How I learned to stop worrying and love IPv6:

Measuring the Internet's readiness for DNS over IPv6

Tobias Fiebig

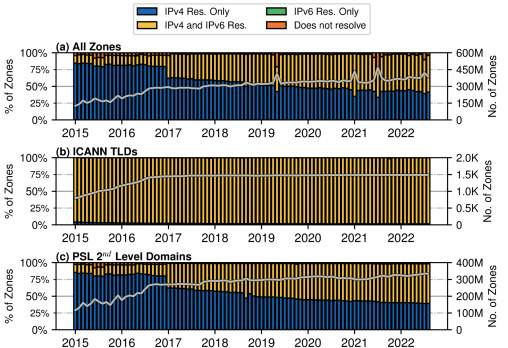
Max-Planck Institut für Informatik

DNS & IPv6: RFC3901

- DNS IPv6 Transport Operational Guidelines
- Published in 2004
- Bottom line:
  - All recursive DNS servers SHOULD be IPv4-only or dual-stack (IPv4 only allowed, IPv6 not required)
  - All zones SHOULD have at least one IPv4 name server (IPv6 implicitly optional)
  - Zone validation SHOULD ensure one IPv4 authoritative

RFC3901 Impact

- IANA TLD requirements only mandate IPv4[a]
- Most TLDs only mandate IPv4
- IPv6 adoption in the DNS is lacking behind (by now ca. 60% of zones)[b] [c]



[a] `https://www.iana.org/help/nameserver-requirements`

[b] Steurer, Florian, Anja Feldmann, and Tobias Fiebig. “A Tree in a Tree: Measuring Biases of Partial DNS Tree Exploration.” PAM, 2025.

[c] Streibelt, Florian, et al. “How ready is DNS for an IPv6-only world?” PAM, 2023.

Claim: DNS is not ready for IPv6

10+ year discussion:[1]

Mostly related to IPv6 fragmentation

- IPv6 does not allow on-path fragmentation
- Path MTU Discovery (PMTUD) is notoriously ‘difficult’ for IPv6
- Large fragment drop rates for IPv6 fragments on the Internet break DNS-over-UDP (especially) for DNSSEC (for `DNSKEY` records)
- TCP fallback does not always work
- DNSSEC needs even more TCP fallback (for `DNSKEY` records)

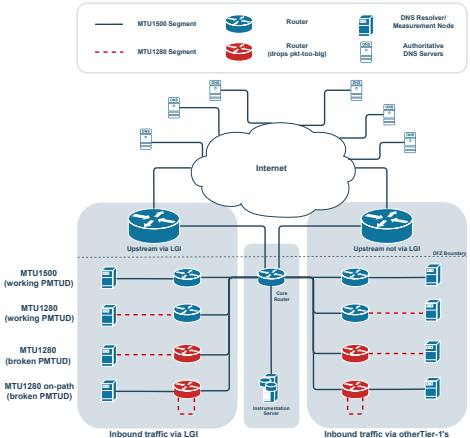
[1] See also (from 2015): https://insinuator.net/2015/11/some-notes-on-the-drop-ipv6-fragments-vs-this-will-break-dnssec-debate/

‘Q: Is DNS resolution impaired by IPv6?’

- **RQ1:** Does DNS resolvability suffer when using IPv6?
  E.g., more TIMEOUTs, SERVFAILs, or TCP fallbacks?
- **RQ2:** What is the impact of challenging MTU scenarios on DNS resolvability?
- **RQ3:** Does fragmentation differ for DNS over IPv4 vs. IPv6?
- **RQ4:** Do IPv6 fragment drops negatively impact DNS resolvability?
- **RQ5:** What is the impact of DNSSEC on DNS resolvability over IPv6 vs. IPv4?

Methodology: Capture DNS Resolution

- Resolve the Google CrUX Top 10M
  Use ‘minimally covering NS sets’[ab]
- IPv4 only/IPv6 only/dual-stack
- Different MTU scenarios:
  - MTU1500
  - MTU1280 on-link (working PMTUD)
  - MTU1280 on-link (broken PMTUD)
  - MTU1280 on-path (broken PMTUD)
- Once via specific Tier-1 (LGI),
  Once via other upstreams



[a] The set of NS sets in which each NS set is represented with one zone (1x DNSSEC, 1x without DNSSEC)

[b] All data is available on https://dns-mtu.measurement.network/ (past 7 days) and https://data.measurement.network/

