AS112-bis.

ggm@apnic.net
$ dig +short ns ip6.arpa
Can we have some context please?

$ dig +short ns ip6.arpa
a.ip6-servers.arpa.
c.ip6-servers.arpa.
b.ip6-servers.arpa.
e.ip6-servers.arpa.
f.ip6-servers.arpa.
d.ip6-servers.arpa.
$
Be more specific

$ dig +short e.ip6-servers.arpa.
Be more specific

$ dig +short e.ip6-servers.arpa.
202.12.29.59
$
$
And you are..

$ dig +short e.ip6-servers.arpa.
202.12.29.59
$
$
And you are..

$ dig +short e.ip6-servers.arpa.
202.12.29.59
$ dig +short -x 202.12.29.59
cumin.apnic.net.
$
And you are..

$ dig +short e.ip6-servers.arpa.
202.12.29.59
$ dig +short -x 202.12.29.59
cumin.apnic.net.
$
And you are..

$ dig +short e.ip6-servers.arpa.
202.12.29.59
$ dig +short -x 202.12.29.59
cumin.apnic.net.
$ ggm@apnic.net
Cut to the chase

- Lots of stupid DNS
- IPv6 brings new kinds of stupid DNS
- Time to re-work AS112 and delegate some IPv6 reverses to AS112
Cut to the chase

• Lots of stupid DNS
• IPv6 brings new kinds of stupid DNS
• Time to re-work AS112 and delegate some IPv6 reverses to AS112
• Now lets go have a curry
Cut to the chase

• Lots of stupid DNS
• IPv6 brings new kinds of stupid DNS
• Time to re-work AS112 and delegate some IPv6 reverses to AS112
• Now lets go have a curry pizza & beer
The Long Version

- IP6.ARPA
  
  
  $ dig +short aaaa wattle.rand.apnic.net.
The Long Version

• IP6.ARPA
  
  $ dig +short aaaa wattle.rand.apnic.net.
  2401:2000:6660::2
The Long Version

• IP6.ARPA

  $ dig +short aaaa wattle.rand.apnic.net.
  2401:2000:6660::2

  $ dig +short -x 2401:2000:6660::2
The Long Version

• IP6.ARPA

  $ dig +short aaaa wattle.rand.apnic.net.
  2401:2000:6660::2

  $ dig +short -x 2401:2000:6660::2
  wattle.rand.apnic.net.
The Long Version

• IP6.ARPA
  
  $ dig +short aaaa wattle.rand.apnic.net.
  2401:2000:6660::2
  
  $ dig +short -x 2401:2000:6660::2
  wattle.rand.apnic.net.

• When its done right, its simple.
The Long Version

• IP6.ARPA

   $ dig +short aaaa wattle.rand.apnic.net.
   2401:2000:6660::2

   $ dig +short -x 2401:2000:6660::2
   wattle.rand.apnic.net.

• When its done right, its simple.
• A brief reminder whats under the hood...
The Longer Long Version

$ dig -x 2401:2000:6660::2

; <<>> DiG 9.6.0-APPLE-P2 <<>> -x 2401:2000:6660::2
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 26395
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 4

;; QUESTION SECTION:
;2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.6.6.6.0.0.0.2.1.0.4.2.ip6.arpa. IN PTR

;; ANSWER SECTION:
2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.6.6.6.0.0.0.2.1.0.4.2.ip6.arpa. 3589 IN PTR wattle.rand.apnic.net.
Problems?

• 32 zone-cut points, potential (re)delegation boundaries
  – Long strings == keystroke errors
  – “Looks like ‘Too much work’ to me” problem
    • Low compliance
This is not my problem.

• 32 zone-cut points, potential (re)delegation boundaries
  – Long strings == keystroke errors
  – “Looks like ‘Too much work’ to me” problem
    • Low compliance
This is my problem

• 32 zone-cut points, potential (re)delegation boundaries
  – Long strings == keystroke errors
  – “Looks like ‘Too much work’ to me” problem
    • Low compliance

• Negative Answers cost more
This is my problem

• 32 zone-cut points, potential (re)delegation boundaries
  – Long strings == keystroke errors
  – “Looks like ‘Too much work’ to me” problem
    • Low compliance

• Negative Answers cost more
  – There are lots of Negative-Answer questions
This is my problem

• 32 zone-cut points, potential (re)delegation boundaries
  – Long strings == keystroke errors
  – “Looks like ‘Too much work’ to me” problem
    • Low compliance

• Negative Answers cost more
  – There are lots of Negative-Answer questions
  – Like IPv6 address types not expected to be seen in the global DNS but which are being looked up
Negatives cost more
Negatives cost more?

