DNS Monitoring with Anteater

Giovane C. M. Moura[1] John Heidemann[2] Wes Hardaker[2]
Jeroen Bulten[3] João Ceron[1] Cristian Hesselman[1,4]

1: SIDN Labs, **2**: USC/ISI, **3**: SIDN, **4**:University of Twente

ICANN 70 – Tech Day
Virtual Conference
2021-03-22

This talk

- Based on a technical report
  - **Old but Gold: Prospecting TCP to Engineer DNS Anycast (extended)**
  - https://www.isi.edu/~johnh/PAPERS/Moura20a.pdf
- We show how rich is DNS over TCP for anycast engineering
- We presented this report at DNS-OARC34, for full video check:
  - https://youtu.be/K_3zTY3gAgo?list=PLCAxS3rufJ1eZ3q9IcQ2QFT4fwasAqttL&t=3754
- Today: more focus on the tool (`Anteater`)
- For DNS/TPC RTT background: check OARC34 presentation and technical report.

Latency is key in DNS (but hard to measure)

- Authoritative OPs will use whatever tools to reduce latency:
  1. multiple NSes
  2. Anycast
  3. Peering/IXPs
  4. ...
- But is **hard to know** client's latency:
  1. Ripe Atlas, Thousand Eyes: good but not complete coverage
  2. Verfploeter [1]: requires ICMP measurements
     - Verfploeter is ran typically daily, as it is expensive
     - Difficult to apply to IPv6 (hitlist)

What if there was a better way ?

- A method that:
    - Comes from *real-clients*
    - Works well with IPv6
    - Requires *no extra* measurements (passive only)
- Well, there is one: **DNS over TCP (DNSTCP)**
    - RTT measured from handshake (or takedown)
    - we've been using for 1.5 years at SIDN (`.nl`)
    - helped to solve several issues
    - fulfills all the above

What if there was a better way ?

- A method that:
  - Comes from *real-clients*
  - Works well with IPv6
  - Requires *no extra* measurements (passive only)
- Well, there is one: **DNS over TCP (DNSTCP)**
  - RTT measured from handshake (or takedown)
  - we've been using for 1.5 years at SIDN (`.nl`)
  - helped to solve several issues
  - fulfills all the above

TCP RTT history: old but gold

- TCP RTT estimation has been used since 1996 [2]
- Widely used in passive analysis of HTTP (FB uses it [5])
- It has been applied on DNS mulitple times:
  - Roy Arends (2012)
  - Casey Deccio (2018)
  - Maciej Andzinski [3] (2019)
  - Our tech report (2020) [4]

Our contribution

So what's NEW with our work?

- extensive and comprehensive methodology validation
  - Is the TCP data representative?
  - Are the UDP and TCP latency comparable?
- acted upon the data with 4 operators (Anycast A, B, B-Root, and Google)
  - We identify several use cases and issues
  - We manipulated BGP to fix those issues
  - We document it carefully
- use in real-time within `.nl` to detect anomalies
  - Route leaks
- Release our monitoring tool (Anteater) open source:
  - https://github.com/SIDN/anteater
- NEW: `dnsanon` also supports DNS TCP RTT: https://ant.isi.edu/software/dnsanon

Requirements for DNS/TCP RTT

TCP traffic **MUST**:

1. Provide enough **coverage** (spatial and temporal)
    - you know, most DNS traffic is still UDP
2. provide **similar latency** to UDP
    - so we can generalize the results

Is DNS traffic representative?

	Queries		Resolvers		ASes	
	Anycast A	**Anycast B**	**Anycast A**	**Anycast B**	**Anycast A**	**Anycast B**
Total	5 237 454 456	5 679 361 857	2 015 915	2 005 855	42 253	42 181
IPv4	4 005 046 701	4 245 504 907	1 815 519	1 806 863	41 957	41 891
UDP	3 813 642 861	4 128 517 823	1 812 741	1 804 405	41 947	41 882
TCP	191 403 840	116 987 084	392 434	364 050	18 784	18 252
ratio TCP	5.02%	2.83%	21.65%	20.18%	44.78%	43.58%
IPv6	1 232 407 755	1 433 856 950	200 396	198 992	7 664	7 479
UDP	1 160 414 491	1 397 068 097	200 069	198 701	7 662	7 478
TCP	71 993 264	36 788 853	47 627	4 6190	3 391	3 354
ratio TCP	6.2%	2.63%	23.81%	23.25%	44.26%	44.85%

Table 1: DNS usage for two authoritative services of `.nl` (Oct. 15–22, 2019).

