Identification Extension for the Internet Protocol

IETF 118 Conference Intarea Session – November 9, 2023

Fred L. Templin – The Boeing Company
(fred.ltemplin@boeing.com)
Motivation

- Some transports (e.g., NFS/UDP) get better performance using segment sizes that exceed the path MTU and invoke IP fragmentation
- Confirmed by modern network performance analysis (‘iperf3’)
- Examined Licklider Transmission Protocol over UDP (LTP/UDP) in NASA High-Rate DTN (HDTN); JPL Interplanetary Overlay Network (ION) distributions:
  - https://github.com/nasa/HDTN
  - https://sourceforge.net/projects/ion-dtn/

- Testbed setup:
  - 2 Dell Precision 3660 workstations w/ Ubuntu 20.04 LTS
  - 12th Generation Intel Core I7-12700Kx20 w/ 32GB memory
  - Intel E810 CQDA2 100Gbps Ethernet NICs
  - NICs connected by Point-to-Point (P2P) link
  - LTP/UDP/IPv4 over P2P Ethernet
100Gbps Ethernet with 1500 MTU

X-axis - LTP segment size in bytes
Y-axis - Data rate in Mbps

Note: Similar performance profiles seen in ION emulation testbed
100Gbps Ethernet with 4500 MTU

X-axis - LTP segment size in bytes
Y-axis - Data rate in Mbps

20432 Mbps @ 4400 segment without fragmentation
25889 Mbps @ 16000 segment with fragmentation
27951 Mbps @ 64000 segment with fragmentation
100Gbps Ethernet with 9702 MTU

X-axis - LTP segment size in bytes
Y-axis - Data rate in Mbps

Note: CRC-32 integrity protection insufficient for MTUs > 9KB
Implications

- Fewer and larger segments perform better than more and smaller segments (confirmed by ION-DTN LTP large segment performance vs. GSO/GRO)
- Increasing path MTU and using transport protocol segments close to MTU size significantly increases performance, but most Internet paths still 1500 or smaller
- Using transport protocol segments that exceed common Internet-sized path MTUs significantly increases performance when IP fragmentation invoked
  - Performance also increases to a lesser degree when IP fragmentation invoked over larger MTUs
  - Larger segment sizes improve performance by reducing header overhead
  - Better performance when all fragments close to MTU size (e.g., 1400 + 1400 + etc)
  - Poorer performance when final fragment is small (e.g., 1400 + 100)

- Need: Robust IP fragmentation/reassembly service for performance maximization
- BCP: “IP Fragmentation Considered Fragile” [RFC8900]
- Proposed update: “Identification Extension for the Internet Protocol”
Issue #1: Identification Length

- IP Identification (IP ID) is 16 bits for IPv4; 32 bits for IPv6
- Problem: Data corruption possible if IP ID wraps/collides within maximum datagram lifetime (MDL) [RFC4963][RFC6864]
- Need: longer IP ID to ensure reassembly integrity; identification uniqueness
- Solution: IP header extensions for longer IP IDs
  - New IPv4 option for IPv4 packets
  - New IPv6 Extended Fragment Header for both IPv6; IPv4 packets
  - Consider both solutions in parallel
Issue #2: Loss Unit Smaller than Retransmission Unit

- Loss unit (IP fragment) smaller than retransmission unit (transport segment)
- Problem: cascading retransmissions based on fragment loss due to persistent congestion/disruption
- Need: fragmentation/reassembly feedback from intermediate/end systems
- Solution: new ICMPv6 Packet Too Big (PTB) message codes
  - Codes 1 and 2 – sent by fragmenting intermediate systems to request smaller fragments
  - Codes 3 and 4 – sent by reassembling end systems to request smaller (fragmented) packets
  - Messages sent subject to rate limiting wrapped in UDP/IP headers to avoid filtering
  - Source responds by adapting segment/fragment sizes
Issue #3 – IP Fragment Filtering

- IP fragments systematically dropped along some paths
- Problem: fragment-dropping middleboxes
  - Some IPv4 network middleboxes filter based on protocol/port numbers and drop all IPv4 fragments when virtual reassembly not possible
  - Some IPv6 network middleboxes drop all packets with IPv6 Fragment Header
- Need: end-to-end fragmentation parameter transport
- Solution: deep packet fragmentation
  - Embed fragmentation parameters in packet body so ordinary middleboxes see whole packets
  - Special UDP port, IP proto, EtherType presents parameters to limited domain nodes [RFC8799]
Draft Status

- Some preliminary list discussion; off-list comments received
- Adopt as working group document?