# But it's Just Over There!

# (aka Scaling up DNS-based Service Discovery)

Tim Chown
University of Southampton
tjc@ecs.soton.ac.uk
(co-chair, IETF dnssd WG)

# So what is the problem?

- DNS-based service discovery (RFC 6763) was designed for zero configuration networking within a link
  - Works with link-local addressing and naming (mDNS, under .local, RFC 6762), and DNS TXT/PTR/SRV records
  - Uses link scope multicast on well-known addresses
  - Multicast is thus not forwarded by routers
  - May be user to device, or device to device, with an ever increasing variety of services for which discovery is desirable

- Great when devices are all on the same link
  - But increasingly they are not, e.g., devices in a campus network, or emerging multi-link home networks
  - Frustrating when you can see the device you want to discover!

# Aside: mDNS at Coseners

- Sampled 30 minutes (ish) of traffic yesterday
  - Using tcpdump on eduroam SSID for port 5353 UDP
- 5,670 mDNS messages
  - Includes 2,753 IPv6 messages
  - Many messages replicated over both IPv4 and IPv6 protocols
- 63 different devices
- 96 different services
- Consider:
  - How would this scale to a campus scenario of 100's of links?
  - What might the privacy issues be?

# mDNS talkers by messages

120 joels-macbook-2012.local
96 iain-phillipss-ipad.local
91 stewart-bryants-iphone.local
63 cheesy-mac.local
60 Xiao-Gaimato-iPad.local
56 David-Rolfes-iPod.local
55 saras-iphone.local
54 lbsml0204889.local
51 fyzhangteki-ipad.local
47 wopr.local
47 chenglins-iPhone.local
44 jamies-iphone.local
39 macintosh-2.local
38 tims-macbook-pro-3.local
32 stephanies-ipad.local
31 Temi-PC.local
25 chromium.local
24 vinge.local
24 macbot-4000.local
22 mircos-macbook-air.local
22 Fabios-MacBook-Pro.local
22 Alex.local
21 OXCEGLAP03.local
21 Eleni-Mykoniatis-MacBook-Air.local
20 ph-pnslap2.local
20 Noors-MacBook-Pro.local
17 nanak.local
16 mateng165.local
16 iain-mbp.local
16 fusion91.local
15 alex-laptop.local
14 neil-bournes-macbook.local
14 macbook-pro-de-laura-isabelle.local
14 Johns-MacBook-Air.local
12 T520.local
11 pulse.local
11 pc-metal.local
11 janes-macbook-air.local
11 Mircos-MacBook-Air.local
10 adms121131-lt.local
10 OXCEGLAP03-2.local
10 Mac-de-Ju.local
8 applematoMacBook-Air-3.local
8 Macintosh-2.local
7 aidans-macbook-pro.local
6 Giannis-MacBook.local
5 Brians-MacBook-Air.local
4 francesco-hautmanns-macbook-air.local
4 Blackhawk. Local
4 Steve-Uhligs-Macbook.local
3 fyzhangteki-iPad.local
3 Minoss-MacBook-Pro.local
3 MacBook-Pro-de-Laura-Isabelle.local
3 LBSML0204889.local
2 zhaoruixinde-iPhone.local
2 patti-HP-Pavilion-15-Notebook-PC.local
2 legend.local
2 Marmot.local
2 Joels-MacBook-2012.local
1 klynphybrisacuk.local
1 Nanak.local

# Services by messages

- Includes:
  742 _googlecast._tcp.local
  711 _apple-mobdev2._tcp.local
  294 _raop._tcp.local
  150 _apple-mobdev._tcp.local
  146 _afpovertcp._tcp.local
  117 _ssh._tcp.local
  115 _airplay._tcp.local
  78 _airport._tcp.local
  32 _rfb._tcp.local
  20 _ipp._tcp.local
  …..

(Plus many more not listed)

# Scalable DNS-based service discovery?

- So, what would we like to build?

- We'd like an extended DNS-based service discovery mechanism
- Applicable to personal, home, campus, enterprise and other network scenarios
- Backwardly compatible with mDNS/DNS-SD
  - And supporting incremental deployment
- May be zero configuration (e.g., in the home), or may benefit from additional administrator configuration (e.g., on a campus)

- The IETF dnssd WG is currently working to agree requirements
  - See draft-ietf-dnssd-requirements-03
  - Hoping to finalise at IETF 90 later this month

# Requirements?

- Be able to determine the "scope" in which to advertise and discover services
  - And be able to choose/select when there are multiple options
- Include topologically independent "scopes"
  - In particular be able to use physical (vs network) proximity
- Operate across non-contiguous links
- Be efficient in all networks, esp. lossy/wireless networks
- Consider energy consumption, nodes in a low power/sleeping state
- Be scalable to > thousands of nodes
- Reflect reality – make new information available in a timely fashion, and likewise remove stale information
- Require no changes to underlying network
- …

# General solution approaches?

- Just define a new "xmDNS" that operates by extending mDNS to instead or also use site scope multicast
  - Not scalable, not interoperable with existing mDNS deployments
- Make the network flat
  - But a campus may have 20,000+ devices…
- Hybrid Unicast/Multicast DNS-Based Service Discovery
  - Leaf routers act as proxies to their local links
  - Includes consideration of duplicate names, and change notification
  - Initial ideas described in draft-cheshire-dnssd-hybrid-01
- … ?

# Components to standardise?

- A mechanism for proxying mDNS/unicast DNS
  - Perhaps in a similar manner to a DHCP relay
  - Want to be able to both publish and resolve services
- Publishing to unicast DNS
  - Nodes may not be able to publish directly
  - Perhaps take an approach like draft-ietf-dhc-addr-registration-04, but DHCP is not universally available
- Service discovery "zone" enumeration
  - How to discover / select the zone(s) to advertise/discover in?
- Change notification mechanism
  - Might be a use case for DNS Long-Lived Queries
  - See the old draft-sekar-dns-llq-01 draft
- And we should at least document the naming issues
  - Unicode vs IDNA – see draft-sullivan-dnssd-mdns-dns-interop

# How to join in

- Take a look at the IETF dnssd WG Charter:
  - http://datatracker.ietf.org/wg/dnssd/charter/
- Check the status pages for existing dnssd drafts:
  - http://tools.ietf.org/wg/dnssd/
- Read and comment on the requirements draft (in WGLC):
  - http://tools.ietf.org/html/draft-ietf-dnssd-requirements-03
- Mail list:
  - dnssd@ietf.org
  - To join: https://www.ietf.org/mailman/listinfo/dnssd
  - All input very welcome!