UNIVERSITY OF AMSTERDAM

Defragmenting DNS

Determining the optimal maximum UDP response size for DNS

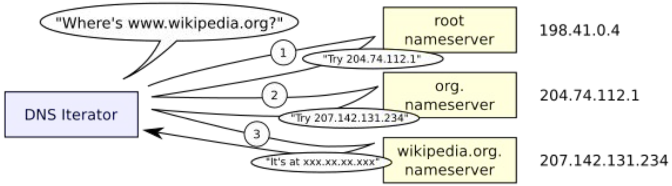
Research Project 2

Security and Network Engineering
University of Amsterdam

Axel Koolhaas & Tjeerd Slokker, July 2020

Background

- The Domain Name System (DNS) translates host names into IP addresses
- DNS works with Resource Records
  A, AAAA, DNAME, etc...



Source: Wikipedia.org

Background

- EDNS(0) are extension mechanisms for DNS, and the current default
    - EDNS has UDP Message Size, communicating response size capability

- The Internet is a network of networks
    - Not every network has the same Maximum Transmission Unit (MTU)

Background

- Path MTU Discovery (PMTUD) discovers Path MTU between two nodes
  - PMTUD is flawed, due to conservativity and failing ICMP messages

- Fragmentation occurs when a packet exceeds the PMTU
  - IP fragmentation introduces fragility to DNS
  - ICMP messages cause problems for DNS servers since they are stateless

Recap

- PMTUD is unreliable
- DNS is connectionless which causes problems with fragmentation of DNS packets

❖ We aim to suggest an optimal maximum EDNS message size for DNS

Research Questions

- What is the optimal EDNS message size to avoid IP fragmentation?
  - Is there a difference between IPv4 and IPv6 regarding PMTU sizes?
  - Which EDNS message size is best in terms of support for DNS stub resolvers?
  - Which EDNS message size is best in terms of support for DNS open resolvers?

Related Work

How many problems does fragmentation cause?

- Weaver, et al. showed that 9% of DNS resolvers don’t receive fragmented UDP datagrams [1]
- Van Den Broek, et al. expanded on this, showing that as much as 10.5% of all resolvers suffer from fragmentation-related connectivity issues [2]

Related Work

How can you measure the PMTU?

- Toorop used custom name servers experiment with different EDNS message sizes [3]
  - Different sub-domains produce different sized responses
- DNS-OARC used a custom DNS server and chained CNAME responses [4]
  - Server sends multiple replies, where each reply decreases in size.

❖ Both use custom name servers, decreasing reproducibility

Related Work

How can fragmentation in DNS be prevented?

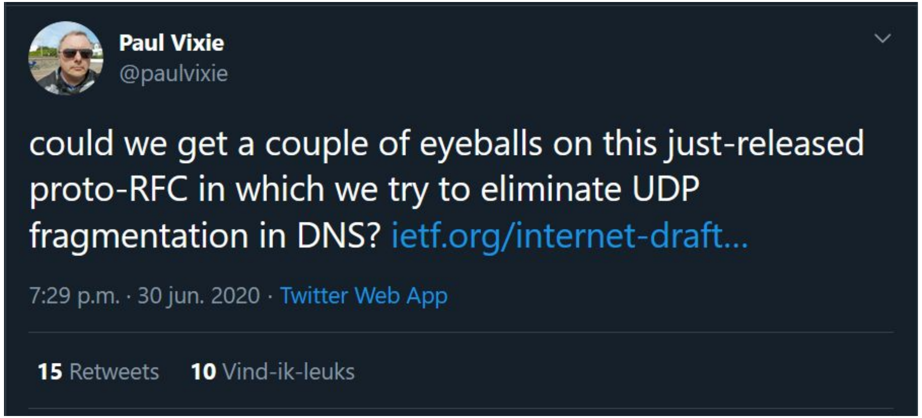
- Fujiwara & Vixie wrote a RFC draft on fragmentation avoidance in DNS [5]
  - A suggestion is made on a possible maximum EDNS message size

Related Work

How can fragmentation in DNS be prevented?

- Fujiwara & Vixie wrote a RFC draft on fragmentation avoidance in DNS [5]
  - A suggestion is made on a possible maximum DNS/UDP payload size

❖ Topical subject!