- 5% of clients, 20% of resolvers, and 44% of ASes
- You get this for free

DNS: TCP vs UDP latency are comparable

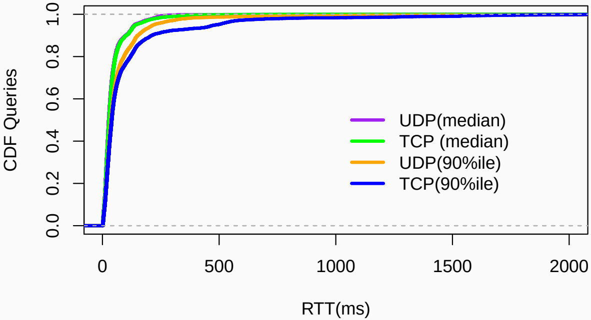


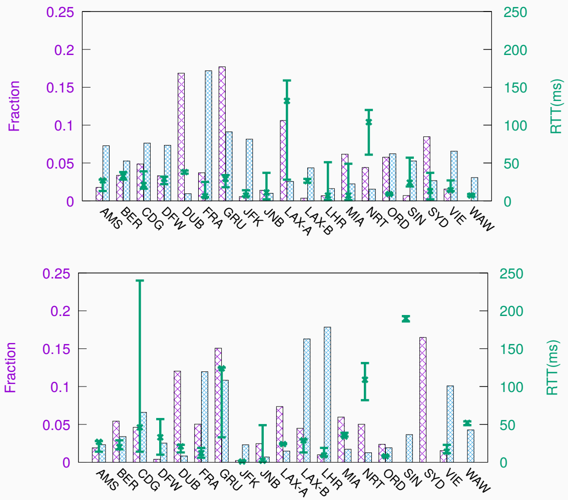
Figure 1: L-Root: CDF of median and 90%ile RTT for DNS/UDP and DNS/TCP.

OK, so what can we do with it?

- DNS/TCP provides enough VPs
- Has similar latency than UDP
- Measure real clients
- No costs
- Easily copes with IPv6
- Requires no extra measurements
- Can be run in real time

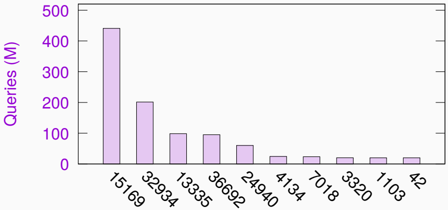
Prioritizing Analysis: by Site

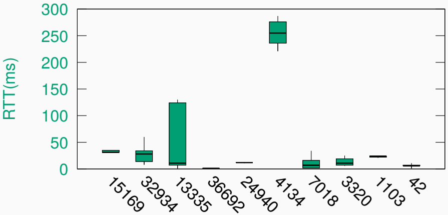
Anycast B: IPv4 and IPv6 RTT per site



Prioritizing Analysis: by client AS

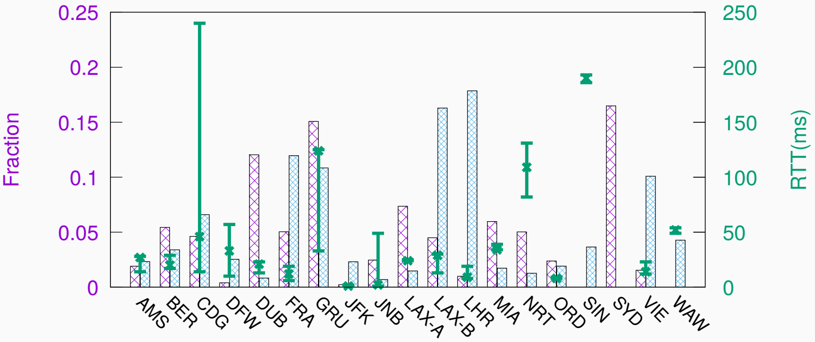
Anycast B: IPv6 queries and RTT per client AS





Problems: Distant Lands

- A client is mapped by BPG to far distant anycast sites
- Some sites have a large RTT value or spread (CDG, SIN, NRT)
- We can see that using DNS/TCP RTT



