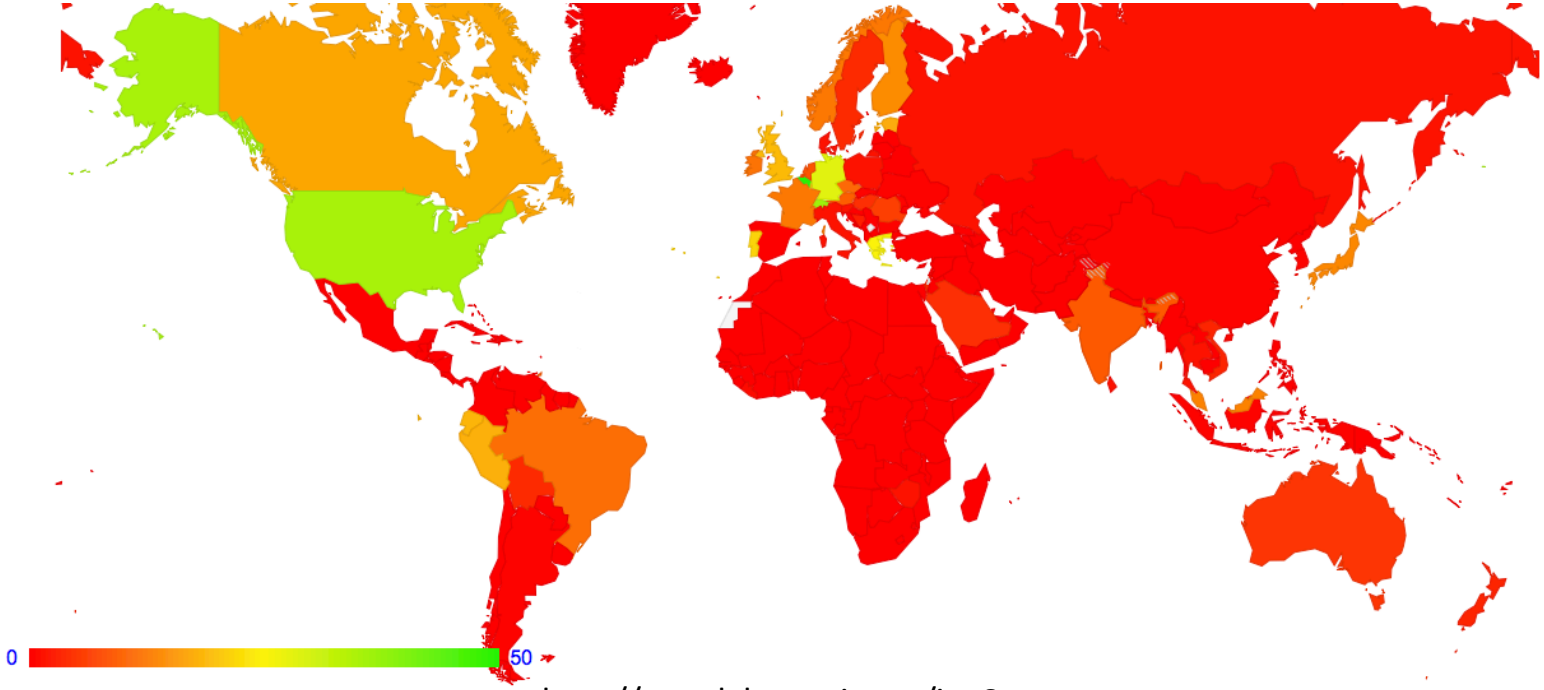
IPv6 and the DNS

Geoff Huston

APNIC

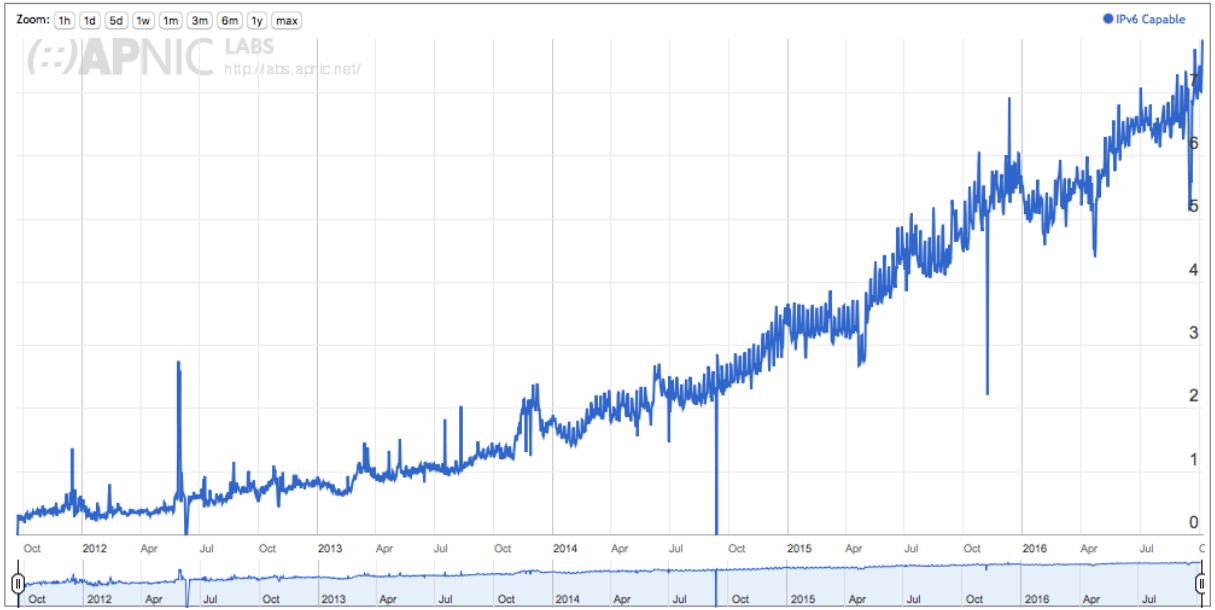
October 2016

IPv6 Adoption



http://stats.labs.apnic.net/ipv6

IPv6 Adoption



http://stats.labs.apnic.net/ipv6

What does it mean?

What are we saying when we say that IPv6 adoption has reached 7% of the Internet?

One way of interpreting this data is that if you hosted a web service on V6 only, some 7% of the Internet's user population could access this service

We think.

What we don’t measure

The Internet is a whole lot more than the web!

But all we measure and all we talk about is web-based metrics

What about other components of the Internet environment?

One critical component is the DNS

So how are we doing with IPv6 in the DNS?

IPv6 DNS questions

- DNS is a multi-faceted environment, populated by authoritative name servers who publish information, and client resolvers who pose queries
- And there is a distinction between whether the query is about resolving a name into an IPv6 address and whether its possible to use IPv6 to pass the query to the name server

- That's a lot of material to cover in a single presentation
- So let's pick one question and dig deeper…

Today’s DNS IPv6 questions

How much of the DNS resolution infrastructure is IPv6 capable?

