

Using Path MTU Discovery (PMTUD) for better IPv6 DNS responsiveness

Willem Toorop

Willem@NLnetLabs.nl



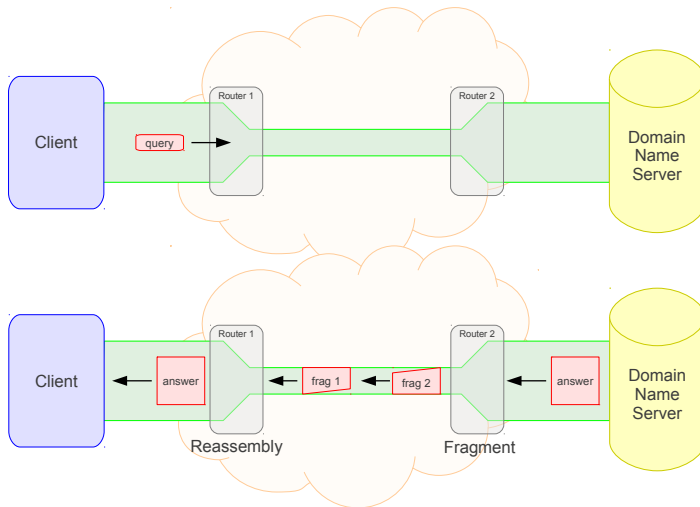
16 October 2013

What is this about?

- ▶ MTU : Maximum Transmission Unit (on a link)
 - ▶ PMTU : Maximum Transmission Unit on a Path
= The smallest MTU on that path.
 - ▶ PMTUD: Path MTU Discovery
-
- ▶ Follow up of UvA student projects at NLnet Labs:
 - ▶ M. de Boer, J. Bosma,
"Discovering Path MTU black holes on the Internet using RIPE Atlas"
(July 2012)
 - ▶ Research performed early this year by UvA Students
 - ▶ H. Bagheri, V. Boteanu,
"Making do with what we've got:
Using PMTUD for a higher DNS responsiveness" (February 2013)

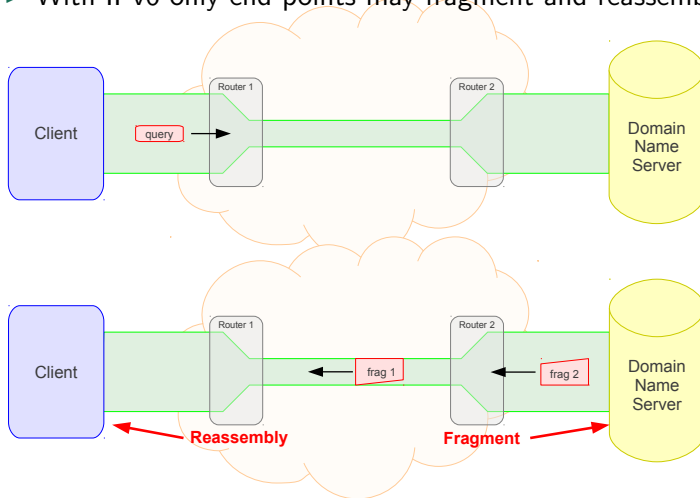
What is this about?

- With IPv4 fragmentation was handled by the network



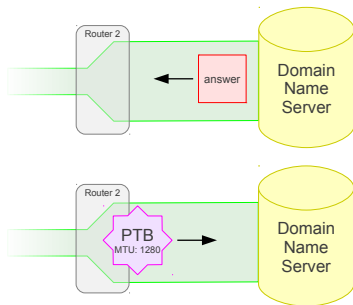
What is this about?

- ▶ With IPv4 fragmentation was handled by the network
- ▶ With IPv6 only end-points may fragment and reassemble



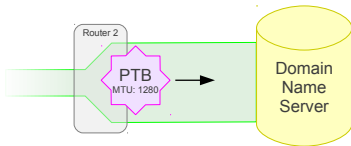
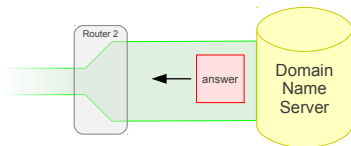
What is this about?

- ▶ With IPv6 only end-points may fragment and reassemble
- ▶ But currently DNS servers do not handle Packet-Too-Big



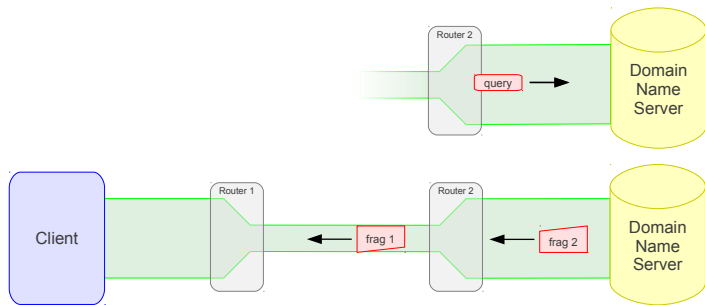
What is this about?