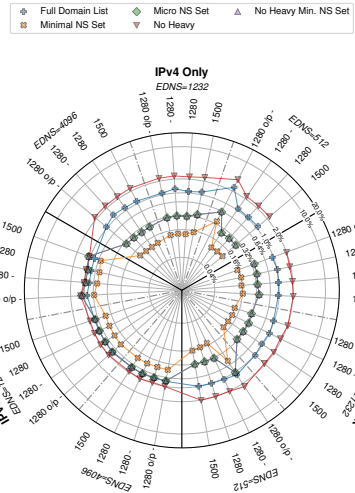
Minimally Covering NS Sets

- Idea from Petr Špaček at IETF121: Hit each NS Set only once
  - Reduces load
  - Should still hit all oddities
- Problems:
  - Some really dominant NS Sets (GoDaddy, Cloudflare, domain parking)
  - Operators with many similar NS Sets (Cloudflare, Amazon)
- **Question:** Is this really representative?
- Project with Taha Albakou

Methodology

- Measure the Google CrUX Top 10M without minimally covering NS Sets
- Scale dataset based on what the minimally covering NS Set would have looked like
- Re-Scale from minimally covering NS Set weighted based on the number of domains
- Slices:
  - Minimally Covering NS Set (one zone per NS Set)
  - Micro Covering NS Set (aggregate heavy hitter NS Sets, CF/Amazon etc.)
  - No Heavy Hitters (Remove Top NS Sets)
  - Minimally Covering NS Set over No Heavy Hitters set

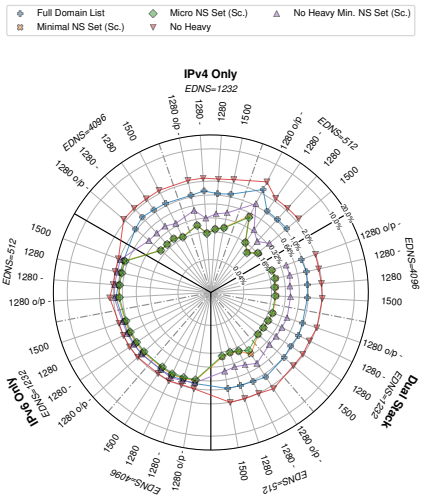
Results: Without DNSSEC



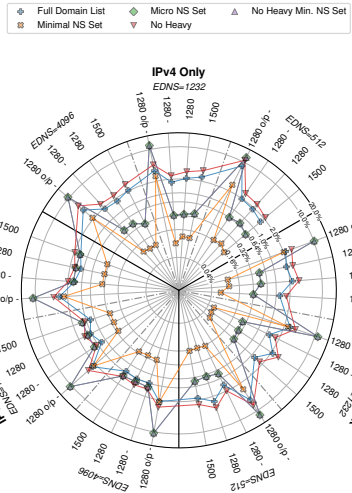
- Relatively, *over*estimates IPv6 `TIMEOUT`s and *under*estimates IPv4 `TIMEOUT`s
- Removing top-NS Sets makes everything worse (top NS Sets introduce positive bias)
- IPv6 looks, overall, better
- Dual-Stack seems to be mostly influenced by IPv4

Results: Without DNSSEC (Scaled)

- For IPv6, scaled minimal/micro NS Sets are very close to measuring the full set
- For IPv4, the distance closes



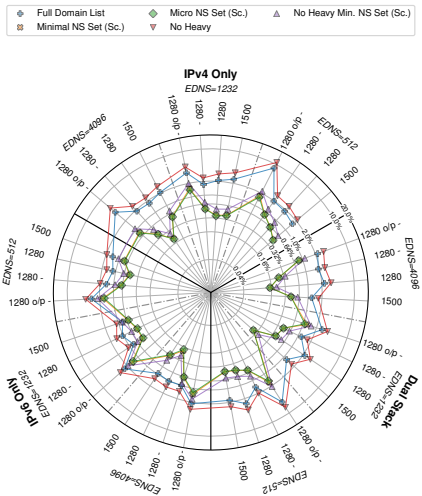
Results: With DNSSEC



- High `TIMEOUT`s for MTU1280 on-path with broken PMTUD regardless of IPv4/IPv6/Dual-Stack (over-amplified in minimal NS-Set)
- IPv6 looks worse in minimal NS Sets than over the full set
- Again, the overall pattern remains similar

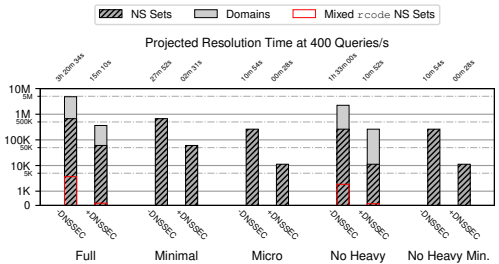
Results: With DNSSEC (Scaled)

- Overall less difference between scaled/full sets than without DNSSEC
- IPv6 again closer to scaled values
- Under-Estimation of `TIMEOUT`s for 1280MTU on-link in minimal sets