Solutions: Distant Lands (NRT)

- Causes: No presence/direct peer with Chinese ISPs
- Chinese int'l connections can exhibit congestion [6]
- Fix: site in China (OPs clients may not be confortable) or direct peer ($)

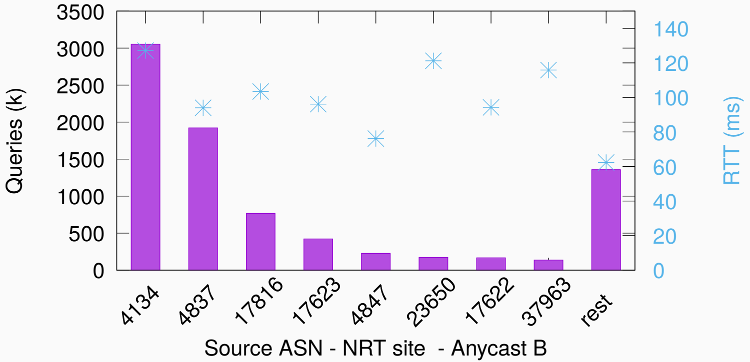
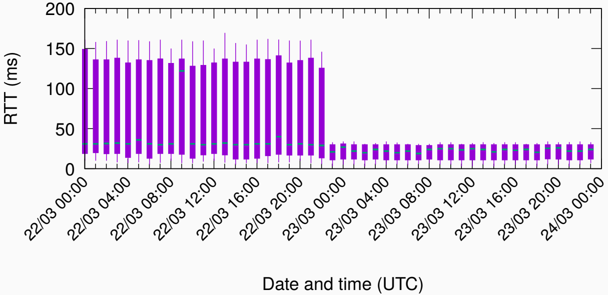


Figure 2: Anycast B, Japan site (NRT): Top 8 querying ASes are Chinese, and responsible for

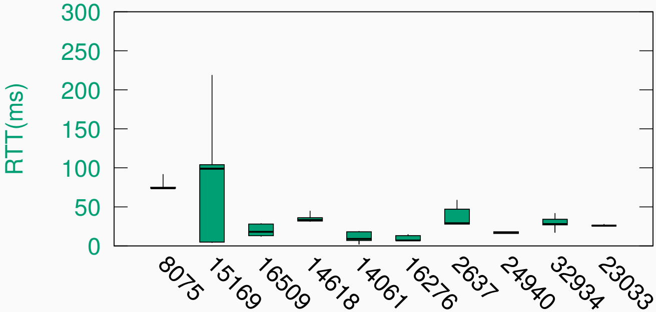
Problems: prefer customer to another continent

- Common BGP policy: `prefer customer`
  - if AS can satisfy route via customer, so be it
- But sometimes it takes clients to another continent
- We found Comcast (US, AS7922) reaching Anycast B via GRU site (Brazil)
- We contacted the Operator; fixed with right BGP community



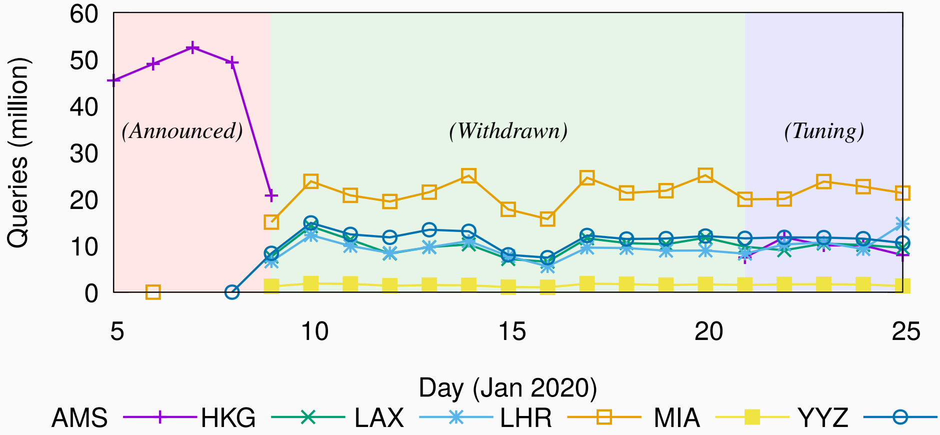
Problem: Anycast Polarization

- We found that MS (8075) and Google (15169) had high latencies to Anycast A
- And they are the top 2 client ASes



