flow [Originally from LISP]

Security/assurance

- Is it ok if an undetected bit error in the VNI ID result in packet misdelivery?
- Different threats to be concerned about:
  - Off-path attackers that can guess the VNI ID and inject packets?
  - On-path attackers that can snoop packet and do cut&paste (combine valid NVO3 header with a different payload)
- Should we reserve place for mechanisms against such attacks in the base header? In some extension?

Extensibility – different payloads?

- Motivations:
  - For L2 NVO3 carrying Ethernet payload is sufficient
  - For L3 NVO3 want to omit Ethernet and carry IPv4/ IPv6
  - Might also need to indicate payload is BFD (for BFD over NVO3 as opposed to BFD over IP over NVO3)
  - Ability to carry e.g., NSH payload
- Ethernet type vs. IP type vs. NVO3-specific payload type field? How many bits?

Extensibility - OAM

- Some OAM mechanisms send what looks like regular packets, however the decapsulating tunnel endpoint should not deliver those to the endpoint
  - Some protocols have an OAM bit as a result [TRILL]
  - There might be other solutions - like a payload type to indicate “drop”?

Extensibility – meta-data, vendor-specific

- Should NVO3 encaps support carrying additional data for futureor vendor-specific reasons?
- An alternative would be a payload type to specify a vendor-specific header, which is followed by the actual payload.
- Should “old” NVO3 middleboxes be able to skip “new” NVO3 extensions?
- Maximum size of such extensions?

Extensibility – others?

- Congestion control data at encapsulation layer if we develop a method that uses inband signaling,
- Data performance optimizations (remote checksum offload and a form of large segment offload have been proposed)
- Possibly a CRC to cover payload
- Should the VNI ID itself be optional data.

Hardware support?

- It would be nice if the base NVO3 encapsulation (without extensions) can be handled by existing commercial switch chips and NICs
  - Facilitates deployment
  - Probably not a requirement

Other encaps format requirements?

- Or things we can remove from the above?

Next steps?