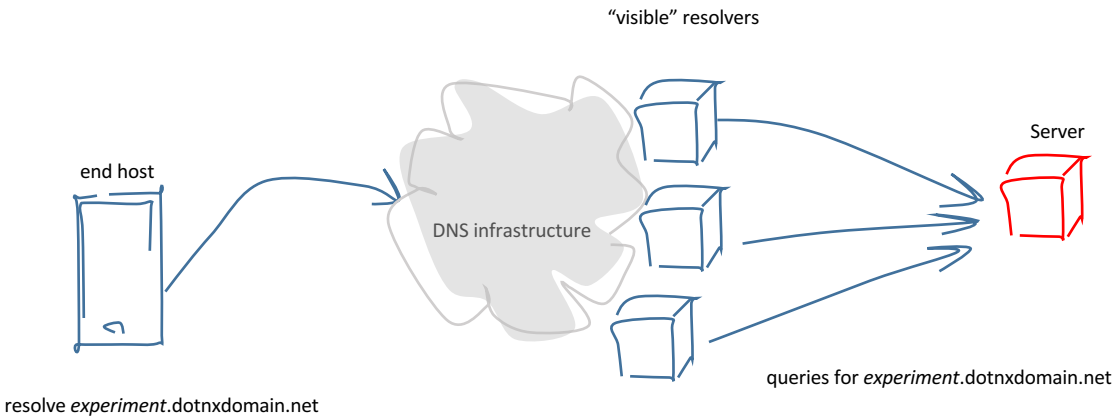
This is a deceptively hard question!

- The DNS is a meta-stable, non-deterministic, chaotic system that still, surprisingly, manages to operate in a manner that appears to be relatively fast, relatively efficient and mostly accurate!

- But underneath the surface a lot is going on:
  - The local resolver function has re-query timers and a locally defined set of resolvers
  - Resolvers themselves have timers and may use forwarders
  - Resolvers may be part of a server farm with active load balancing

- All the authoritative name server sees is a set of queries coming from “visible” resolvers
  - The interactions internally between the local host and its resolvers and the chaining of queries is largely opaque

A view of the DNS infrastructure



Our Approach

- It’s hard to instrument all parts of the Internet and make sense of the data streams
- Our approach is to seed a known event in end hosts that are intended to cause DNS resolution activity, and instrument the authoritative DNS server
- We infer aspects of the behaviour of the DNS from the transactions we see at the authoritative name server

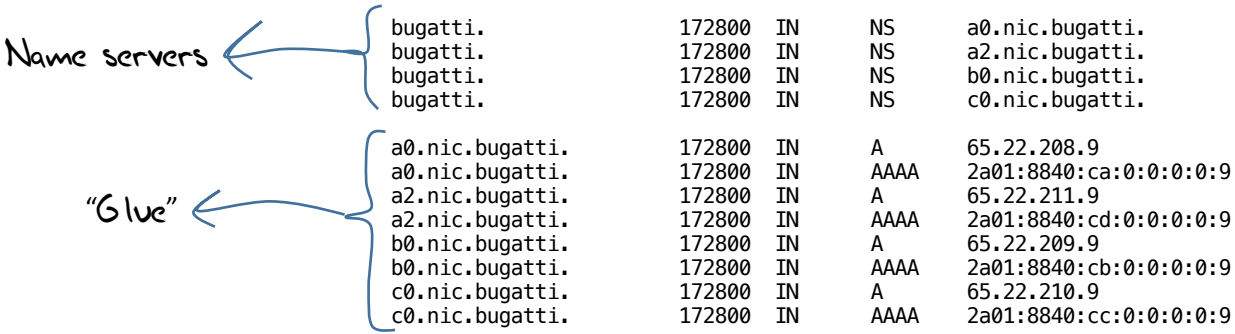
Our approach

- We use the Ad platform to enrol end points to attempt to resolve a DNS name
  - The DNS name is served from our authoritative servers
  - Each endpoint is provided with a unique name string (to eliminate the effects of DNS caching)
  - Each DNS name contains a name creation time component (so that we can disambiguate subsequent replay from original queries)
  - We have structured the measurement name space so that the behaviour is visible solely in the DNS (it does not rely on a subsequent web fetch to show that the response was received)

Name Delegation and “Glue”

- When a name is delegated, the “parent” zone normally includes the IP address of the delegated zone’s name servers as additional information