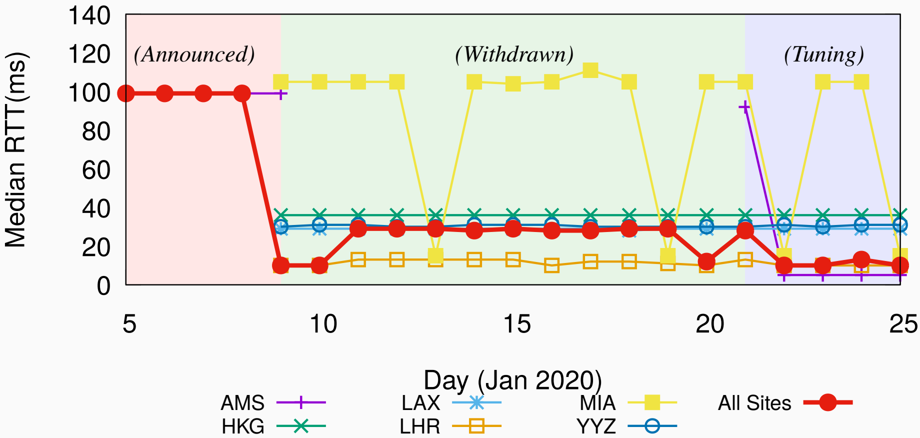
Problem: Google Polarizided → high latency

- All Google Traffic was going to AMS site only : RTT 100ms



Solution: Depolarizing traffic from Google (BGP)

- We fixed the issue with BGP manipulations
- Median latency: from 100ms to 10ms.



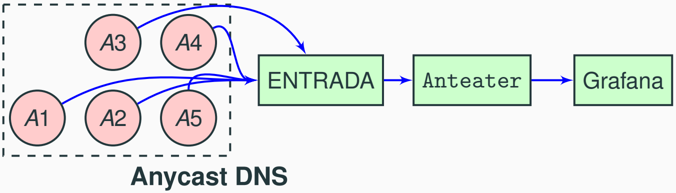


Figure 6: DNS/TCP RTT near real-time monitoring at `.nl`

https://github.com/SIDN/anteater

DEMO

Summary

- DNS/RTT are useful for Anycast Engineering
- We document Anycast Polarization, and shed latency in 90ms
- We've been using it for over 1.5 year at SIDN (`.nl`)
- We've released `Anteater` open source! Download it
  - https://github.com/SIDN/anteater
- Tech report: https://www.isi.edu/~johnh/PAPERS/Moura20a.html

References i

[1] de Vries, W. B., de O. Schmidt, R., Haraker, W., Heidemann, J., de Boer, P.-T., and Pras, A.

Verfploeter: Broad and load-aware anycast mapping.

In *Proceedings of the ACM Internet Measurement Conference* (London, UK, 2017).

[2] Hoe, J. C.

Improving the start-up behavior of a congestion control scheme for tcp.

In *Proceedings of the ACM SIGCOMM Conference* (Stanford, CA, Aug. 1996), ACM, pp. 270–280.

[3] Maciej Andzinski .

Passive analysis of DNS server reachability.

https://www.nic.cz/files/nic/IT_19/prezentace/12_andzinski.pdf, 11 2019.

[4] Moura, G. C. M., Heidemann, J., Hardaker, W., Bulten, J., Ceron, J., and Hesselman, C.

Old but gold: Prospecting TCP to engineer DNS anycast (extended).

Tech. Rep. ISI-TR-740, USC/Information Sciences Institute, June 2020.

[5] Schlinker, B., Cunha, I., Chiu, Y.-C., Sundaresan, S., and Katz-Bassett, E.

Internet Performance from Facebook's Edge.

In *Proceedings of the Internet Measurement Conference* (New York, NY, USA, 2019), IMC '19, ACM, pp. 179–194.

[6] Zhu, P., Man, K., Wang, Z., Qian, Z., Ensafi, R., Halderman, J. A., and Duan, H.

Characterizing transnational internet performance and the great bottleneck of china.

Proc. ACM Meas. Anal. Comput. Syst. 4, 1 (May 2020).