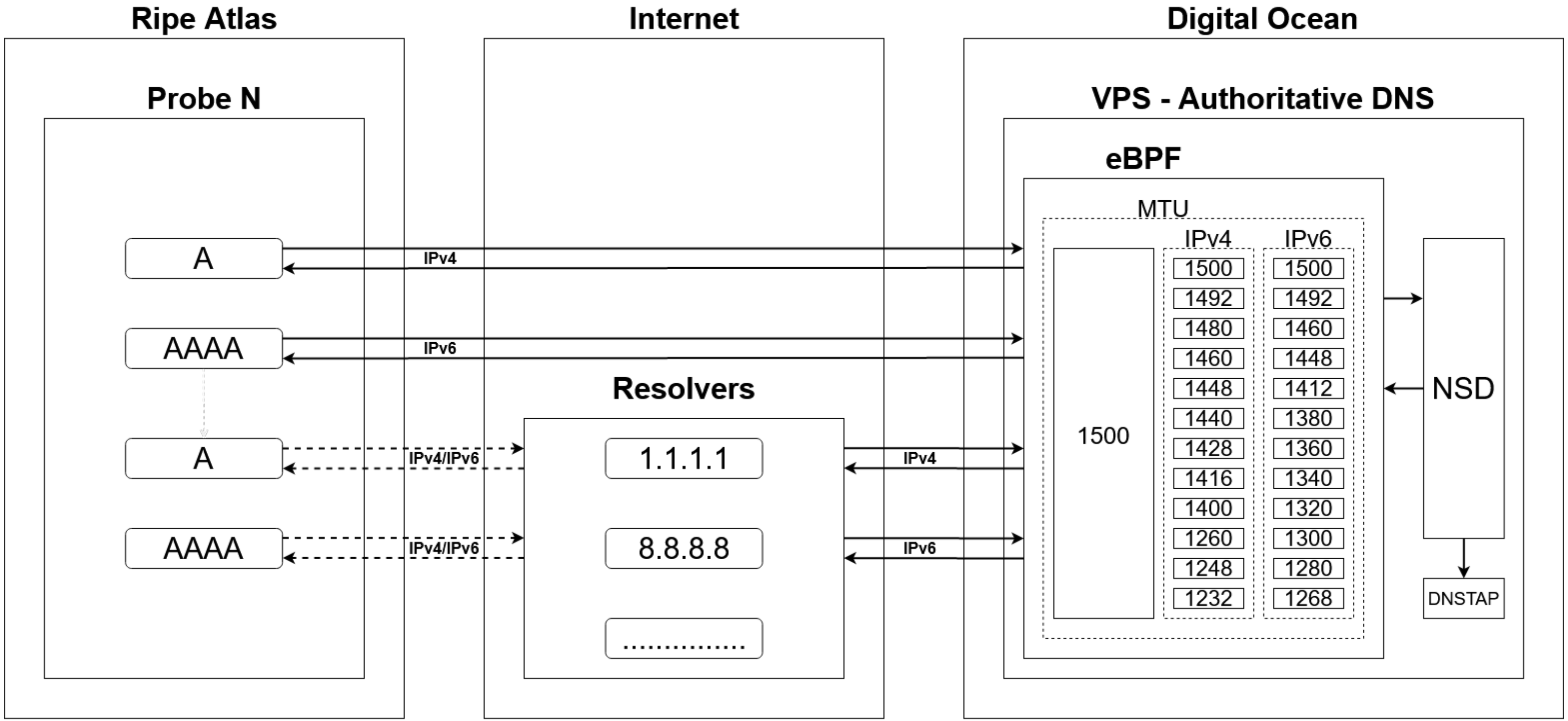
Paul Vixie
@paulvixie

could we get a couple of eyeballs on this just-released proto-RFC in which we try to eliminate UDP fragmentation in DNS? ietf.org/internet-draft...

7:29 p.m. · 30 jun. 2020 · Twitter Web App

15 Retweets 10 Vind-ik-leuks

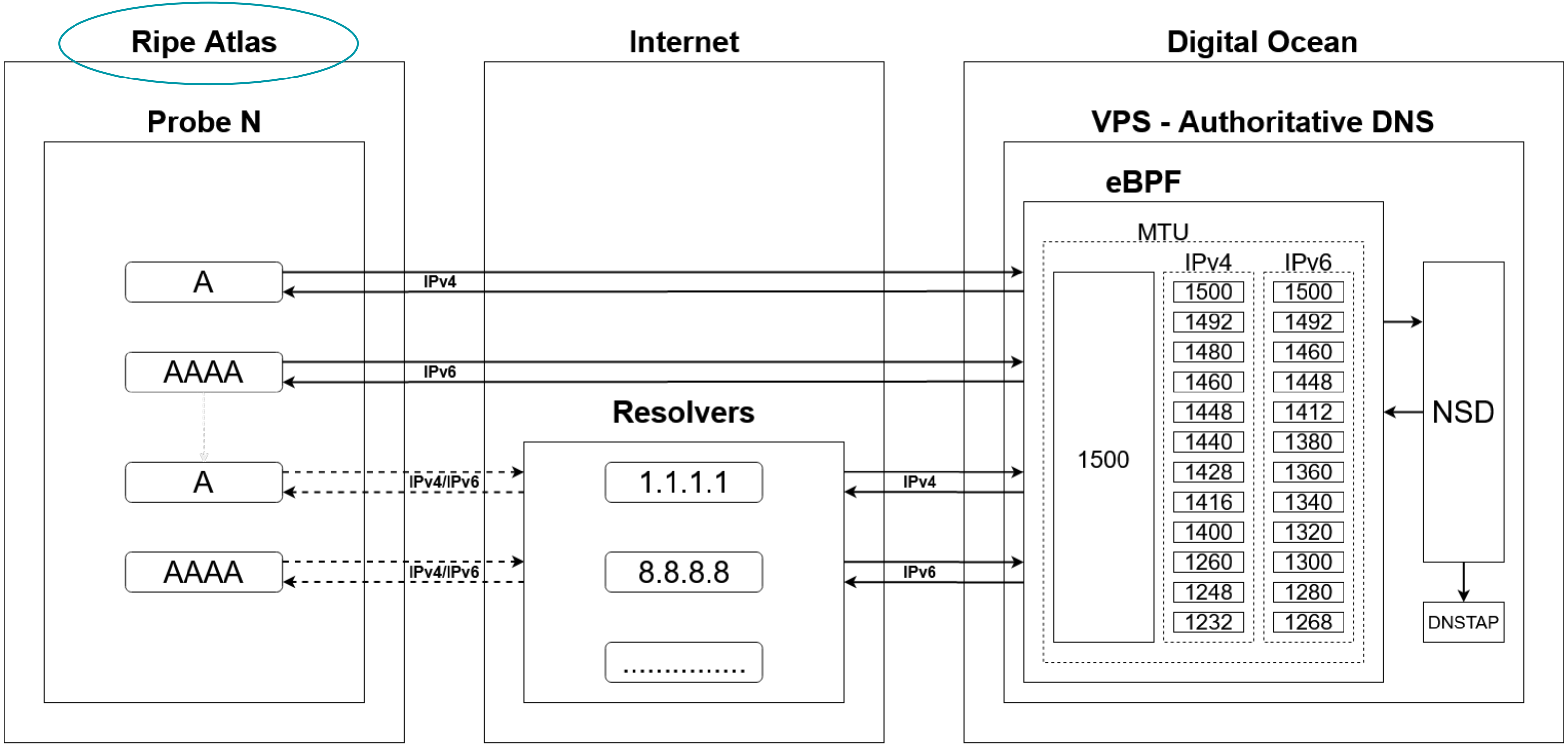
Methodology



IPv4	IPv6
1500	1500
1492	1492
1480	1460
1460	1448
1448	1412
1440	1380
1428	1360
1416	1340
1400	1320
1260	1300
1248	1280
1232	1268