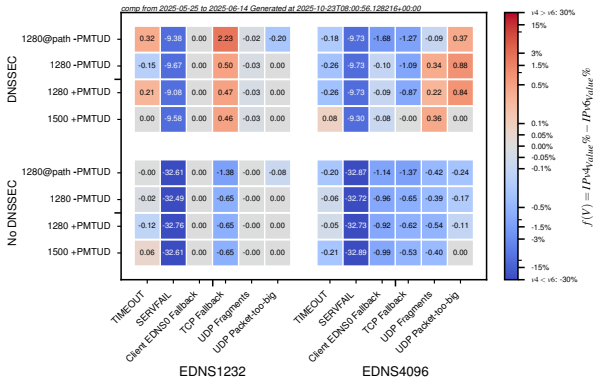
Minimal NS Sets: Bottom Line

- Minimally covering NS Sets / Micro covering NS Sets save a lot of time when measuring
- A fraction of NS Sets (around 0.1%) has ‘mixed’ `rcodes`
- Scaling minimal NS Sets gets closer to the absolute values to expect
- *Relative* values are mostly fine (IPv6 has it a bit harder, though)



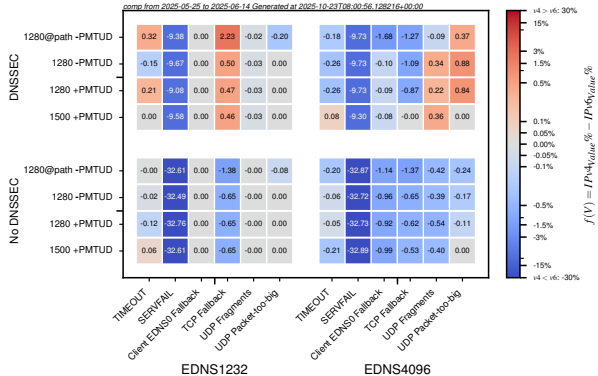
Results: IPv4 vs. IPv6

- Mostly comparable: $< \approx 3\%$
- `SERVFAIL` for No DNSSEC $\approx$ 33%, but $\leq$ 10% for DNSSEC
- Some IPv6 packets get UDP packet too big @EDNS0 1232



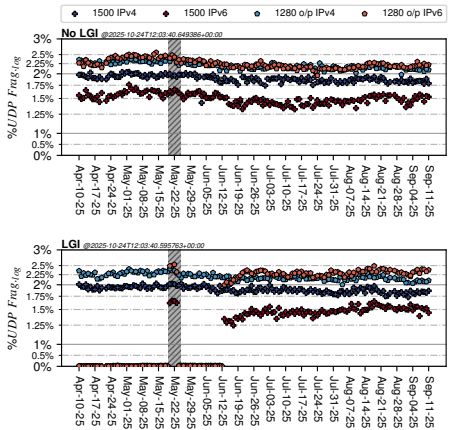
Results: IPv4 vs. IPv6 cont.

- Small baseline of broken IPv6 NS sets ($\leq$0.15%)
- For DNSSEC: *More* fragments for IPv4 @EDNS0 4096
- For broken PMTUD, IPv4 has more `TIMEOUT`s with DNSSEC @EDNS0 1232



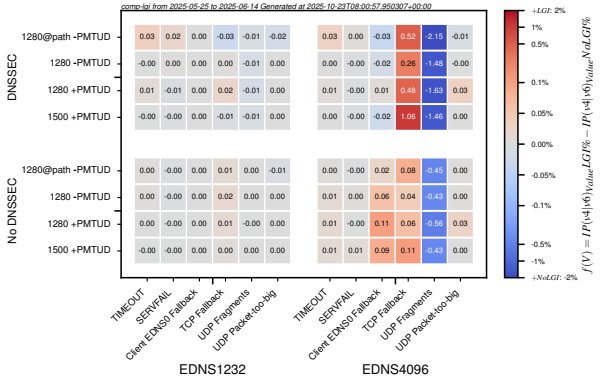
Results: IPv6 Fragment Drop Rates

- One Tier-1–AS6830–(used to) drop IPv6 fragments for transit
- Tier-1 fixed this *very* quickly with *a lot* of effort
- Result: More IPv6 than IPv4 fragments for broken MTU cases



