Update to the IPv6 flow label specification

draft-carpenter-6man-flow-update-03

Brian Carpenter
University of Auckland

Sheng Jiang
Huawei

Presented by: **Shane Amante**
Level 3 Communications, LLC

July 2010

Why?

- ***RFC 3697 says:***
  - ***Flow label must not be changed en route.***
  - ***Nodes must not assume any mathematical or other properties of Flow Label values***
  - ***Router performance should not depend on the distribution of Flow Label values... Flow Label bits alone make poor material for a hash key.***
- ***These rules have caused difficulty for almost all proposed use cases.***

History

- -00 version presented at IETF 77
  - Use MSB of flow-label as signal to receiving node about semantics of flow-label, e.g.:
    - to follow existing 3967 rules (end2end immutable); or,
    - flow-label is “locally defined” (mutable)
  - *Operationally challenging to reset “locally defined” flow-label on exit from a ‘Flow Label Domain’*
  - Downstream AS could easily misinterpret semantics of a received “locally defined” flow-label resulting in unintended consequences, (e.g.: poor ECMP or LAG load distribution).
- Several discussions on 6man list
- Published -03 version for IETF 78

Several challenges with IPv6 flow-label

- (-) Largely unused by both hosts and routers
- (-) No integrity 'guarantee' of flow-label
  - Not protected by header checksum
  - (Outer header) flow-label not protected by IPSec
- (+) Fixed location in header make it straightforward for [very] high-speed routers to use as input-key for LAG and/or ECMP versus:
  - (-) Variable offset of "Next Header" containing Transport protocol info {proto, src_port, dst_port}
  - (-) Brittle nature of existing "Next Header" that do not have TLV structure. Thus, unknown next-headers *cannot* easily be skipped over to find input-keys for ECMP or LAG[1].

[1]draft-krishnan-ipv6-ext-header could fix this, assuming it is moving forward (?)

One, potential conclusion (?)

1. Operationally challenging to restore or reset flow-label at FL domain *exit* routers
   - Nowhere to store an existing flow-label value inside a packet at FL domain ingress
   - No guarantee FL *exit* router will (be properly configured to) restore/reset flow-label
2. No integrity protection of IPv6 flow-label

3. The flow-label is analogous to the IPv4 DSCP and IPv6 TC fields. If a locally defined flow-label is pursued, routers at ingress to a FL domain MUST either ignore or reset the FL.

Where to, from here?

From Brian Carpenter e-mail to 6man WG mailing list on May 6, 2010:

"There appear to be two viable approaches:

1. *End2End Immutable Flow Label*: Definitively forbid locally defined use of the flow label. Strengthen RFC 3697 to say that hosts SHOULD set a pseudo-random label value, which would clarify and limit its possible uses. In particular, its use for load balancing and possibly as a nonce would be encouraged.
2. *Mutable Flow Label*: Encourage locally defined use of the flow label. This approach would make the flow label mutable and would exclude any use case depending on end-to-end immutability. It would encourage applications of a pseudo-random flow label, such as load balancing, on a local basis, but it would exclude end-to-end applications such as [I-D.blake-ipv6-flow-label-nonce]."

Suggested Recommendations

A. Publish this draft as Informational RFC, outlining challenges with flow-label (?)

-- OR --

B. Create & publish RFC 3967bis with either:

- Option 1: Flow Label is end2end IMMUTABLE
  - ASBR *MUST NOT* change flow labels on ingress
  - May allow flow-label to use for load-balancing or *as a nonce (by end hosts) for detecting 3rd party DoS attacks*.
- Option 2: Flow Label is MUTABLE
  - Each AS may ignore or change incoming flow-label
  - Similar to IPv4 DSCP or IPv6 Traffic Class field
  - Egress ASBR's ARE NOT EXPECTED to "fix" (restore, reset) flow-label – too operationally complex & it's a no-op.

Thank You!