Platform to perform measurements with

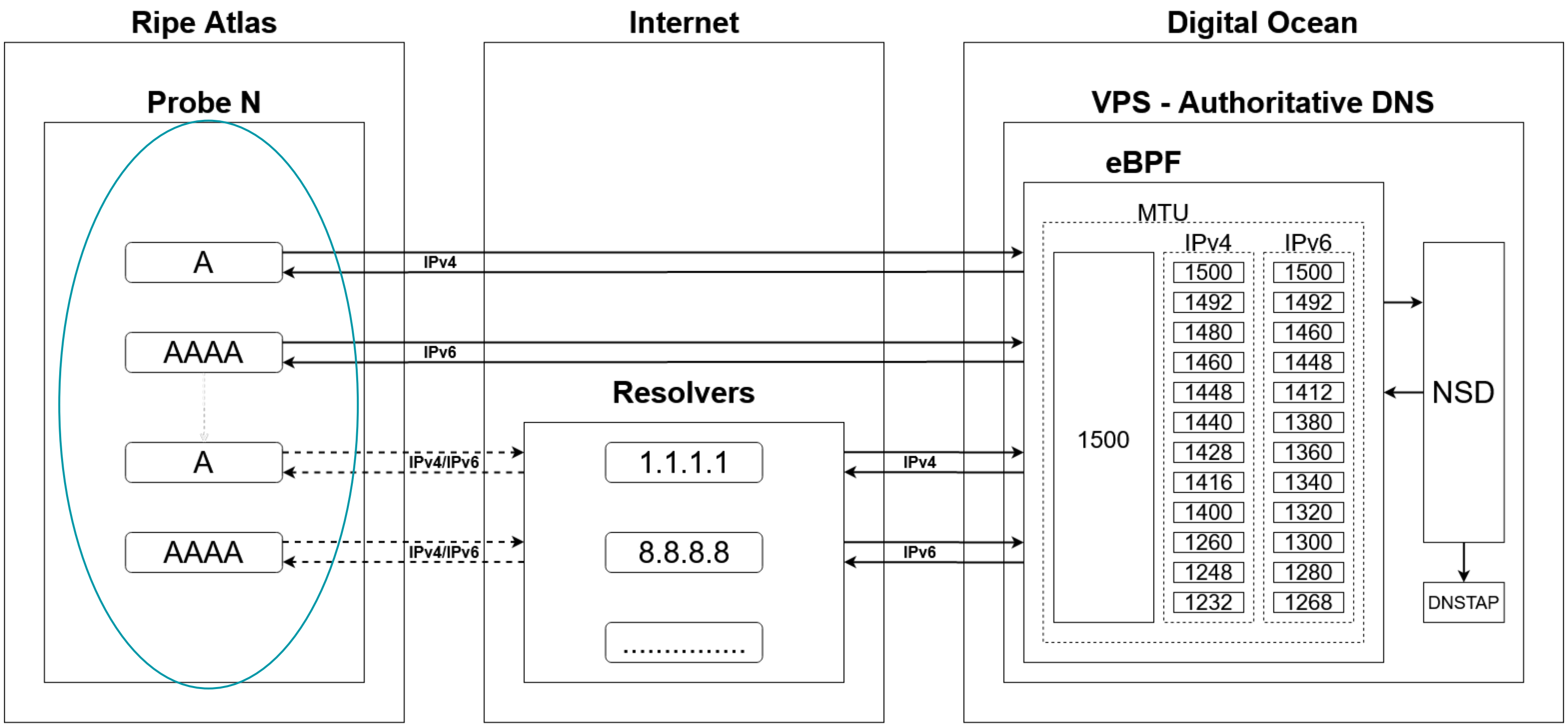
Methodology



IPv4	IPv6
1500	1500
1492	1492
1480	1460
1460	1448
1448	1412
1440	1380
1428	1360
1416	1340
1400	1320
1260	1300
1248	1280
1232	1268