• NXDOMAIN on average is 2-3x longer than OK
• DNSSEC makes this worse
  – Additional RRSET/NSEC sections in reply
  – Answer now approaching 1kb per query.
• How bad can this get?
  – Depends how much IPv6, and
  – what kind(s) of stupid questions get asked
    • dunnit?
What kind of Questions get Asked?
Too many to count ..........

- Link Local
- Site Local
- Multicast
  - Link and site-local multicast
- Unique Local Address (ULA)
- Tunnelled
  - 6RD, 6to4, Teredo
- Un-delegated in reverse,
  - but otherwise global unicast
Too many to count (ok 6)

- Link Local
- Site Local
- Multicast
  - Link and site-local multicast
- Unique Local Address (ULA)
- Tunnelled
  - 6RD, 6to4, Teredo
- Un-delegated in reverse,
  - but otherwise global unicast

What we get in IPv4 right now
AS112 is designed to mitigate

New in IPv6
Stop whining, give me some numbers
A typical day in 2011

transport
  v4: 369,917,141
  v6:  6,605,575

v6/v4 ratio: 0.0178

PTR: 341,620,046
valid PTR: 341,271,155
invalid PTR: 322,778
odd PTR: 25,827
null PTR: 286

valid PTR: 341,271,155
in-addr: 317,287,473
ip6.arpa: 23,983,682

ip6/in-addr ratio: 0.0756

1.78% of query carried in V6

7.56% of query about V6
A typical day in 2011

transport
  v4: 369,917,141
  v6: 6,605,575

v6/v4 ratio: 0.0178

PTR: 341,620,046
  valid PTR: 341,271,155
  invalid PTR: 322,778
  odd PTR: 25,827
  null PTR: 286

in-addr: 317,287,473
ip6.arpa: 23,983,682

ip6/in-addr ratio: 0.0756

1.78% of query carried in V6
7.56% of query about V6

5% NXDOMAIN = Negative Answer Required
7.56%? What’s the problem?

• Risk management is about **planning for the worst case**
  – In **this** case, the worst case is “IPv6 succeeds”
• The volume of queries seen in IPv4 therefore become the volume of queries seen in IPv6
  – Plus, all the new stupid queries
  – Most of which are NXDOMAIN
• So, how many stupid queries do I see?
Drilling down into stupid queries
Atmel has announced a range of AVR wireless microcontroller (MCU) devices targeting wireless applications such as Zigbee and IPv6/6LoWPAN.

... Atmel’s picopower technology offers ultra-low power consumption to enable longer battery life for wireless Zigbee applications, including smart energy, building automation, telecom and health care.
At the Cisco Live! conference this week, Cisco and Nivis demonstrated an operational wireless IP mesh network using the low-power IPv6 protocol, dubbed 6LoWPAN. The demo relied on Nivis wireless sensors and routers to link a parking meter with several streetlights, a sensor ring in a parking space, and what was described as a Cisco cell phone.

The demo then used this network to alert a driver of an available parking space and to send another message that the meter had expired. Cox notes that the setup could also be used to relate any number of other driving or traffic related messages: to tell security guards to turn on parking garage lights, or provide traffic meter staff with information about expired or inoperable meters.
Uh.. Can you summarize?

• For example ‘smart electricity meters’ in the home in California will use IPv6, and may well use ULA
  – There are 40,000,00 people in California
• ‘the internet of things’ is a strong possibility
• ULA is out there in the wild
• And, it leaks
  – DNS lookups inside the ULA cloud, out to the world?
ULA query growth, 2009-2011

Unique Local Address queries/Day

- 2009
- 2010
- 2011
Fine. Nobody uses site/link local..
Fine. Nobody uses site/link local..

[root@wally /usr/home/ggm]# tcpdump -i em0 'ip6 and not port 22'
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on em0, link-type EN10MB (Ethernet), capture size 96 bytes

06:06:02.926011 IP6 fe80::216:9dff:fe7a:8001 > ff02::d: PIMv2, Hello, length 82
06:06:07.221534 IP6 fe80::20e:cff:fe4b:f987 > fe80::216:9dff:fe7a:8001: ICMP6, neighbor solicitation, who has fe80::216:9dff:fe7a:8001, length 32
06:06:07.221715 IP6 fe80::216:9dff:fe7a:8001 > fe80::20e:cff:fe4b:f987: ICMP6, neighbor advertisement, tgt is fe80::216:9dff:fe7a:8001, length 24
Actually, *everybody* uses it

- NTP uses it
- V6 ‘arp’ uses it
- V6 ‘find my nearest router’ uses it
- ....And, we’re all logging it, and doing reverse-dns on it...
Scoped address query growth 2008-2011

link-local and site-local queries/day

- 2009
- 2010
- 2011
IPv6 is “chatty”

