draft-hoffman-dispatch-dns-over-https

Paul Hoffman

Patrick McManus

IETF 99, Prague

Purpose of the draft

- H2 is a more reliable transport for DNS queries and answers than DNS

- Web apps currently can't do real DNS queries; with this protocol, they could

- Make DNS information available to HTTP clients
  - Full DNS semantics, not just "give me the addresses of this host"
  - For example, DNSSEC

Best practice HTTP semantics

- Uses normal HTTP content negotiation mechanisms for selecting variants
- Purposely aligns itself with HTTP features such as caching, proxying, redirects, authentication, multiplexing, push, existing status codes and compression
- Default media formatting types for requests and responses

Non-requirements for the draft

- Supporting network-specific DNS64
- Supporting other network-specific inferences from plaintext DNS queries
- Supporting insecure HTTP
- Supporting legacy HTTP versions

How it looks (today)

- Uses DNS wire format, so the H2 client needs to have at least a simple marshaller
  - But protocol supports full DNS, including any extension
- Current draft shows GET and POST, which each has their own merits
  - WG may want to pick one or the other or both

GET

- :method = GET
  :scheme = https
  :authority = dnsserver.example.net
  :path = /.well-known/dns-query?
  content-type=application/dns-udpwireformat&
  body=q80BAAABAAAAAAAAA3d3dwdleGFtc
  GxlA2NvbQAAAQAB
  accept = application/dns-udpwireformat,
  application/simpledns+json

POST

- :method = POST
  :scheme = https
  :authority = dnsserver.example.net
  :path = /.well-known/dns-query
  accept = application/dns-udpwireformat,
  application/simpledns+json
  content-type = application/dns-udpwireformat
  content-length = 33

  <33 bytes represented by the following hex
  encoding>
  abcd 0100 0001 0000 0000 0000 0377
  77770765 7861 6d70 6c65 0363 6f6d 0000
  010001

The response

- :status = 200
  content-type = application/dns-udpwireformat
  content-length = 64
  cache-control = max-age=128

  <64 bytes represented by the following hex encoding>
  abcd 8180 0001 0001 0000 0000 0377
  77770765 7861 6d70 6c65 0363 6f6d 0000
  01000103 7777 7707 6578 616d 706c 6503
  636f6d00 0001 0001 0000 0080 0004 5db8 d822

Can also do different content-types

- Content negotiation is native to this model
- For example, JSON for the web apps
  - draft-hoffman-simplednsjson
  - Query: { "name": str, "type": str}
  - Response: { "code": int,
    "v4": [zero or more addresses as strs],
    "v6": [zero or more addresses as strs] }

Why DISPATCH

- There are enough interrelated parts that getting a variety of folks' attention would be good
- Should be easy to charter, finish, and test