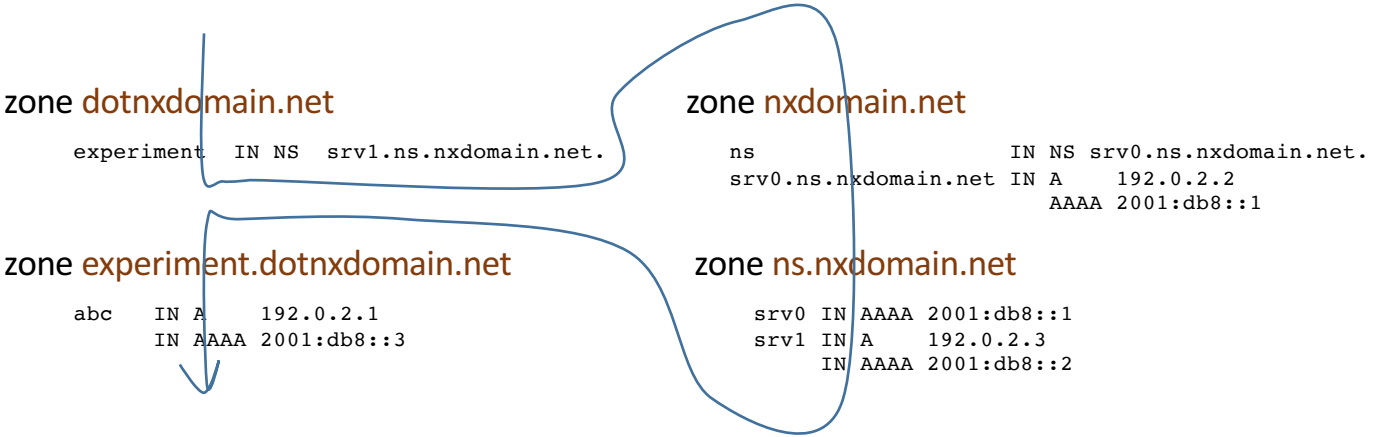
For example, here’s a snippet from the root zone for the delegation of the gTGLD “.bugatti”



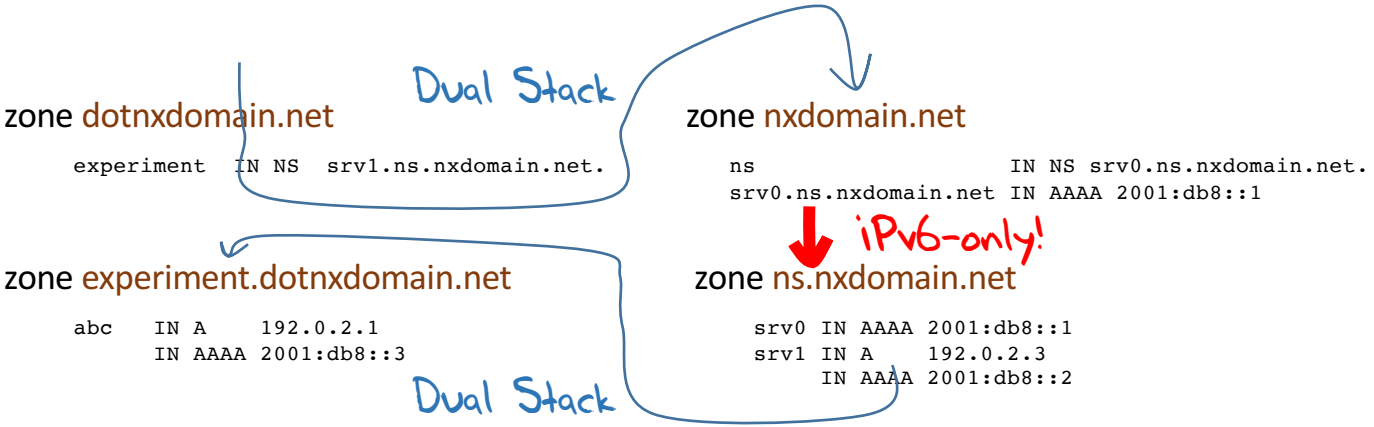
“Glueless” Delegation

- ”Glue” records provide helpful hints to resolvers, but they are not mandatory, nor are they authoritative
- If a resolver performing a top-down resolution sequence encounters a delegation without glue then it pauses the resolution process of the original name and commences resolution of the name server name.
- If this secondary resolution succeeds then it resumes the resolution process of the original name