- ▶ With IPv6 only end-points may fragment and reassemble
- ▶ But currently DNS servers do not handle Packet-Too-Big
- ▶ The OS caches PMTU for 10 minutes, or so...
- ▶ and requery happens after 5 seconds, or so...



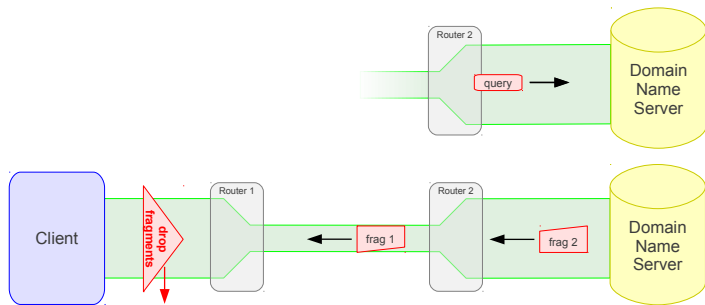
What is this about?

- ▶ With IPv6 only end-points may fragment and reassemble
- ▶ But currently DNS servers do not handle Packet-Too-Big
- ▶ [draft-andrews-dnsex-udp-fragmentation-01](#)



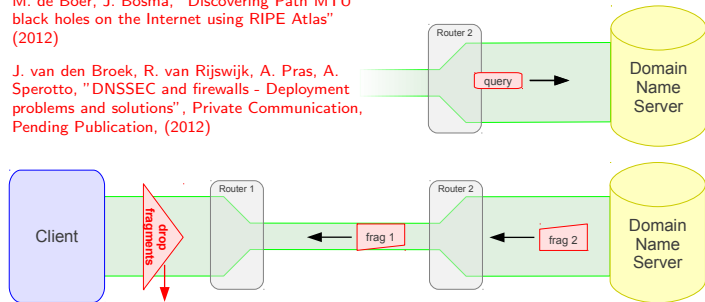
What is this about?

- ▶ With IPv6 only end-points may fragment and reassemble
- ▶ But currently DNS servers do not handle Packet-Too-Big
- ▶ draft-andrews-dnsex-udp-fragmentation-01
- ▶ But then messages in size range 1232-1452 *packet size* 1280–1500 will be fragmented too!



What is this about?

- ▶ With IPv6 only end-points may fragment and reassemble
- ▶ But currently DNS servers do not handle Packet-Too-Big
- ▶ draft-andrews-dnsex-udp-fragmentation-01
- ▶ But then messages in size range 1232-1452 *packet size* 1280–1500 will be fragmented too!
- ▶ **And $\pm 10\%$ of all end-points/resolvers discard IPv6 fragments!**
 - ▶ M. de Boer, J. Bosma, "Discovering Path MTU black holes on the Internet using RIPE Atlas" (2012)
 - ▶ J. van den Broek, R. van Rijswijk, A. Pras, A. Sperotto, "DNSSEC and firewalls - Deployment problems and solutions", Private Communication, Pending Publication, (2012)



[illegible]

- ICM
com
pac
the

```

Router IPv6
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|Version| Traffic Class |                               Flow Label                               |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Payload Length = 1240                               | Next = ICMP | Hop Limit |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
/                               Source Address = Router's IPv6 address                               /
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
/                               Destination Address = Name server's IPv6 address                               /

```

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Type = PTB | Code = 0 | Checksum |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| MTU |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

Domain Name Server IPv6	+-----				
-------------------------------	--	--	--	--	--

UDP Header	
Source port	Destination port
Length	Checksum

[illegible]

What is this about?

- ▶ ICMPv6 Error Messages contain as much of invoking packet as possible without the ICMPv6 packet size exceeding 1280
- ▶ Utilizing ICMPv6 PTB messages to send bigger unfragmented answers (in the 1232-1452 range)
- ▶ Increase DNS responsiveness

Router IPv6

```

+-----+-----+-----+-----+-----+-----+-----+-----+
|Version| Traffic Class |                               Flow Label |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Payload Length = 1240 | Next = ICMP | Hop Limit |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Source Address = Router's IPv6 address |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Destination Address = Name server's IPv6 address |
+-----+-----+-----+-----+-----+-----+-----+-----+
  
```

ICMPv6
Error msg

```

+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Type = PTB | Code = 0 | Checksum |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               MTU |
+-----+-----+-----+-----+-----+-----+-----+-----+
  
