https://www.flickr.com/photos/52340452@N05/4916419197/in/photostream/

#612

9.2.2 requires ALPN capabilities beyond RFC7301

Section 9.2.2 places restrictions on the ciphers that are acceptable for a http/2 connection that are different to the acceptable ciphers for a https connection that may be offered over the same handshake.

To comply with 9.2.2, as server accepting an ALPN connection must either: a) influence the cipher selection to ensure an acceptable h2 cipher is selected; b) be informed of the cipher selected and if it is not acceptable then select http/1.1 instead of h2 as the protocol.

Neither of these capabilities are required of a RFC7301 compliant implementation. Specifically there is no requirement for an ALPN extension to be able to influence cipher selection, nor is there a requirement for an ALPN to make the cipher that will be selected available to the protocol selection.

“My API doesn’t support that” isn’t a technical issue

- we define protocols, not APIs, and can’t be constrained by any single API’s capabilities

However, if widespread lack of capability threatens success of the protocol, it *is* a technical issue.

- We need “running code” to prove our design.

- Firefox
- Chrome
- Twitter
- Akamai
- Google Front End
- node-http2
- …

Running Code

Does 9.2.2 introduce future risk?

1. Impact of Non-Conformance

- If TLS negotiation results in non-conforming suite, h2 fails
  - Client might retry (perf penalty)…
  - If not, server will presumably notice & fix

2. Cipher Sync

- New or deprecated cipher suites introduce uncertainty
  - “*Does my peer have the same list of acceptable suites as I do?*”
- Result: introduction/deprecation encounters friction

Straw-Man List-Informed Proposal

(SMLIP)

1. Make cipher suite requirements specific to TLS 1.2
2. Nominate a fixed list of suites for use with H2+TLS12
3. Keep the required interop suite (mandatory to *deploy*)
4. Clarify that cipher suite requirements apply to deployments, not impl
5. Relax requirement to generate INADEQUATE_SECURITY
6. Require support for TLS_FALLBACK_SCSV w/ TLS1.3+ (?)