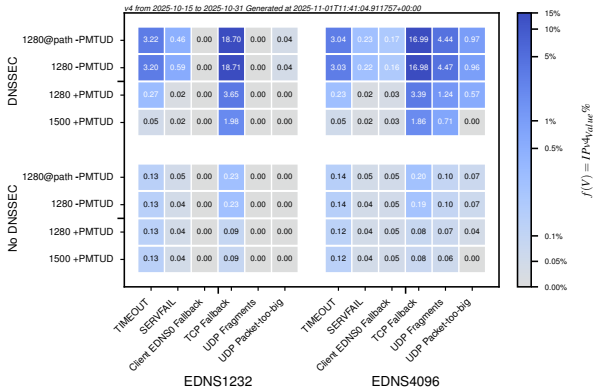
Different Perspective–No IPv6 Fragments

- EDNS0=1232
  - Following BCP: No notable difference
- EDNS0=4096
  - Worst-case: $\approx$ 1% more TCP fallbacks
  - Without DNSSEC: Slightly more EDNS0 fallback ($\leq$0.12%)
  - Difference in `TIMEOUT`s is almost non-existent



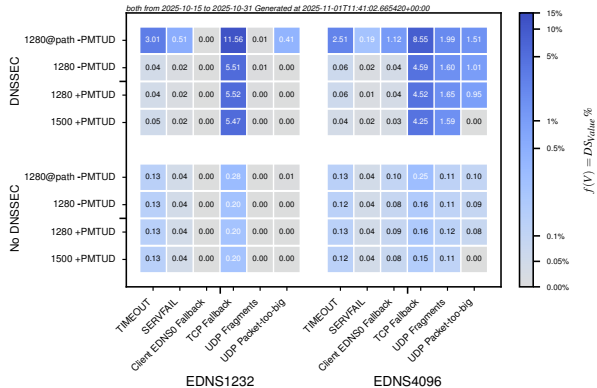
NAT64 (with a Minimally Covering NS Set... Set)

- Timeout baseline for ‘No DNSSEC’ (likely from forced DualStack)
- Less `TIMEOUT`s w. DNSSEC @MTU1500
- Broken PMTUD is TCP Fallback heavy
- Difference between an on-path MTU break and an on-link break is limited



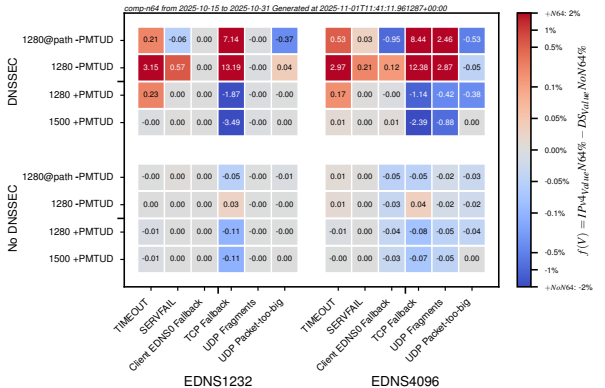
Dual-Stack

- DS also suffers for on-path breaks without PMTUD
- Similar `TIMEOUT` baseline without DNSSEC
- Overall consistent with prior findings for v4/v6



NAT64 vs. Dual-Stack

- Comparable @MTU1500, and generally without DNSSEC
- OK-ish similar for on-path MTU break without PMTUD & on-link with PMTUD
- Worse for on-link break without PMTUD (ca. similar to on-path in class)



‘Q: Is DNS resolution impaired by IPv6?’

- **RQ1:** Does DNS resolvability suffer when using IPv6?
  E.g., more TIMEOUTs, SERVFAILs, or TCP fallbacks?
  - No—Base-line differences most likely due to broken IPv6 authoritative NS
- **RQ2:** What is the impact of challenging MTU scenarios on DNS resolvability?
  - IPv4/v6 differences are still limited; RFC9715 (EDNS0=1232) helps most
- **RQ3:** Does fragmentation differ for DNS over IPv4 vs. IPv6?
  - Yes, up until LGI stopped dropping fragments; Access networks might experience this differently.

‘Q: Is DNS resolution impaired by IPv6?’

- **RQ4:** Do IPv6 fragment drops negatively impact DNS resolvability?
  - No, not even if DNSSEC is in use.
- **RQ5:** What is the impact of DNSSEC on DNS resolvability over IPv6/v4?
  - There seem to be *more* NS-Sets supporting IPv6 *and* DNSSEC; Likely due to technology-adoption-affinity.

Conclusion

- The Internet is ready for more IPv6 in the DNS
- Fragmentation in IPv6 – no major role for DNS
- Misconfigured IPv6 DNS has *an order of magnitude* more impact on `TIMEOUT`s than fragment dropping
  - Better guidance on use of IPv6 (SHOULD) and authoritative testing (SHOULD) in an RFC3901bis will help
- IPv4 does not necessarily ‘perform better’ for DNS resolution
- People following RFC9715, doing IPv6 for DNS, and running NS sets on which at least one domain has DNSSEC seem to coincide

Conclusion

- Minimally covering NS Sets work (but better scale them back up if you are looking for absolute values)
- Better skip on NAT64 for DNS and try to go native

Data: `https://data.measurement.network`

(Ask me for NFS access if you want to copy $everything)