Draft-lear-httpbis-srvinfo-rr

Eliot Lear
Problem

- We need a way to upgrade from 1.1 to 2.0
- Some would like to eliminate the first round trip to avoid “Upgrade:”
- Let’s not simply move the round trip
MUST do Upgrade

- What is described here is an optimization, and not a full scale alternative
Design goals

- Application protocol version must be discoverable within the DNS
- Transport protocol (e.g., tcp, sctp) information must be discoverable within the DNS
- Performance of the application must not be impacted
- Multiple instances and versions of http should be supported on the same system.
- No new URIs
  (Bus side problem)
Possible Approaches: SRV

- Used by many applications
- Allows for an additional level of redirection
- Target may or may not be in the same zone as the QNAME
  
  _http._tcp.www.example.com in srv 10 10 49080 foo.bar.com

- _tcp.example.com is often a separate zone for load balancing purposes
  
  This may complicate domain configurations (e.g., split DNS, etc)
SRV Packets

For record: _http._tcp.example.com in srv 20 20 80 www.example.com

_nameserver for _tcp.example.com

_client resolver

_http._tcp.example.com SRV?

_http_tcp.example.com in srv 20 20 80 www.example.com
www.example.com in a 10.1.1.1
SRV Packets

For record: _http._tcp.example.com in srv 20 20 80 www.example.com

Client resolver

_http._tcp.example.com SRV?

_nameserver for _tcp.example.com

_http._tcp.example.com in srv 20 20 80 www.example.com

www.example.com AAAA?

www.example.com in a 10.1.1.1
SRV Packets

For record: _http._tcp.example.com in srv 20 20 80 www.example.com

Client resolver

_nameserver for _tcp.example.com

_http._tcp.example.com SRV?

_http_tcp.example.com in srv 20 20 80 www.example.org

www.example.org AAAA?

www.example.org in a 10.1.1.1

Name server for www.example.org
NAPTR and URI Records

• NAPTR:
  Very powerful search and replace mechanism
  Builds on SRV
  Allows for transport protocol discovery
  Does not provide protocol version information

• URI
  Maps a domain to multiple URIs
  Lacks protocol version information
Running a race

- Used by Happy Eyeballs to determine best accessible IP versions
- Doesn’t provide protocol information on its own
- May be necessary to reduce latency
- Certainly advance queries will help
**SVCINFO Resource Record**

- domain TTL Class SVCINFO InstanceId Priority Proto Port Version
- No additional indirection on QNAME
  - No risk of required sequential lookups
- Priority, Protocol, Port, Version self-explanatory
- InstanceID is used to index against the port in the URI
  - Two records with matching InstanceID mean that the same service is described by both records for a given name
For record: _http._tcp.example.com in srv 20 20 80 www.example.com

example.com SVCINFO?
Example.com A?

eample.com in svcinfo 1234 10 tcp 80 1

eample.com in a 10.1.1.1

Nameserver for example.com

Client resolver
Qualities of DNS Records

- They are cached – sometimes
- Clients don’t know where zone cuts are
- DNS is one of three approaches prior to connection to provide version information
  - A new URI
  - Specification as part of HTML
Questions

• Is the optimization worth it for one protocol turnaround?
• Is there an interest in other transport protocols for HTTP?
• Should we combine proto, version into a “profile”?