Planning for Protocol Transitions
draft-iab-protocol-transitions-03

Dave Thaler <dthaler@microsoft.com>
Brian Trammell <ietf@trammell.ch>
There are different types of transitions

• Transition (n.): the process or a period of changing from one state or condition to another

• Technical transitions
  • IPv6, DNSSEC, https, IDN, EAI, ...

• Organizational transitions
  • IANA, web site host, ...

• Focus is on protocol transitions (though some principles will probably also apply to other kinds)
Some principles from RFC 5218

• Incentive: Easiest when benefits come to those bearing the costs
  • To succeed, the benefits must outweigh the costs at each entity
• Incremental Deployability: Backwards compatibility is easier
  • Easiest when changing only one entity still benefits that entity
• Total Cost: Don’t underestimate the costs of things other than the hardware/software
  • Operational tools and processes, training, accounting/billing, legal, etc.
• Extensibility: Design for extensibility so that things can be fixed up later
Example Cost/Benefit Graphs
Some Observations From ITAT Workshop (RFC 7305)

- Early-Adopter Incentives: Part of bitcoin’s strategy was extra incentives for early adopters
- Policy Partners: Policy-making orgs (RIRs, ICANN, etc.) can be important partners
Transition vs Co-existence

• Backwards compatibility means no significant difference
• Else either need transition (i.e. replacement) or co-existence (i.e., overlap period)
  • “Flag day” style transition increasingly impractical as number of entities involved increase
  • Coexistence increases costs during overlap period
  • An extended overlap period might result in further deployment of old mechanism

Any transition strategy for a non-backward-compatible mechanism should include a discussion of duration of overlap period (if any)
Backward compatibility, or lack thereof

• A translation/adaptation layer is often required if the mechanisms are not interoperable.
• Translation in the middle of the path can hamper end-to-end
• Translation at the end can be a resource issue if in a constrained node

Any transition strategy for a non-backward-compatible mechanism should include a discussion of where it is placed and a rationale.
What makes for a good transition plan?

1. Explanation of incentives for each entity involved
2. Description of phases
   • e.g.: pilot, co-existence, deprecation, removal
3. Timeline
4. Way to measure whether transition is succeeding
5. Contingency plan in case something goes wrong
6. Way to communicate plan to entities affected and incorporate feedback
TSV: an area in transition

- **MPTCP**
  - Purely end-to-end: incentives are relatively simple...
  - ...but lots of design work around option-meddling middleboxes

- **Explicit Congestion Notification**
  - Server support has passed 70%, default client support rolling out.
  - Development ongoing: TSVWG hummed for “reclamation” of ECT(1) for L4S.

- **QUIC**
  - Replaces TCP for some applications.
  - Plan is indefinite coexistence with TCP for fallback.
  - Discussion of interplay between protocol design and UDP impairment/blocking.