```

Domain
Name Server
IPv6

```

+-----+-----+-----+-----+-----+-----+-----+-----+
|Version| Traffic Class |                               Flow Label |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Payload Length | Next = UDP | Hop Limit |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Source Address = Name server IPv6 address |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Destination Address = Requester's IPv6 address |
+-----+-----+-----+-----+-----+-----+-----+-----+
  
```

UDP Header

```

+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Source port | Destination port |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Length | Checksum |
+-----+-----+-----+-----+-----+-----+-----+-----+
  
```

Beginning of
Answer

```

+-----+-----+-----+-----+-----+-----+-----+-----+
|                               ID | R|Opcode | A|C|D|A|Z|D|D| RCODE |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               QDCOUNT | ANCOUNT |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               NSCOUNT | ARCOUNT |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Query |
+-----+-----+-----+-----+-----+-----+-----+-----+
  
```

Observations

- Bypass BCP38: Anyone can spoof a source address.

Router IPv6

```

+-----+-----+-----+-----+-----+-----+-----+-----+
|Version| Traffic Class |                               Flow Label |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Payload Length = 1240 | Next = ICMP | Hop Limit |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Source Address = Router's IPv6 address |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Destination Address = Name server's IPv6 address |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

ICMPv6
Error msg

```

+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Type = PTB | Code = 0 | Checksum |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               MTU |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Domain
Name Server
IPv6

```

+-----+-----+-----+-----+-----+-----+-----+-----+
|Version| Traffic Class |                               Flow Label |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Payload Length | Next = UDP | Hop Limit |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Source Address = Name server IPv6 address |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Destination Address = Requester's IPv6 address |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

UDP Header