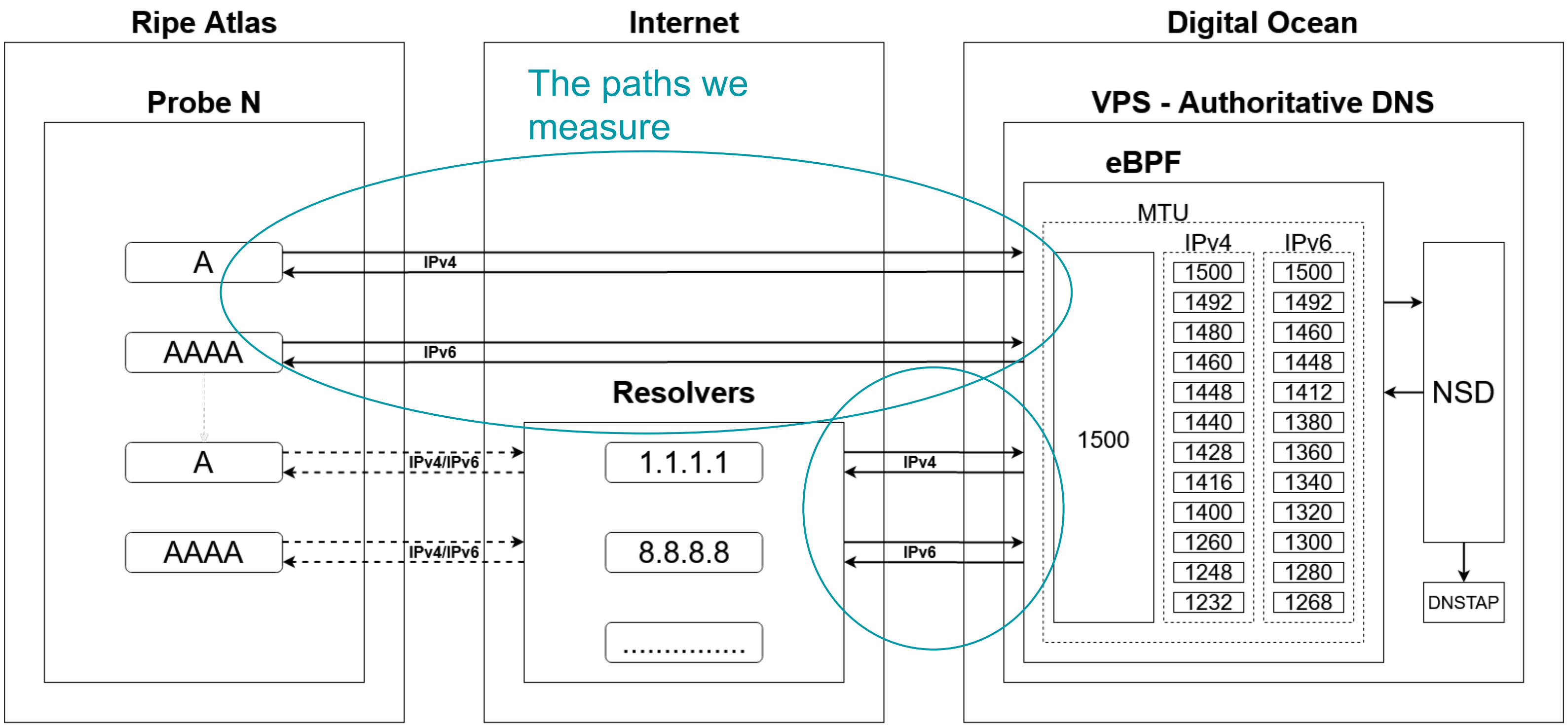
Four separate Atlas measurements

Methodology



IPv4	IPv6
1500	1500
1492	1492
1480	1460
1460	1448
1448	1412
1440	1380
1428	1360
1416	1340
1400	1320
1260	1300
1248	1280
1232	1268

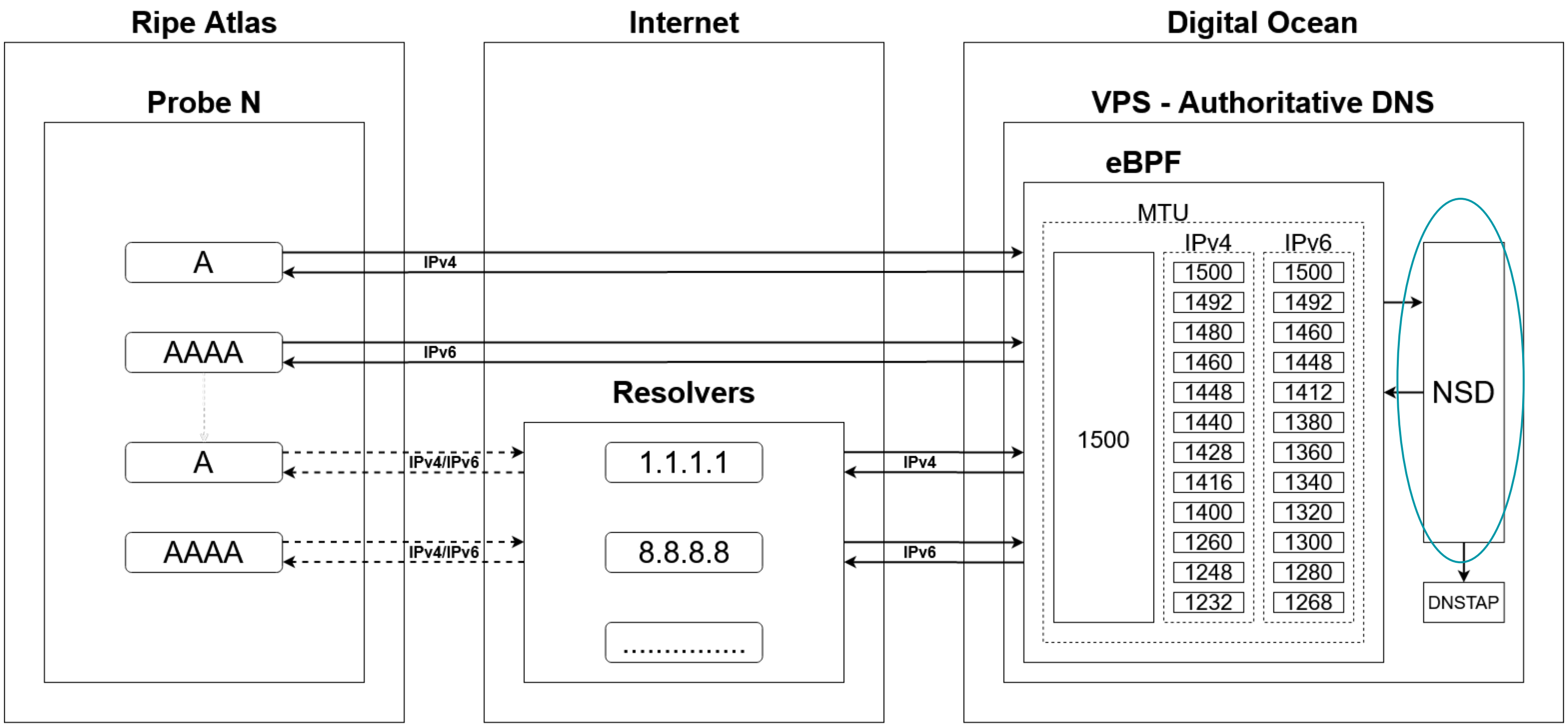
Methodology



IPv4	IPv6
1500	1500
1492	1492
1480	1460
1460	1448
1448	1412
1440	1380
1428	1360
1416	1340
1400	1320
1260	1300
1248	1280
1232	1268

