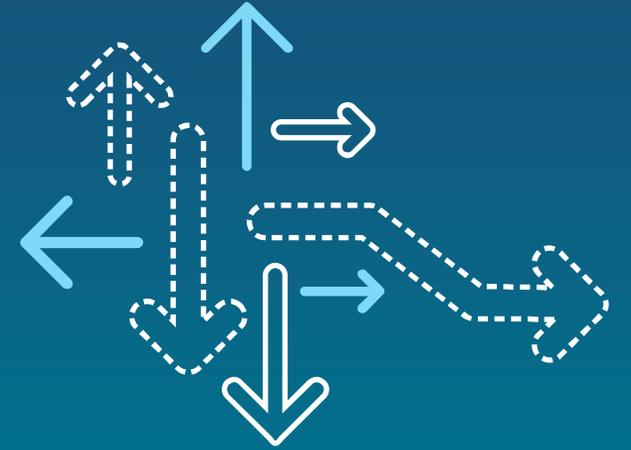


Optimizing DNS-SD Query

draft-aggarwal-dnssd-optimize-query-00

Ashutosh Aggarwal (Qualcomm)
aaggarwa@qce.qualcomm.com

dnssd WG meeting, IETF90, Toronto, July 24th 2012



Before I start..

- I would like to acknowledge Dave Thaler for his guidance and insightful comments while developing this contribution
- I apologize I can't be there in person due to last minute airline cancellation

Problem Statement

- The current DNS-SD query mechanism may not scale well when the number potential responders that match the service-name based query increase
 - IoE (IoT) era will get us there sooner than we think!
- What does not scale well mean?
 - Higher latency in finding the desired service instance from the querier's point-of-view
 - Increased traffic on the network due to all the responses and subsequent service negotiation
 - Increased processing for the querier and all potential responders
- The problem scope encompasses DNS-SD discovery using mDNS or unicast DNS

Sample Use Cases

Problem Illustration

- A client application is looking to find color printers on the local network
- A lighting application needs to discover lighting fixtures or bulbs from a given manufacturer before establishing a session with each device to control the fixtures

Background

What can be do within the current scope of DNS-SD?

- Place DNS TXT records in the additional section of the DNS response message
 - Server has to “guess” what might be useful to the client application
 - Client application might have to establish a connection anyway
- Use subtype as part of the question
 - 63 octets or less
 - Either part of service protocol in question and/or “notes” field with service registration with IANA
 - Intended for two level scenarios
 - Used to discover a subset of services within a larger set
- Can we do something to optimize the discovery process without requiring DNS message changes?

Proposed Change

- A client application can query for service name and key/value pairs defined within the scope of that service
 - Key/value pairs are sent as DNS TXT records in the additional section of the DNS-SD query
 - Client application can choose not to invoke this capability i.e. continue to send service name in the query
- Semantics:
 - If multiple keys are present in the DNS TXT record, they are AND'ed
 - If multiple DNS TXT records, they are OR'ed
- Small set of additional keys will be very valuable in narrowing down the search context
 - See examples on the next slide

Realization of the Proposal

Revising Two Use Cases

- A client application looking for color printer can add color=true in the DNS TXT record as part of the additional section of the query
- A lighting application looking to discover bulbs by a certain manufacturer (such as Philips), can add the DNS TXT record in the additional section of the query with manuf=Philips
- Key names are for illustrative purposes only

Deployment Considerations

- Analyze the behavior of existing mDNS responder and unicast DNS with DNS TXT record in the DNS-SD query
- mDNS
 - If TXT record is not recognized,
 - response will be sent only if there is a match for the service name
 - degenerates to service name based query only
- Unicast DNS
 - DNS will respond strictly based on the service name in question

API Considerations

- Android and iOS provide DNS-SD APIs
- Additional impact would be for the client application to initiate the query with specific key/value pairs
 - Should be designed as optional parameter

Questions/Comments

- There was one comment received stating that mandatory and optional keys should be defined within the scope of service protocol specification
 - Author agrees with that comment
- Other comments?