TCPng BOF

"TCPng: An Insurmountable Opportunity?"

Bob Braden

USC/ISI

Premise

- **TCP will have to be modified slightly for IP6.**
  New pseudo-header, expanded address format...
- **IP6 requires significant kernel mods in most sys's, and many applications will have to change.**
- **We should take this opportunity to update TCP while kernels are "opened up"**

But this premise may be debatable...

Definitions

For today:

"TCP4" means current TCP over IPv4.

"TCP6" means current TCP with minimal changes for IP6.

"TCPng" means anything else.

- **Some say: "anything else" => Protocol#(TCPng) != 6**
- **Must consider: TCPng/IP6 *and* TCPng/IPv4**

Suggested Changes in TCPng

A. Minimal Extension

- A1: New Pseudo-header
- A2: MSS revision
- A3: Max segment lifetime

B. Better Option Format

C. Headers aligned for 64-bit processors

D. Incorporate TCPLW extensions (RFC1323)

E. 32-bit ports

F. Miscellaneous small changes

G. Functionality extensions

DETAILS...

A1. New Pseudo-Header --

o Simply substitute 4B -> 16B ?

[Can preserve checksum
between TCP6/IP6 <--> TCP4/IPv4]

o Or, get along without pseudo-header ?

o Or, use TCP-id [ChristianH] ?
-- Random number
-- Passed in SYN segments
-- Used to ‘seed’ checksum.

o Protocol-independent pseudo-header [O’Malley]

DETAILS...

A2. Maximum Segment Size (MSS)

- o Not needed for IP6 (MTU Discovery required) (?)
- o Interoperation with TCP4??

DETAILS...

A3. Maximum Segment Lifetime (MSL)

o Don't leave home without it...

o Must increase MSL, assume routers engineered to limit Q time

(e.g., 30 secs x 30 hops ~ 1000 secs)

o PAWS (TCPLW) protects only same connection.

o Could extend PAWS to save [not cache] latest timstamp per association. *See Appendix B, RFC1323*.

DETAILS...

B. Better Option Format:

- o Make “Next-Option field”, optional sub-headers as in IP6.
- o Immediate goal: expand option space to handle Selective ACKs.

C. Headers Aligned:

- o Efficient processing at high speeds is an important goal for TCP.
- o But ... is 64-bit header alignment really important to performance,

  compared to data copies...?

DETAILS...

D. Subsume TCPLW

o See: *TCP Extensions Considered Harmful*, O'Malley & Peterson, RFC1263, Oct 1991.

o 32-bit windows *[improve functionality]*

o 64-bit sequence numbers

o Move timestamps from options into header.

E. 32-Bit Ports

o Does not help T/TCP -- must still shorten T-W state

DETAILS...

F. Miscellaneous small changes...

- o Urgent Ptr into options
- o Add DEC bit to ACKs.
- o Trailing checksum
- o ACK request bit
- o Record-Mark bit
- o Version field

Functionality Extensions (1)

o Renumbering 'hot' connections

For dynamic host re-configuration

o Mobility and multihoming support

See: draft-huitema-multi-homed-00.txt

o Reliable multicast

[!? We couldn't even agree on the problem!]

Functionality Extensions (2)

- Transactions

  See: RFC1644

- TMUX -- transport-layer multiplexing

- Security (?)

What are the right goals?

- Compatibility with TCP6/TCP4
- "Clean" design
- Efficient processing

(Riiiiiiiight...!)

Next Steps (?)

- Develop TCP6 [=> minimal changes]
- Start a WG to design TCPng ??