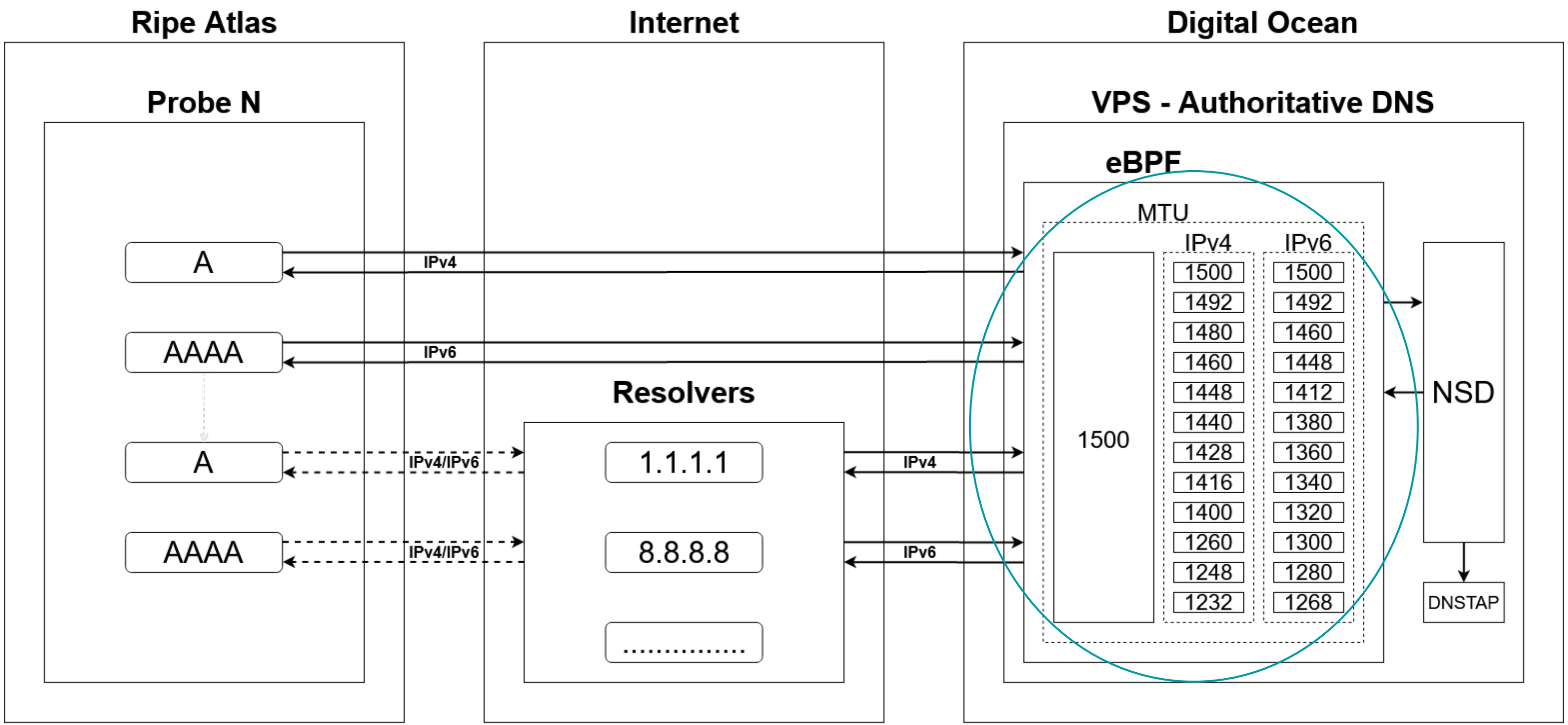
Methodology

Our DNS server



Methodology

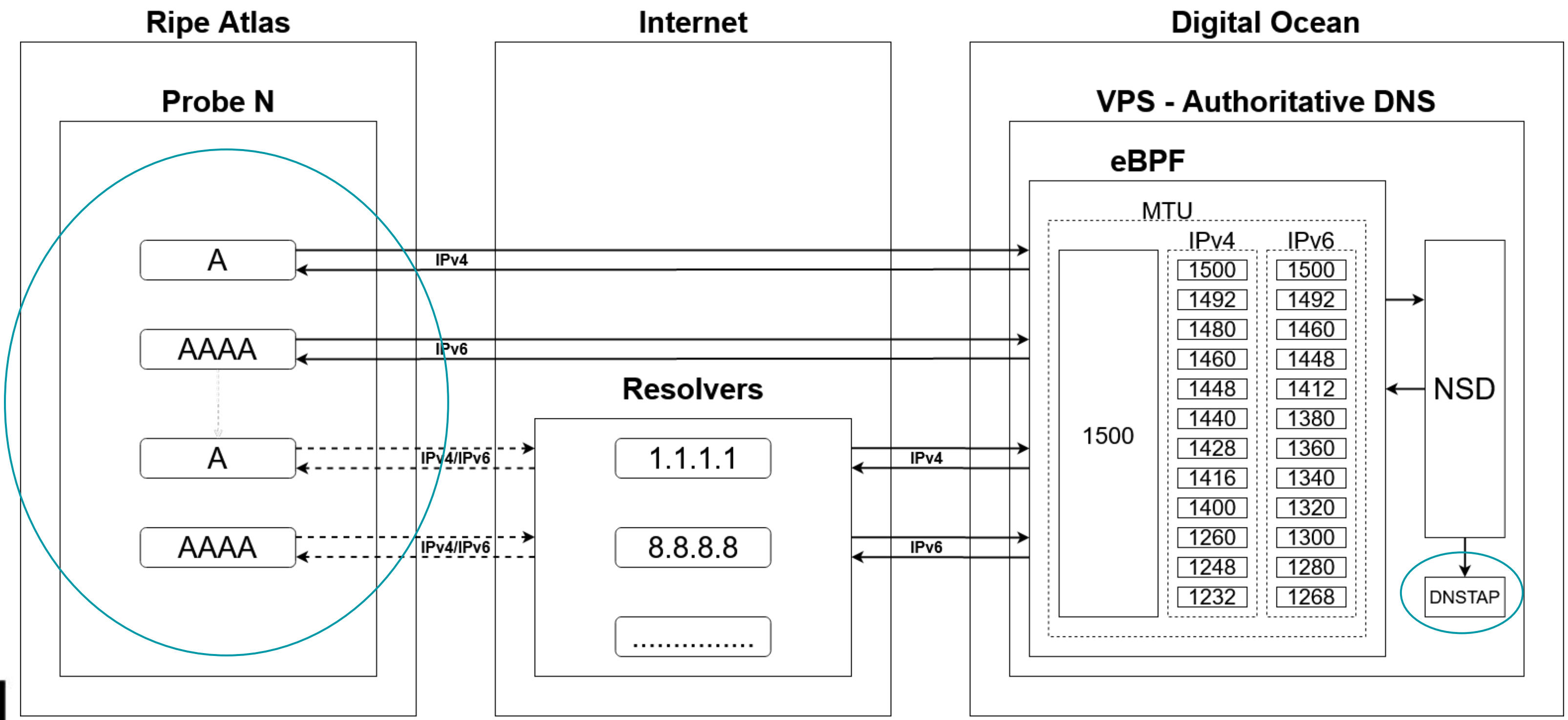
Solution to universal query



IPv4	IPv6
1500	1500
1492	1492
1480	1460
1460	1448
1448	1412
1440	1380
1428	1360
1416	1340
1400	1320
1260	1300