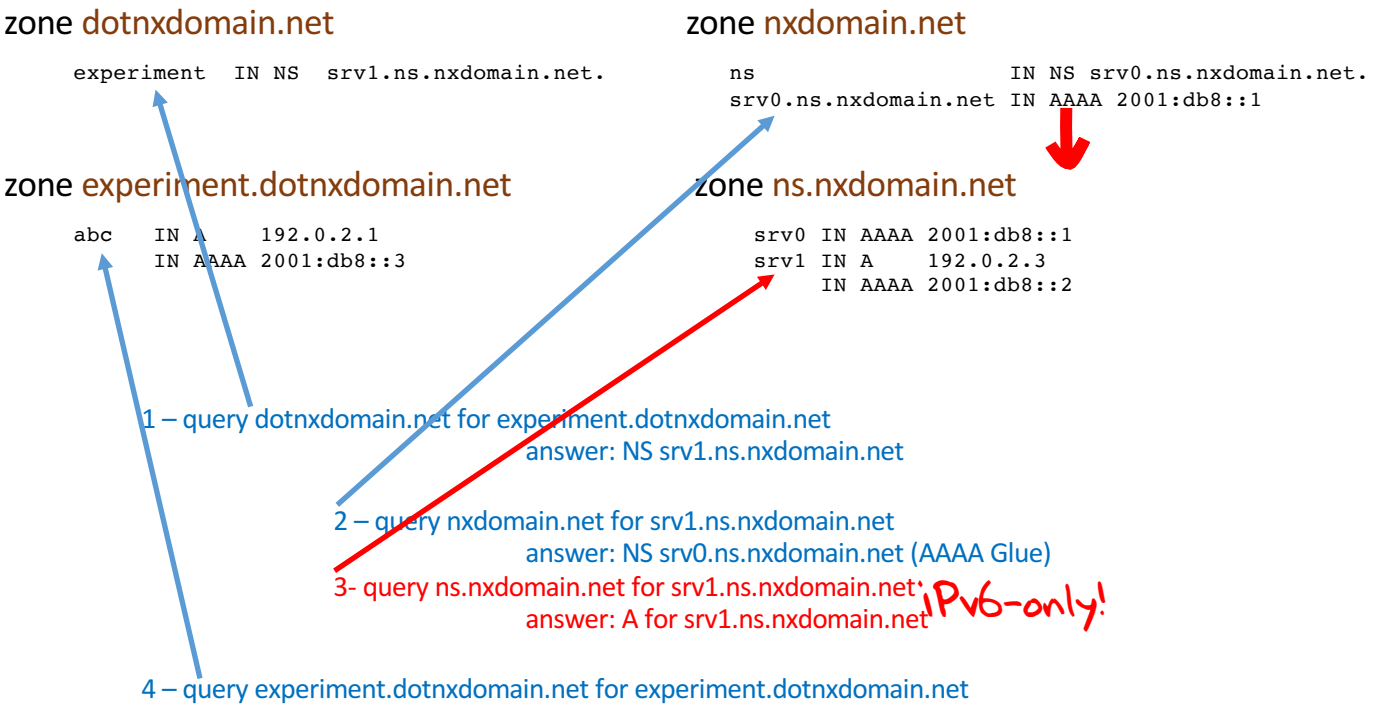
”Glueless” Delegation



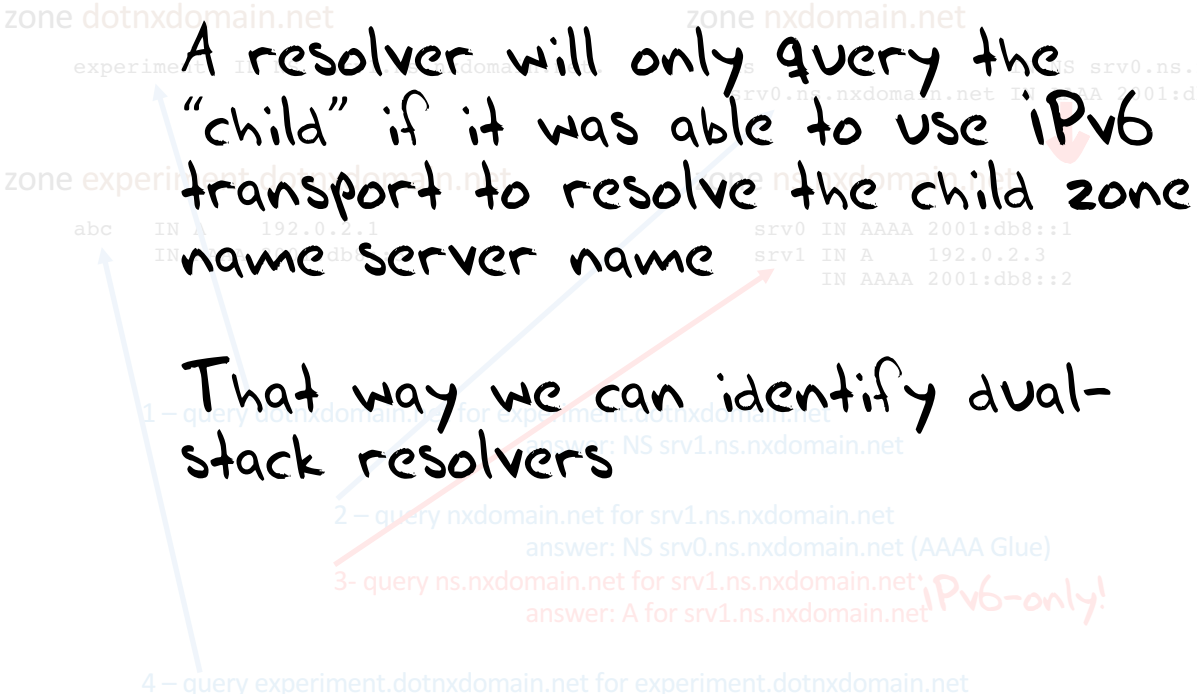
We can use this…



We can use this…



We can use this…



A resolver will only query the "child" if it was able to use IPv6 transport to resolve the child zone name server name

That way we can identify dual-stack resolvers

The measurement

- The Ad campaign ran across July - August 2016 running between 5M and 10M ads per day
- We collected some 400M results spanning most of the Internet

“Visible” Resolver Totals

345,394 unique resolvers asked the auth server for the “parent” zone

268,218 of these resolvers appear to be V4 only (did not pose the IPv6 query to the “sibling” server)

59,372 resolvers asked the “parent” query using IPv4, and asked the “sibling” query using IPv6

77,812 resolvers in total queried the parent, sibling and child servers

i.e. some **22%** of visible resolvers are capable of using IPv6 to make DNS queries

