Path
Maximum Transmission Unit Discovery
(pmtud)

IETF PreWG
7/17/03

Co-Chairs
Matt Mathis <mathis@psc.edu>
Matt Zekauskas <matt@Internet2.edu>

New mailing list
pmtud {-request} @ietf.org
Agenda

- Preliminaries: Blue sheets, Note takers, etc
- WG Status
- Short history and work to date
- Robustness Issues
- Other Stakeholders
- Plans
Working Group Status

- Recharter/reactivate pmtud WG
- We have formal IESG approval
- Not quite fully launched yet (no announce message)
- New co-chair: Matt Zekauskas
- New mailing list: pmtud {-request} @ietf.org
Charter Tweaks

- Slightly extended from the draft:
  - Charter does not restrict us to just one new method

- Milestones:
  - Jul 03 - Reorganized Internet-Draft. Solicit implementation and field experience.
  - Dec 03 - Update Internet-Draft incorporating implementers experience, actively solicit input.
  - Feb 04 - Submit completed Internet-draft and a PMTUD MIB draft for Proposed Standard.
Standard Path MTU Discovery

- RFC1063 (1988 - obsolete)
- RFC1191 (1990), RFC1981 (IPv6 only, 1996)

- General method
  - Send a large packet with the Don’t Fragment (DF) bit set
  - Routers forward packets as possible; send an ICMP message when the packet is too big for the outgoing link
  - Host receives ICMP message, starts sending smaller messages

- Known problems: RFC2923 (2000)
  - ICMP delivery is fragile and insecure
  - Resulting in connection and application hangs
Recent Work up to IETF 56

- **Documents**
  - Unpublished work following 41st IETF (1998)
  - Non-IETF web draft (Nov 2002)
  - draft-mathis-plpmtud-00.txt (Feb 2003)

- **Implementations**
  - Unpublished running code in 1998
  - Two written to the Nov 2002 draft
    - John Heffner (Linux)
    - Kevin Lahey (NetBSD)

- **PLPMTUD BoF @ IETF 56 to discuss WG status**
Since IETF 56 (Mar 2003)

- WG Chartered

- Completely restructured draft
  - draft-mathis-pmtud-method-00.txt (June, 2003)

- New Implementation
  - Yoshifumi Nishida (ns-2)
The Proposed New Algorithm

- Start with a "moderate" MTU (1k?)

- Test larger MTUs by probing
  - Raise MTU if successful
  - (Optional) process any RFC1191 ICMP
  - Do not reduce TCP window on lost (unsuccessful) probes

- Most of the algorithm runs in the transport layer
  - TCP, SCTP, or higher layer (e.g. NFS)

- Use cached/shared state in the IP layer
Document Issues

- Minimal RFC2026 standards language
  - Lost probes do not trigger TCP congestion control...

- Nearly everything else is heuristic
  - Mostly just recommendations
  - How to avoid over specification?
  - Tweaks are likely to continue forever

- Current (restructured) draft
  - Start from RFC1981 (good layering language!)
  - Generalize to include IPv4
  - Add new algorithm
  - Relax ICMP requirements
Plans for the Next Draft

- Edit pass - consistent terminology
  - Add some missing TCP text

- Seek input (text?) from additional experts
  - IPsec
  - Multicast
  - SCTP

- Good enough for next round of implementors ASAP
A Plea for Implementors

- Most of the document describes recommendations motivated by a desire to maximize robustness in the presence of less than ideal implementations of other parts of the Internet protocol suite.

- Identifying all "less than ideal" requires field experience, based on running code.
Open Robustness Issues

- What can be done if raising MTU raises the loss rate?

- What to do on hard (repeated) timeouts?

- What happens when DF is not honored?

- Can we anticipate any other pmtud failures?
When Raising MTU Causes Loss

- Multi-path across paths with differing MTUs
  - Require an additional lossless RTT

- Parametric failures (e.g. clock skew)
  - Links that don’t enforce MTU, but should

- General solution: heuristic to reduce MTU if window becomes too small
Hard (Repeated) Timeouts

- On double(?) timeouts, reset MTU to 512(?)
  - and trigger timeout actions at other layers...?
    - router discovery, server pooling, etc
    - comprehensive language would be useful

- Do we need to consider the recurrent failure case?
DF Woes

- Gear that ignores IPv4 DF bit
- Tunnels that hide payload DF
- Is this a serious problem?
What else can go wrong?

- We want Murphy now, not later
Other stakeholders

- Who else should be encouraged to participate?
  - Multicast
  - IPsec
  - SCTP
  - DCCP
pmtdud {-request} @ietf.org