

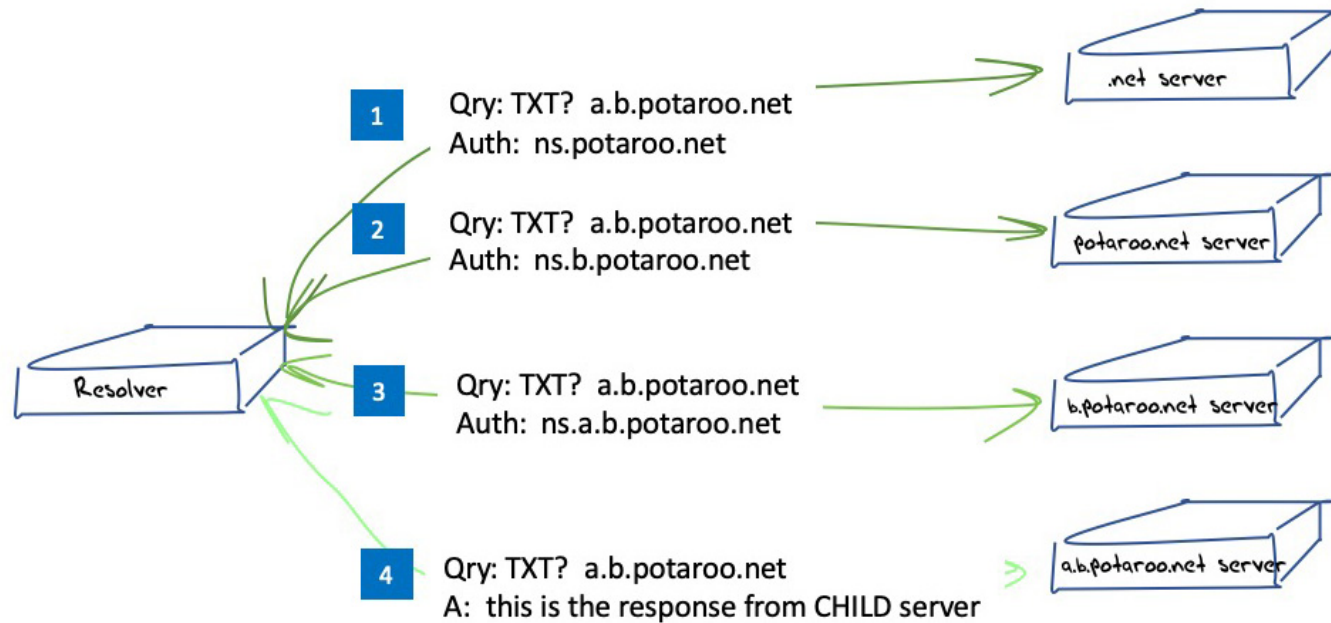
# Measuring Query Name Minimization

Geoff Huston  
Joao Damas

APNIC Labs  
August 2020

# Quick Summary

## NON-query name minimisation resolution sequence



# Quick Summary

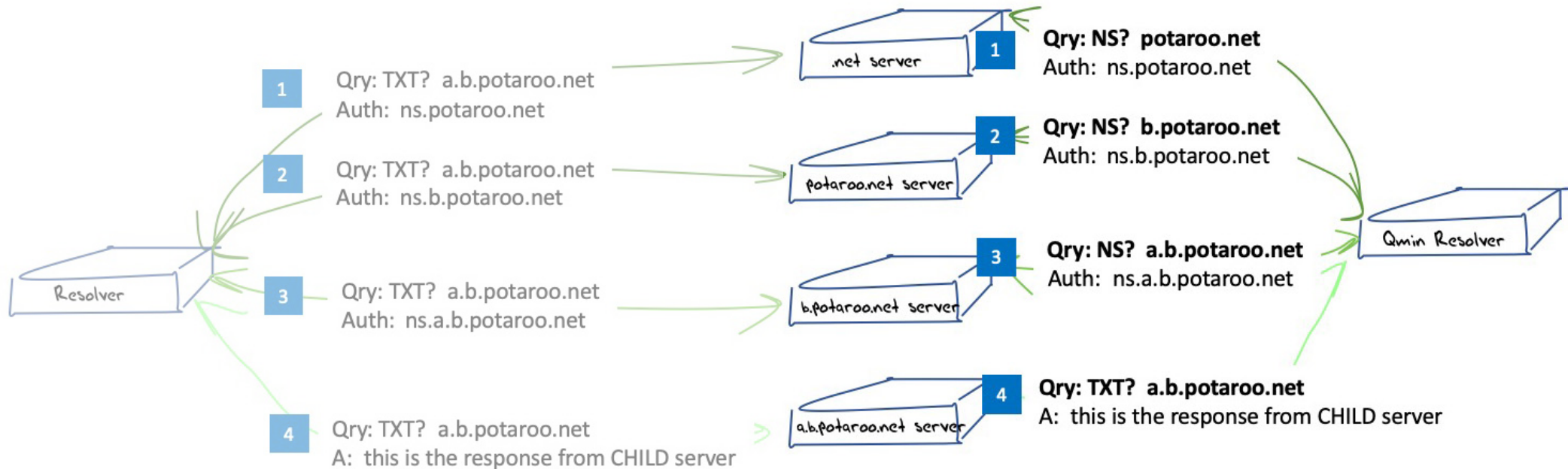
## Query name minimisation technique described in RFC 7816

Instead of sending the full QNAME and the original QTYPE upstream, a resolver that implements QNAME minimisation and does not already have the answer in its cache sends a request to the name server authoritative for the closest known ancestor of the original QNAME. The request is done with:

- o the QTYPE NS
- o the QNAME that is the original QNAME, stripped to just one label more than the zone for which the server is authoritative

# Quick Summary

Query name minimisation technique described in RFC 7816



# Common Resolver Implementation Status

- BIND 9
  - Implemented in 9.14, active in “relaxed” mode by default
- Unbound
  - Implemented in 1.7.2, active in “non-strict” mode
- Knot
  - Implemented in 1.2.2, active by default
- Power DNS Recursor
  - Implemented in 4.3.0-alpha1, enabled by default since 4.3.0-beta 1

# Common Resolver Implementation Status

- BIND 9