1248	1280
1232	1268

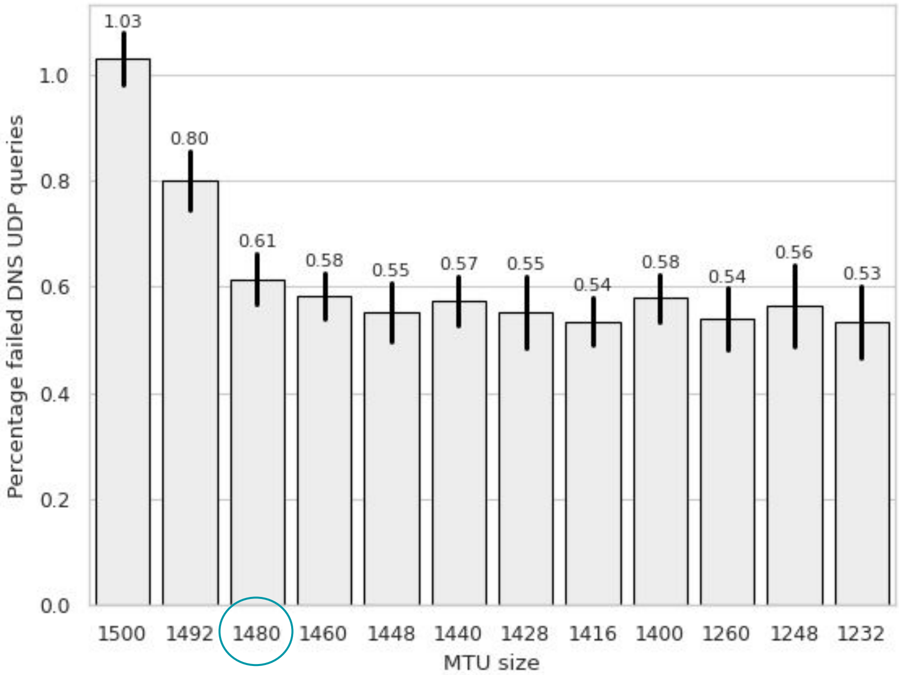
We aggregate our results from the Atlas API and dnstap logs

Methodology



IPv4	IPv6
1500	1500
1492	1492
1480	1460
1460	1448
1448	1412
1440	1380
1428	1360
1416	1340
1400	1320
1260	1300
1248	1280
1232	1268

