

Link ARQ issues for IP traffic

draft-ietf-pilc-link-arq-issues-00.txt

Phil Karn

PILC WG, 49th IETF,
Friday 16 December 2000.

presenting for

Gorry Fairhurst (gorry@erg.abdn.ac.uk)

Lloyd Wood (lwood@cisco.com)

who are currently stuck in Aberdeen and Edinburgh.

Audience participation!

Insert any pointless colourful corporate logos of your choice here
(and imagine some gradient-fill background, if you want)

Motivation of draft

Advice to Internet subnet designers draft doesn't discuss ARQ persistence issues for serial links.

Aim was simply to provide some text for *Advice* draft; by writing arguments down, we came up with this draft.

Focus of draft

ARQ retransmission persistency:

'How long a link is permitted to delay an IP packet, in an attempt to retransmit it reliably'.

Short summary

Low ARQ persistence is the lesser of two evils when you can't safely separate IP traffic into classes.

ARQ options

- *perfect* persistence - not recommended.
- *high* persistence - may be suitable, but has pitfalls; needs a smart sliding-window ARQ implementation.
- *low* persistence - always a safe bet for IP traffic; especially if you can't differentiate between flows.

Ordering

- preserving ordering is desirable, *but* not at the expense of blocking other traffic or leading to packet/frame drops by the link.
- *ideally* identify flows and preserve ordering *within* individual flows.
- needs clear information from IP layer (*flow id* ideal).

Important

You need to consider impact on the current flow *and* the impact on other flows also sharing the link.

Separating traffic into different flows

What it *is* isn't what it *does*

- peeking at transport headers and port no. tuples to identify flows may be a quick and tempting fix.

Assumptions about traffic requirements

- links guessing what flows do/want is questionable.

Traffic changes over time

e.g. contrast TCP use before/after web became popular:
How would that affect TCP performance across a persistent ARQ scheme designed for TCP use pre-HTTP?

IPSec can make peeking impossible. Tunnelling can make peeking harder - especially when tunnels aggregate flows!

Practical links vary in complexity

Real-world examples complicate analysis:

- more elaborate ARQ procedures
e.g. hybrid/adaptive ARQ.
- vulnerabilities to particular patterns of loss
caused by interference/noise.
- MAC persistency with resource allocation in the mix.
- the need to support link CoS / QoS...

**We can't prescribe for or describe all of these -
so we don't even try.**

...but we hope that:

...we show clearly what IP does in the face of ARQ.

...laying out ARQ issues well provides insight into effects on IP for future link designers - even designers faced with real-world complexity and things we don't consider.

We don't aim to categorise ARQ definitively, just to make people aware of what it can do to IP traffic.

Low-persistency link ARQ is safe and desirable

Higher persistency may have added benefits - in *some* cases.

Please send your comments on
draft-ietf-pilc-link-arq-issues-00.txt
to authors and *pilc* list!