- Implemented in 9.14, active in "rel-

- Unbound

- In

- *it looks like all recursive resolvers should be doing query name minimisation these days.*

- P

*Right?*

ursor

default

implemented in 4.3.0-alpha1, enabled by default since 4.3.0-beta 1

# Measurements

We will look at adoption from the perspectives of:

Users and

Resolvers

Users whose Queries are handled with  
Qname Minimization



# User Measurements

## 2019 Results

Experiments	Qmin	Query Type			
		NS	A	AAAA	
429,773,288	11,089,823	2,811,053	8,336,008	1,721	
	3%	1%	2%	0%	<b>% of all experiments</b>
		25%	75%	0%	<b>% of Qmin experiments</b>

# User Measurements

## 2019 Results

Experiments	Qmin	Query Type			
		NS	A	AAAA	
429,773,288	11,089,823	2,811,053	8,336,008	1,721	
	3%	1%	2%	0%	% of all experiments
		25%	75%	0%	% of Qmin experiments

## 2020 Results

Experiments	Qmin	Query Type			
		NS	A	AAAA	
17,370,478	3,136,177	238,903	2,909,184	0	
	18%	1%	17%	0%	% of all experiments
		8%	93%	0%	% of Qmin experiments

# User Measures

The proportion of users who use recursive resolvers that perform Query Name minimization has risen from 3% of users to 18% of users in the past 12 months.