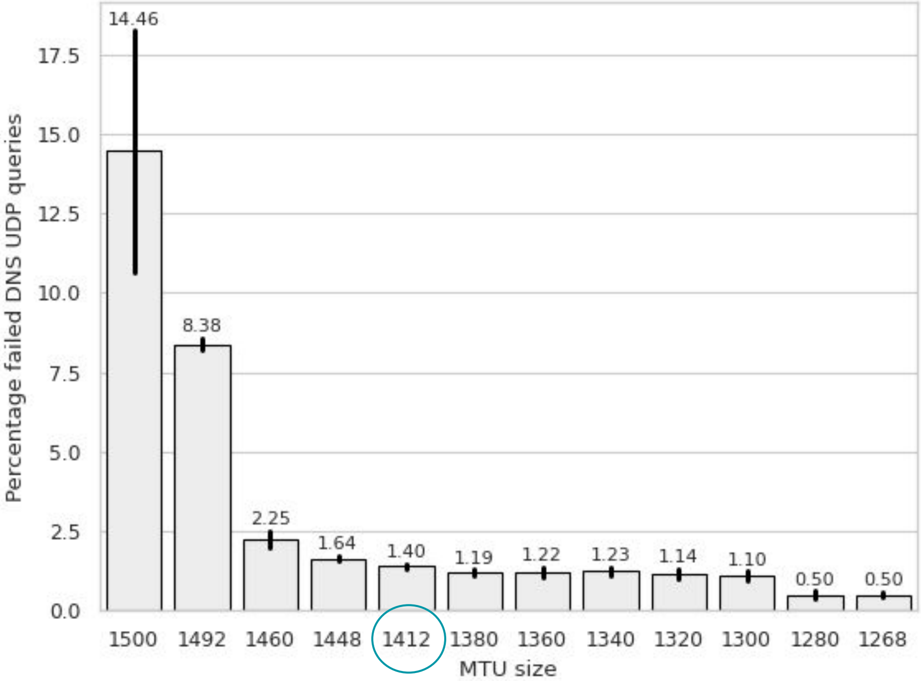
Results IPv4 Stub Resolver



Note: this is the EDNS message size, so MTU minus IP and UDP headers

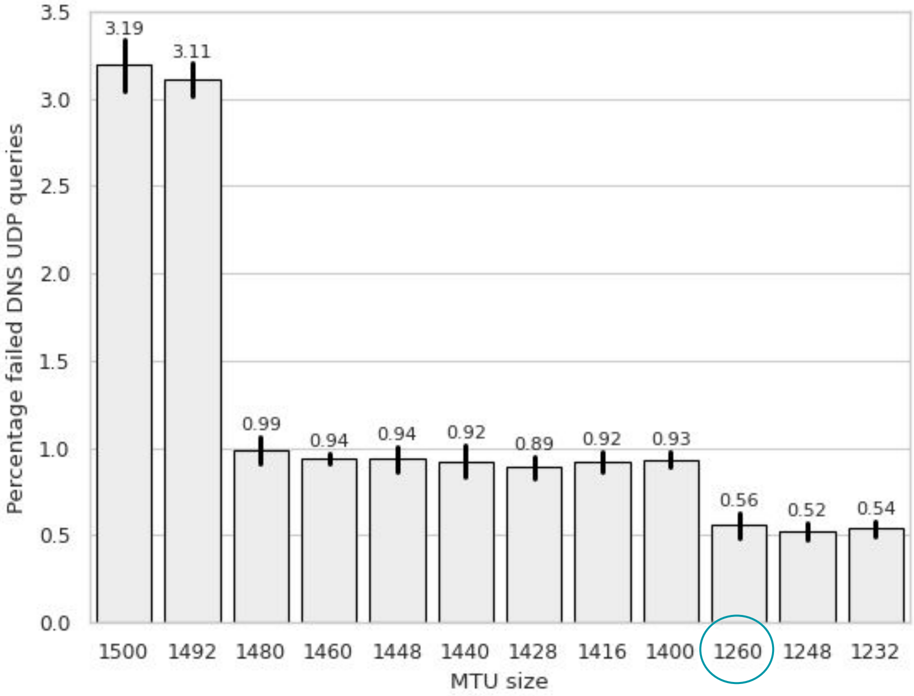
	Stub	**Resolver**
IPv4	1452	
IPv6		

Results IPv6 Stub Resolver



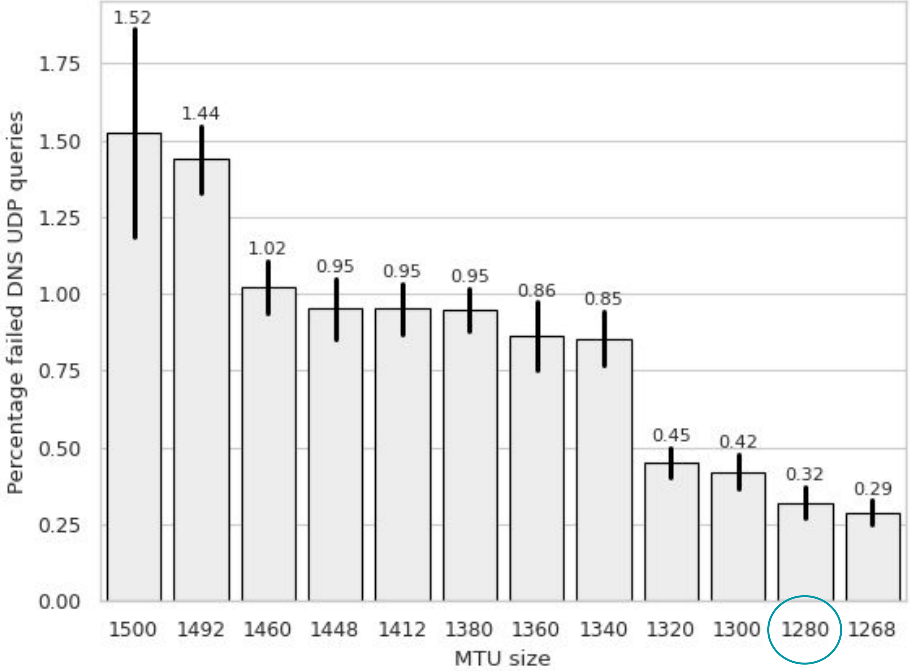
	Stub	Resolver
IPv4	1452	
IPv6	1364	

Results open IPv4 Resolver



	Stub	Resolver
IPv4	1452	1232
IPv6	1364	

Results open IPv6 Resolver

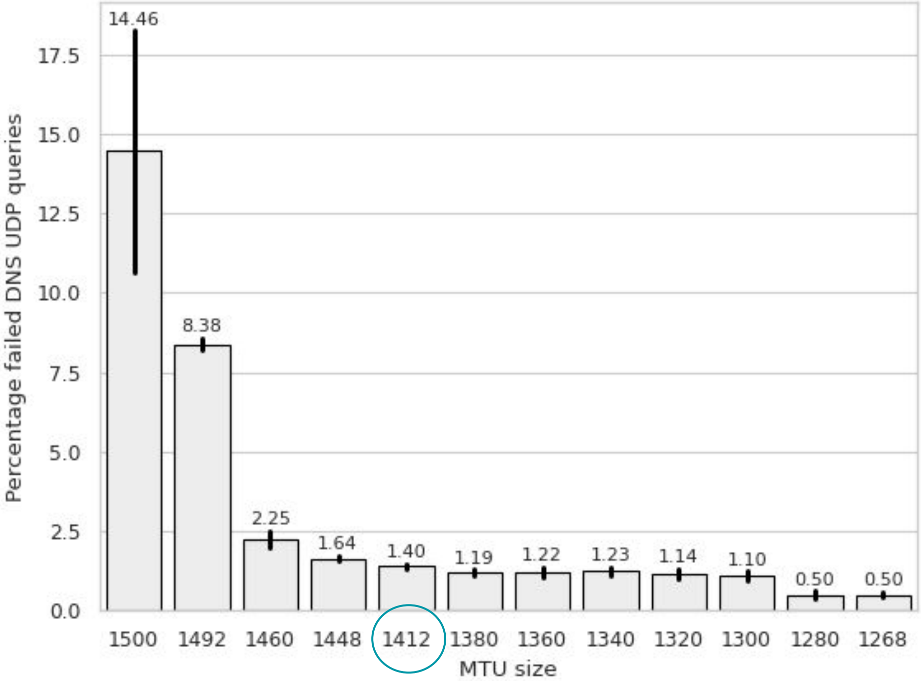


	Stub	Resolver
IPv4	1452	1232
IPv6	1364	1232

Discussion - Results

- MTUs 1500 & 1492 stand out
- IPv6 Stub
- IPv4/6 Resolvers

Results IPv6 Stub Resolver

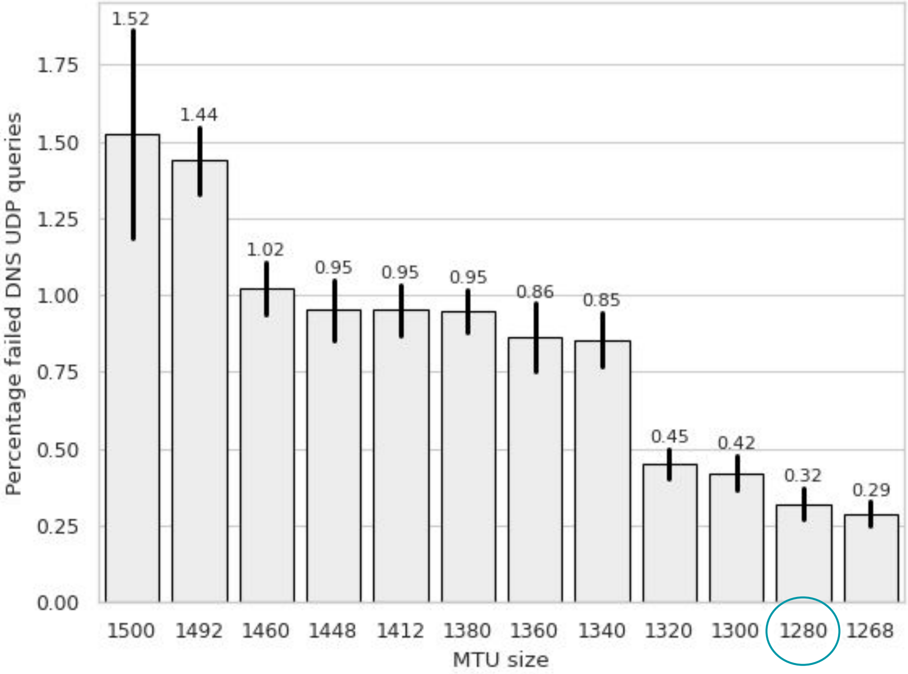


	Stub	Resolver
IPv4	1452	
IPv6	1364	

Discussion - Results

- MTUs 1500 & 1492
- IPv6 Stub
- IPv4/6 Resolvers

Results open IPv6 Resolver



	Stub	**Resolver**
IPv4	1452	1232
IPv6	1364	1232

Discussion - Limitations

- MTU support Digital Ocean
- Dynamic paths
- Failing probes
- RIPE Atlas bias

Conclusion

- Created publicly available reproducible environment [6]
- EDNS(0) message sizes

	Stub	Resolver
IPv4	1452	1232
IPv6	1364	1232

Conclusion

- Created publicly available reproducible environment [6]