```

+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Source port | Destination port |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Length | Checksum |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Beginning of
Answer

```

+-----+-----+-----+-----+-----+-----+-----+-----+
|                               ID | R|Opcode | A|C|D|A|Z|D|D| RCODE |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               QDCOUNT | ANCOUNT |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               NSCOUNT | ARCOUNT |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Query |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Observations

- ▶ Bypass BCP38: Anyone can spoof a source address.
- ▶ Simply re-inject with TC bit: No! (cache poisoning)
- ▶ So re-evaluate query at Domain Name Server (or resubmit spoofing the source)

Router IPv6

Version	Traffic Class	Flow Label
Payload Length = 1240	Next = ICMP	Hop Limit
Source Address = Router's IPv6 address		
Destination Address = Name server's IPv6 address		

ICMPv6 Error msg

Type = PTB	Code = 0	Checksum
MTU		

Domain Name Server IPv6

Version	Traffic Class	Flow Label
Payload Length	Next = UDP	Hop Limit
Source Address = Name server IPv6 address		
Destination Address = Requester's IPv6 address		

UDP Header

Source port	Destination port
Length	Checksum

Beginning of Answer

ID	R Opcode	A C D A Z D D	R CODE
QDCOUNT	ANCOUNT		
NSCOUNT	ARCOUNT		
Query			

Observations

- ▶ Bypass BCP38: Anyone can spoof a source address.
- ▶ Simply re-inject with TC bit: No! (cache poisoning)
- ▶ So re-evaluate query at Domain Name Server (or resubmit spoofing the source)
- ▶ What message size is client willing to receive?
- ▶ Original EDNS0 is lost

Router IPv6

Version	Traffic Class	Flow Label
Payload Length = 1240	Next = ICMP	Hop Limit
Source Address = Router's IPv6 address		
Destination Address = Name server's IPv6 address		

ICMPv6 Error msg

Type = PTB	Code = 0	Checksum
MTU		

Domain Name Server IPv6

Version	Traffic Class	Flow Label
Payload Length	Next = UDP	Hop Limit
Source Address = Name server IPv6 address		
Destination Address = Requester's IPv6 address		

UDP Header

Source port	Destination port
Length	Checksum

Beginning of Answer

ID	R Opcode	A C D A Z D D	R CODE
QDCOUNT	ANCOUNT		
NSCOUNT	ARCOUNT		
Query			

Observations

- ▶ Bypass BCP38: Anyone can spoof a source address.
- ▶ Simply re-inject with TC bit: No! (cache poisoning)
- ▶ So re-evaluate query at Domain Name Server (or resubmit spoofing the source)
- ▶ What message size is client willing to receive?
- ▶ Original EDNS0 is lost
- ▶ Payload length from UDP: No! (amplification attack)
- ▶ So, set EDNS0 udp size to ICMPv6 packet size - 48

Router IPv6

Version	Traffic Class	Flow Label
Payload Length = 1240	Next = ICMP	Hop Limit
Source Address = Router's IPv6 address		
Destination Address = Name server's IPv6 address		

ICMPv6 Error msg

Type = PTB	Code = 0	Checksum
MTU		

Domain Name Server IPv6

Version	Traffic Class	Flow Label
Payload Length	Next = UDP	Hop Limit
Source Address = Name server IPv6 address		
Destination Address = Requester's IPv6 address		

UDP Header

Source port	Destination port
Length	Checksum

Beginning of Answer

+-----															
--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Tests and Measurements

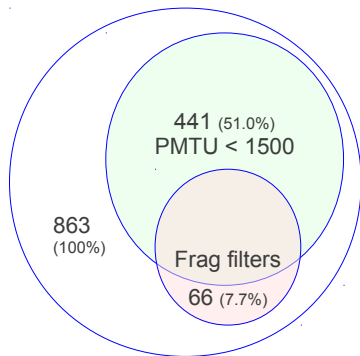
- RIPE ATLAS to query messages from 863 probes

measurement	message size	packet size
baseline	1280	1280
fragment filters	1600	1280
PMTU	1600	1500

Tests and Measurements

- ▶ RIPE ATLAS to query messages from 863 probes

measurement	message size	packet size	# answered	
baseline	1280	1280	863	100.0%
fragment filters	1600	1280	795	92.3%
PMTU	1600	1500	422	49.0%

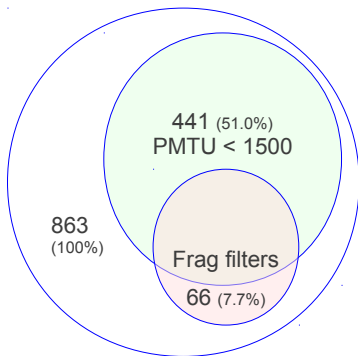


1280	115	1452	2
1300	1	1456	2
1398	1	1460	3
1400	4	1464	3
1418	1	1468	1
1420	1	1472	8
1424	1	1476	6
1428	1	1480	169
1434	3	1488	1
1440	3	1492	76
1450	4	1500	7

Tests and Measurements

- ▶ RIPE ATLAS to query messages from 863 probes

measurement	message size	packet size	# answered	
baseline	1280	1280	863	100.0%
fragment filters	1600	1280	795	92.3%
PMTU	1600	1500	422	49.0%



ICMPv6 type	#	rtt
address unreachable	2	0.03
administratively prohibited	18	0.03
reassembly time exceeded	13	60.09
Packet Too Big	9	0.07

Observation:

- ▶ 18 out of 80 send administratively prohibited

Tests and Measurements

- RIPE ATLAS to query messages from 863 probes

measurement	message size	packet size	# answered	
baseline	1280	1280	863	100.0%
fragment filters	1600	1280	795	92.3%
PMTU	1600	1500	422	49.0%

with proof of concept program running

828 probes	1500	1500	805	97.2%
------------	------	------	-----	-------

Tests and Measurements

- Last week from 1059 the same probes
Each queried more than 10 times

measurement	message size	packet size	# answered	
baseline	small	-	1059	100.00%
fragment filters	1600	1280	986	93.11%
PMTU	1500	1500	587	55.43%
With PMTUD	1500	1500	1044	98.58%

Tests and Measurements

- ▶ Last week from 1059 the same probes
Each queried more than 10 times

measurement	message size	packet size	# answered	
baseline	small	-	1059	100.00%
fragment filters	1600	1280	986	93.11%
PMTU	1500	1500	587	55.43%
With PMTUD	1500	1500	1044	98.58%

- ▶ Number of probes behind fragment filters in time

	# probes	% filtered
July 2012	500	10.0%
June 2013	863	7.7%
October 2013	1059	6.9%

Relevance: Real world capture analysis

	<i>SIDN</i>	<i>SURFnet</i>
answers 1232-1500	34	1763
answers > 1500	3999	1278
fragmented answers	3999	1632
Packet-Too-Bigs	41	16
administratively prohibited	67	2
reassembly time exceeded	333	26

Counting ICMPv6 Messages only when the payload is

- ▶ UDP with size > 1232
- ▶ A first fragment containing UDP

Relevance: Real world capture analysis

	<i>SIDN</i>	<i>SURFnet</i>
answers 1232-1500	34	1763
answers > 1500	3999	1278
fragmented answers	3999	1632
Packet-Too-Bigs	41	16
administratively prohibited	67	2
reassembly time exceeded	333	26
<i>lost answers prediction</i>		
extrapolate admin prohibit	298	8
extrapolate time exceeded	2049	160

Final remarks

- ▶ TODO: Same measurements from probe's network resolver
- ▶ TODO: Structural tracking of PMTU and fragment problems
- ▶ For more info, the student report and working Proof-Of-Concept implementation see blog entry at

<http://www.nlnetlabs.nl/blog/2013/06/04/pmtud4dns/>