The common resolver behaviour is to perform the discovery queries using query type A, not NS or AAAA

# Where are these Users?

CC	Qmin Ratio	Experiments	Qmin	CC Name
NE	71%	20,020	14,275	Niger
BW	70%	4,478	3,114	Botswana
CY	69%	4,550	3,120	Cyprus
IR	63%	266,671	169,234	Iran
MV	61%	1,627	985	Maldives
NP	56%	30,706	17,073	Nepal
IN	55%	2,533,754	1,388,757	India
GM	52%	2,321	1,203	Gambia
PT	47%	33,931	15,899	Portugal
ZW	45%	20,466	9,250	Zimbabwe
CG	42%	2,276	953	Congo
FR	41%	225,958	93,004	France
ZA	40%	150,206	60,289	South Africa
BY	40%	32,183	12,818	Belarus
EC	38%	48,153	18,431	Ecuador
AF	36%	22,971	8,330	Afghanistan
NZ	33%	18,932	6,248	New Zealand
FI	29%	20,336	5,910	Finland
GA	29%	5,103	1,481	Gabon
GH	29%	57,957	16,768	Ghana


# Resolver Measures

## What's a “resolver”?

- Always hard to tell these days.
- Over a 3 day period we saw 48,191 distinct IP addresses of resolvers
  - 23,728 IPv4 addresses  
7,249 distinct /24 subnets
  - 24,463 IPv6 addresses  
6,549 distinct /48 subnets


Did we observe 48,191 resolvers or somewhere around 8,000 distinct resolvers?

# Open Resolvers



Open DNS Resolver	Qmin Ratio	Experiments	Qmin Experiments
Google DNS	0%	21,464,119	0
Cloudflare	50%	1,664,811	832,671
dnspai	0%	1,103,837	44
Open DNS	69%	1,097,567	756,414
oneDNS	0%	584,099	9
114dns	1%	295,734	1,959
Verisign	0%	279,524	0
Quad9	70%	216,629	152,121
Neustar	59%	117,956	69,597
DNS Watch	56%	45,271	25,532
Oracle Dyn	59%	34,265	20,150
Hurricane Electric	98%	10,015	9,840
Yandex	0%	2,027	0
Uncensored DNS	0%	1,223	0

What's behind these 50%-70% ratios? Is Qmin only partially deployed in the DNS service anycast constellation?



This is more expected!

# ISP Resolvers

ASN	Qmin Ratio	Experiments	Qmin Experiments	ASN Name	CC
4134	9%	13,197,623	1,249,992	China Telecom	CN
55836	58%	7,153,342	4,172,335	Reliance Jio	IN
4837	7%	2,702,633	177,075	CHINA Unicom	CN
9808	12%	2,178,630	252,338	Guangdong Mobile	CN
9498	0%	2,100,413	0	BHARTI Airtel	IN
58543	0%	1,981,946	0	China Telecom Guangd	CN
7922	0%	1,326,612	123	COMCAST	US
56046	56%	1,296,970	722,674	Jiangsu Mobile	CN
6730	50%	1,243,256	624,727	SUNRISE	CH
24560	0%	814,734	0	Bharti Airtel Broadband	IN
30986	32%	774,841	250,644	SCANCOM	GH
4835	56%	730,662	405,636	China Telecom	CN
7552	0%	615,492	0	Viettel	VN
28573	0%	549,425	8	CLARO	BR
7018	0%	544,352	22	AT&T	US
12322	60%	505,703	302,042	PROXAD	FR
8151	0%	479,355	6	Uninet	MX
17676	2%	470,505	10,448	Softbank BB	JP
22394	0%	446,925	0	CELLCO	US
56040	0%	413,381	0	Guangdong Mobile	CN

# Observations

- Query name minimisation is gathering momentum in the past 12 months (3% of users in mid 2019 to 18% of users in mid-2020)
- While all common vendor code has enabled Query name minimisation, enabling this behaviour in ISP and open resolvers is fragmentary
  - Why is it not deployed? What's the concern?



# Questions

- Where and why is Query Name minimisation important? Does it differ by scale?
  - Small scale recursive resolvers at the edge of the network?
  - ISP-operated recursive resolvers?
  - Open recursive resolvers?
- Is the query name alone a privacy threat or is the combination of the recursive resolver with the query name the problem?
- Are there residual issues with handling of empty non-terminals?

Thanks!