- EDNS(0) message sizes

	Stub	**Resolver**
IPv4	1452	1232
IPv6	1232	1232

Conclusion

- Created publicly available reproducible environment [6]
- EDNS(0) message sizes

	Stub	Resolver
IPv4	1452	1452
IPv6	1364	1412

Future Work

- Spread of probes within ASs
- Failing probes
- Continuation

There is no single “magical” EDNS(0) message size for all DNS resolver implementations.

Special thanks to Willem Toorop from NLnet Labs for all his help.

References

[1] - Weaver, N., Kreibich, C., Nechaev, B., & Paxson, V. (2011, April). Implications of Netalyzr’s DNS measurements. In *Proceedings of the First Workshop on Securing and Trusting Internet Names (SATIN), Teddington, United Kingdom*.
[2] - Van Den Broek, G., van Rijswijk-Deij, R., Sperotto, A., & Pras, A. (2014). DNSSEC meets real world: dealing with unreachability caused by fragmentation. *IEEE communications magazine*, *52*(4), 154-160.
[3] - Toroop. (2013) https://medium.com/nlnetlabs/using-pmtud-for-a-higher-dns-responsiveness-60e129917665
[4] - OARC. https://www.dns-oarc.net/oarc/services/replysizetest
[5] - Fujiwara & Vixie. (2020) Fragmentation Avoidance in DNS
[6] - https://github.com/shoaloak/defragDNS