draft-venaas-bier-mtud-01

Stig Venaas, stig@cisco.com
Mahesh Sivakumar
IJsbrand Wijnands, ice@cisco.com
Les Ginsberg, ginsberg@cisco.com

BIER MTU discovery

- MTU basics
- MTU discovery
- What discovery options do we have?

- draft-ietf-bier-path-mtu-discovery provides MTU discovery
  - But probe based and only gives MTU for current receiver set and paths
  - If anything changes, need new probe to find new MTU

- Idea is to find an MTU value that is mostly stable and can be used for all BIER packets in a sub-domain, rather than finding the optimal MTU per receiver set and path

MTU basics

- For an IP packet to reach its destination it must be small enough to traverse all the links to the destination.
  - The size must not exceed the IP MTU of any of the links on the path
- Fragmentation by routers (in-flight fragmentation)
  - An IPv4 packet without DF (Don’t Fragment) set is fragmented as needed
  - If a router determines it is too big for the out-going interface, it fragments the packet
- Fragmentation by hosts (and routers when originating packets)
  - IPv6 packets and IPv4 packet with DF are not fragmented when being forwarded
    - Link layer may provide fragmentation though (including tunneling)
  - Originator needs to ensure packet is small enough
- For IPv6 and IPv4 DF, originator can ensure the datagram is small enough
  - Applications can be configured with a safe size, or use e.g., a socket option
  - TCP can also adjust, but we are interested in multicast here.
- What size is small enough?
  - Use a safe minimum? 1500, 1280, or less? Use MTU recommended by network admin?
  - Use MTU of the host interfaces, assuming no link on the path has a smaller MTU?
  - MTU discovery?

MTU discovery and BIER

- MTU discovery allows originator to send packets at an optimal size
  - Application/TCP can send packets at the optimal size, no fragmentation
  - Fragmented packets can be split in the fewest possible number of fragments
  - Optimal size means less packets originated and forwarded
  - With BIER pim/igmp overlays we want to maximize pim J/P and igmp reports (many groups in each)
- IP Path MTU discovery
  - IPv6 usually performs PMTUD (unless assuming max 1280), optional for IPv4 (use of DF)
  - When a router determines packet too big for the outgoing interface, it sends an ICMP message to the IP source address with the max allowed value
  - This might happen multiple times along the path, and when the path changes.
  - Host stack starts with a default MTU for a destination and adjusts it based on ICMP responses
- BIER In-flight fragmentation a bad idea
  - With BIER it would require decap, IP fragmentation, reencap etc, or BIER fragments
- MTU discovery and BIER
  - Can do IP PMTUD where BFR sends ICMP response to the source.
    - Requires decapsulation and reachability to the source. Possibly relay message via BFIR
  - If BFIR knows what MTU is safe across the BIER domain for the specified receiver set, it can send ICMP to the source
  - If BFIR/BFER knows the BIER MTU, it can use it for pim/IGMP overlay (j/p report size)

BIER MTUD solutions

- How can a BFIR know/learn the MTU that can be used?
  - For a given flow the exact BIER path MTU depends on the path/tree, which depends on the receiver set and entropy
- Configuration of safe value, all BIER links have same MTU?
- PMTUD similar to IP (no current proposal)
  - A BFR signals BFIR (source) when data packet is too big for an OIF
    - New BIER message, BIER encapsulated ICMP message?
- PMTUD based on probing
  - draft-ietf-bier-path-mtu-discovery
  - Relies on probes following same path as the data
  - Potentially slower recovery than the above when topology changes
- Sub-domain MTUD (this draft)
  - A compromise. Not optimal. “Weakest link” (the link with the smallest MTU) in the sub-domain determines the MTU.

BIER Sub-domain MTUD

- “Weakest link” (thinnest) in the sub-domain determines the MTU
- MTU will often not be optimal, but better than a safe default
  - An admin could potentially configure the same with full knowledge of the topology, but would also need to reconfigure all routers if the weakest link MTU changes.
- No probing or data triggered signaling
- No new message types, instead a new IGP sub-TLV
- Independent of which BIER links a flow traverses
  - Robust, not impacted by receiver set changes or most routing/topology changes*
  - Less per flow state
- Only possible topology change impact is if the “weakest link” goes down
  - This may cause the sub-domain MTU to increase
  - Draft proposes dampening, it is not urgent to learn the bigger MTU
    - Avoids MTU updates if the link is flapping
- May be combined with BIER PMTUD solutions
  - One may use this MTU as an initial value, but for selected flows increase the MTU based on PMTUD

BIER MTUD issue 1

- What MTU should be advertised?
  - It appears the text in the draft may be ambiguous
- Draft tries to say that routers advertise BIER payload MTU (option 1)
  - That is, how big an IP packet can be encapsulated
- Should it be the largest possible BIER packet? (option 2)
  - Size of IP packet plus BIER encapsulation, size after encapsulation.
- We need BFIR to learn the BIER payload MTU
  - With option 1, BFIR uses the smallest advertised value. Each BFR considers which encapsulations are used on the different links (or to different neighbors) to determine what to announce
  - With option 2, BFIR can subtract the max encapsulation overhead for all BIER encapsulations (all defined, or all known to be deployed in the domain)
  - Option 2 likely simpler. But it may depend on how multiple encapsulations may be used. Can we have multiple encapsulations in a BIER domain, in an IGP area, on a single link? Can the encapsulation change as a packet is forwarded?
- Encapsulation needs to be considered by all MTU discovery mechanism

BIER MTUD issue 2

- How to handle tunneling or repair-paths that require additional encap?

- Have not given this much thought yet. Also applies to other discovery mechanisms.
- If a neighbor is known to be reachable only via tunneling, then that affects what is the biggest BIER payload that can be sent to that neighbor, so this is already taken care of in the draft.
- Not sure how to handle a dynamic event such as a repair path
  - Could potentially announce a new smaller MTU when repair is in place, but the goal is to have a fairly static MTU value. Hence a router supporting use of repair paths should probably announce an MTU that allows for reaching the neighbors via a repair path.