• All hosts, switches, lots of devices do a lot of service-rendesvous and hunting using link and site local addresses
  – Your HP printer came configured to do bonjour over IPv6 and is continually hunting, and being hunted by self-addressed Mac/OSX instances

• But, that’s low level and doesn’t trigger reverse DNS does it?
AAAAA vs A

• Modern apps routinely do AAAA/A joint lookup
  – And then do stupid things with the result
• If the app ‘thinks’ it finds IPv6 enabled it *will* attempt a connect
  – Which will often fall back on the link-local /64
  – Which may well fail, but will tickle something to log
    • Which generates an IPv6 reverse address lookup
Tunnels

• Least-good choice said to be Teredo
  – Highest apparent failure rate
  – Slow, endpoint selection semi-broken
  – New binding per site visited (!)
  – 2001::/32 prefix

• 6to4 (was) held to be better
  – But just as broken in its own way
  – 2002::/16 prefix
And the winner is...
And the winner is... (queries)
And the winner is... (queries)

LOG scale

- teredo
- 6to4
- global-unicast
Tunnels a problem?

• We added 2.0.0.2.ip6.arpa to DNS
  – Ugly but solved problem

• Its harder to add Teredo
  – More random tunnel binding (per session)
  – Inherently unscaleable

• In any case, these queries are mostly about FAILING tunnels:
  – The Teredo doesn’t reflect actual usage seen at applications-level logs, tests
Mapped addresses
Mapped addresses

• You think you might have V6
• You don’t know your V6 prefix
  – Simple! Whack your V4 into a V6 address
  – Set the upper 96 bits to 0
  – Go forth and prosper…
• AND somebody does IPv6 reverse-lookup on it
  – Its not meant to leak, but it does.
Mapped IPv4 addresses queries

Unspecified address queries/day

- 2009: 4,000,000
- 2010: 5,000,000
- 2011: 3,000,000
I want the curry. Can we stop now?
Its Log scale. 100x more silly Questions

- 2009:
  - silly questions: [Graph Data]
  - global unicast: [Graph Data]

- 2010:
  - silly questions: [Graph Data]
  - global unicast: [Graph Data]

- 2011:
  - silly questions: [Graph Data]
  - global unicast: [Graph Data]
Its Log scale. 100x more silly Questions

<table>
<thead>
<tr>
<th>Year</th>
<th>Silly Questions</th>
<th>Global Unicast</th>
</tr>
</thead>
<tbody>
<tr>
<td>2009</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2011</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

100x
I want the pizza. Can we stop now?

• There is at least 1, if not 2 decimal orders of magnitude more ‘silly’ DNS queries than useful ones in IPv6.

• This problem will not go away without work
  – Code fixes to reduce unneeded DNS requests
  – Local delegations in bind-9, but do people use them?
  – AS112 set-aside is looking compelling..
AS112? Pretend I don’t know..

• Anycast DNS delegate for the bogus queries that flood the root
  – Traffic localizes to nearest anycast NS instance
• Simple to run, open, localizes traffic
• Documented at
  – draft-dnsop-as112-under-attack-help-help
  – draft-dnsop-as112-ops
What does a Draft look like?

• “Dear IAB. Please instruct IANA to delegate the following reverse zone in ip6.arpa to AS112”
  
  • e.f.ip6.arpa
  • f.f.ip6.arpa
  • 0.0.0.0.ip6.arpa
  • :
What does a Draft look like?

• “Dear IAB. Please instruct IANA to delegate the following reverse zone in ip6.arpa to AS112”
  • e.f.ip6.arpa
  • f.f.ip6.arpa
  • 0.0.0.0.ip6.arpa
  • :

• (Plus about 5 pages of boilerplate)
What does a Draft look like?

• “Dear IAB. Please instruct IANA to delegate the following reverse zone in ip6.arpa to AS112”
  • e.f.ip6.arpa
  • f.f.ip6.arpa
  • 0.0.0.0.ip6.arpa
  • :

• (Plus about 5 pages of boilerplate)
• draft-michaelson-as112-ipv6-00
What can I do?

• Get newer bind configs
  – Operate with local master for the ‘silly’ V6 spaces
• Check your logs!
• Think about what all this multicast and discovery IPv6 traffic is doing?
  – How far does it flow?
  – Does it leak off-link, off-site?
  – There is no private IP any more. It all leaks
Not another ‘V6 is doomed’ pack

• Remember this only scales to disaster if IPV6 succeeds
  – The Teredo problem goes if tunnels go
• Skepticism aside, this has potential to become a large problem, high in the DNS server tree
  – For the life of dual-stack, if not beyond
• We dodged this in IPv4 by taking action (AS112)
• This pack is arguing we just extend it to IPv6