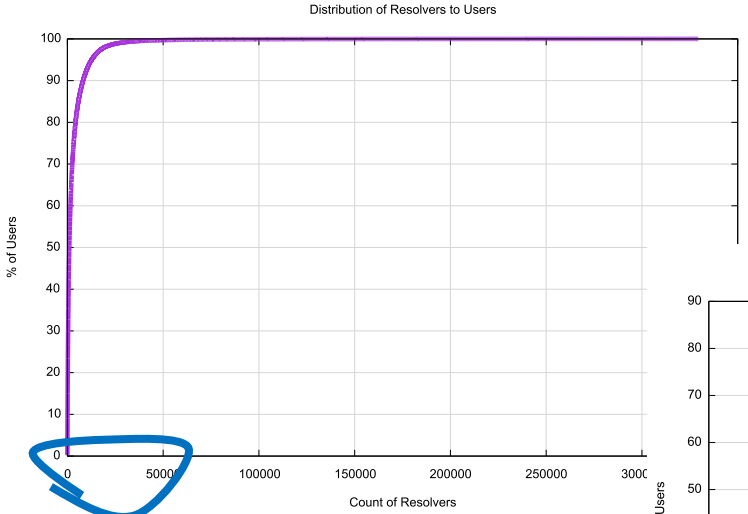
“Visible” Resolvers

22% of visible response are capable of performing queries using IPv6 transport

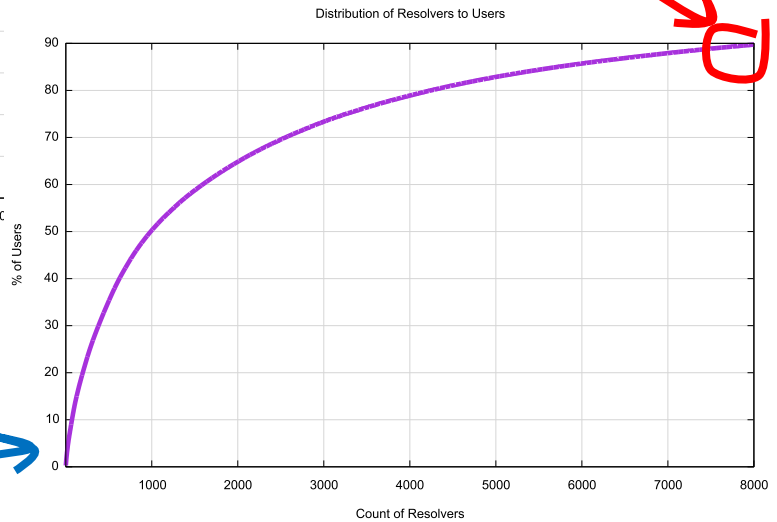
But maybe there is a difference between counting resolvers and counting the users who use resolvers

i.e. what differences exist when looking at the intensity of use of individual resolvers?

All resolvers might be equal, but some resolvers are more equal than others!



8,000 distinct IP addresses (2.3÷ of all seen IP addrs) for resolvers serve 90÷ of all experiments



IPv6 Usage Results by Query

194M unique experiment ids asked the auth server for the “parent” zone

122M (63%) did NOT ask the “sibling” server for the NS zone using IPv6

2.9M (1.5%) did NOT ask the “child” server for the target name

68.5M (35%) appeared to complete the DNS resolution task

i.e. some **35%** of experiments were able to use IPv6 to resolve a DNS name !!

IPv6 Usage Results

- While some 22% of visible resolvers are IPv6-capable, it appears that around 35% of users direct these queries to these IPv6-capable resolvers
- While this is visible using an IPv6-only glue server, what is the query profile when we use a Dual Stack server?
  - i.e. Do Dual Stack capable DNS resolvers prefer to use one protocol or the other?

V6 Capable vs V6 Preference

25% of experiments pass queries to resolvers who are IPv6 capable

Out of 3,113M queries made in this experiment to the Dual Stack ”parent” server, some 352M queries were over IPv6

i.e. **11%** of query sequences pass queries to resolvers who are Dual Stack capable

If the choice of protocol was random, then this number would be **17%**, so this data suggests that there is some slight inherent bias in protocol selection to use IPv4 by resolvers when the server is advertising Dual Stack reachability

This may be due to the local selection of resolvers, where a user may be configured with IPv4-only and dual-stack recursive resolvers

Which resolvers are they using?

Top 25 Visible IPv6-capable resolvers, grouped by Origin AS, ranked by relative use by end users

AS15169 31.9% GOOGLE - Google Inc., US United States of America
AS7018 13.5% ATT-INTERNET4 - AT&T Services, Inc., US United States of America
AS7922 11.5% COMCAST-7922 - Comcast Cable Communications, LLC, US United States of America
AS36692 3.4% OPENDNS - OpenDNS, LLC, US United States of America
AS8151 2.7% Uninet S.A. de C.V., MX Mexico
AS17676 2.4% GIGAINFRA Softbank BB Corp., JP Japan
AS4134 1.7% CHINANET-BACKBONE No.31,Jin-rong Street, CN China
AS28573 1.6% CLARO S.A., BR Brazil
AS9498 1.6% BBIL-AP BHARTI Airtel Ltd., IN India
AS3320 1.4% DTAG Internet service provider operations, DE Germany
AS2516 1.2% KDDI KDDI CORPORATION, JP Japan
AS6147 1.1% Telefonica del Peru S.A.A., PE Peru
AS18881 1.0% TELEFONICA BRASIL S.A, BR Brazil
AS22773 1.0% ASN-CXA-ALL-CCI-22773-RDC - Cox Communications Inc., US United States of America
AS55836 1.0% RELIANCEJIO-IN Reliance Jio Infocomm Limited, IN India
AS55644 0.9% IDEANET1-IN Idea Cellular Limited, IN India
AS6713 0.9% IAM-AS, MA Morocco
AS4713 0.9% OCN NTT Communications Corporation, JP Japan
AS6128 0.9% CABLE-NET-1 - Cablevision Systems Corp., US United States of America
AS20115 0.8% CHARTER-NET-HKY-NC - Charter Communications, US United States of America
AS3352 0.8% TELEFONICA_DE_ESPANA , ES Spain
AS852 0.8% ASN852 - TELUS Communications Inc., CA Canada
AS22394 0.5% CELLCO - Cellco Partnership DBA Verizon Wireless, US United States of America
AS6799 0.5% OTENET-GR Athens - Greece, GR Greece
AS15557 0.4% LDCOMNET , FR France

A word of caution

- Adding IPv6 to a resolver is not without its element of risk in terms of resolution performance
- The problem lies in the issues with large DNS responses, IPv6 fragmentation and IPv6 Extension header handling
- Dropped IPv6 responses cause resolver timeouts triggering re-queries, extending resolution time

IPv6 Response Reliability

- In the context of the “glueless” setup, the resolver will query for the target name if and only if it can receive a response to the IPv6-only query for the address of the NS name
- We tested 3 NS response sizes: 361, 1156 and 1425 octet responses
- We used a local MTU setting of 1500 octets, reducing the level of source-initiated IPv6 fragmentation

IPv6 Failure Behaviours

Repeated queries with large EDNS0 buffer size

- Indicative of the resolver unable to receive the IPv6 response

Repeated queries with no EDNS0 buffer size

- Where the UDP response is a Truncated DNS payload. This is indicative of either being unable to receive the IPv6 DNS response or being unable to initiate a TCP session

Completion Rate

What proportion of experiments completed the IPv6 NS lookaside operation after making a query to the “sibling” Name Server by making a query to the target name?

Size	completion/sibling lookup	Rate
361:	68M/71M	96%
1125:	68M/71M	96%
1425:	68M/71M	96%

We used a local MTU setting of 1500 octets!

IPv6 and the DNS?

In resolution infrastructure we seem to be further along the transition than the web: 35% of users pass their queries to resolvers that are capable of using IPv6, and about half of that show a preference for using IPv6

In terms of reliability, as long as you take some care in the configuration*, this should be just fine!

* Try and avoid IPv6 fragmentation by using a local UDP MTU size of 1500 octets, and ensure that there are no local ICMP6 filters
  At the same time use an IPv6 TCP MSS size of 1220 octets to avoid PTMU blackholing

Thanks!