How to explore EDNS-Client-Subnet Supporters in your Free Time

DENOG5, Darmstadt

Florian Streibelt

<florian@inet.tu-berlin.de>

TU-Berlin, Germany - FG INET

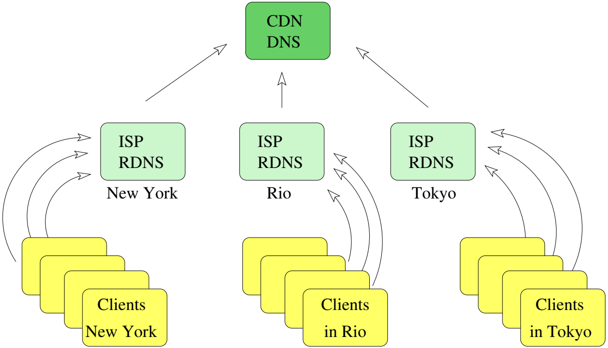
www.inet.tu-berlin.de

November 14th 2013

Florian Streibelt, Jan Böttger, Nikolaos Chatzis,
Georgios Smaragdakis, Anja Feldmann

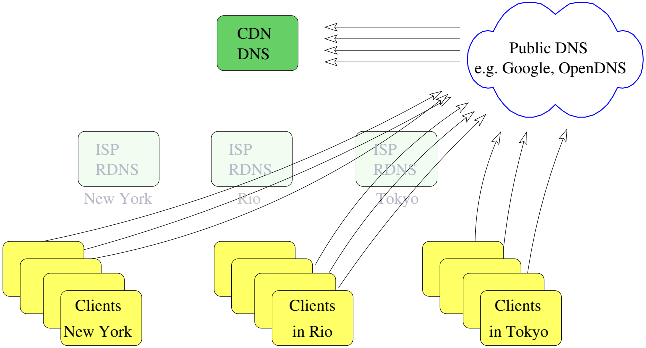
With special thanks to Walter Willinger.

Using DNS for client location



- Clients use ISP nameservers
- Distance between client and RDNS is relatively low
- Client location inferred from source IP of request

Non-ISP (aka 'public') DNS usage increases



Usage at 8.6% in December 2011

According to Otto et al. in "Content delivery and the natural evolution of DNS: remote DNS trends, performance issues and alternative solutions" (IMC 2012)

Challenge for CDNs/CPs

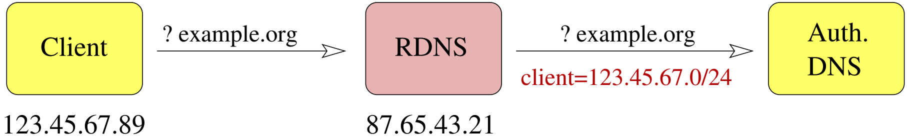
- Non-ISP resolvers are gaining momentum
- Clients are far away from resolvers
- CDNs often make heavy use of DNS for client location
- Using the DNS request origin for client-location now leads to (more) wrong results
- Mis-location of clients gives end-users bad performance

Introducing: Client IP information in EDNS (ECS)

- Recursive nameserver adds client subnet information (network prefix) to the query directed at the authoritative nameserver
- EDNS0 extension is introduced to transport this data
- Note: Do not confuse EDNS with DNSSEC - EDNS is the underlying extension mechanism
- Proposal by Google, OpenDNS and others
  (A faster Internet consortium)
- Performance gain can be observed, again see Otto et al. (IMC 2012)
- We find roughly 13% of the top 1M Alexa list seem to support this extension already

Use of ECS

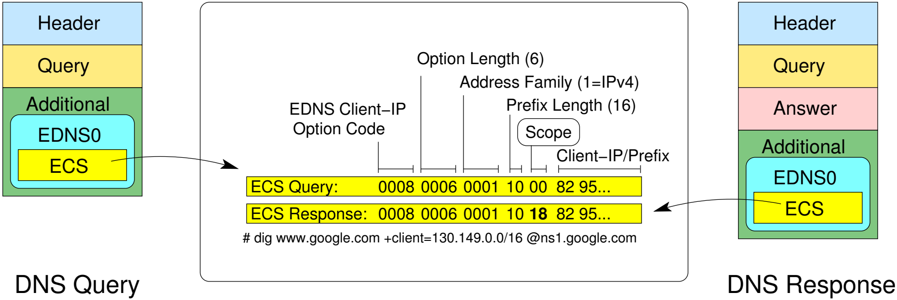
Intended use of ECS:



How to enable ECS?

- Authoritative nameservers must be ECS enabled (Supported by e.g., PowerDNS but not Bind, Unbound)
- If there are other systems in front: these as well
- Not all vendors of DNS appliances publicly announce this as a feature
- Primary nameservers need to be whitelisted (manually) by e.g., OpenDNS, Google
- For debugging, a patched version of dig and python libs exist

Protocol: Client IP information in EDNS (ECS)



ECS Query: 0008 0006 0001 10 00 82 95...

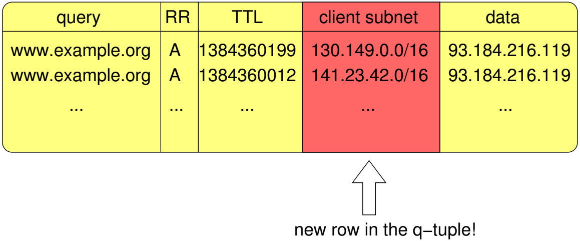
ECS Response: 0008 0006 0001 10 **18** 82 95...

```
# dig www.google.com +client=130.149.0.0/16 @ns1.google.com
```

- The scope returned allows for caching (applied as netmask)
- The client IP information cannot be checked

Protocol: ECS Caching

Simple abstraction of a DNS-Cache:



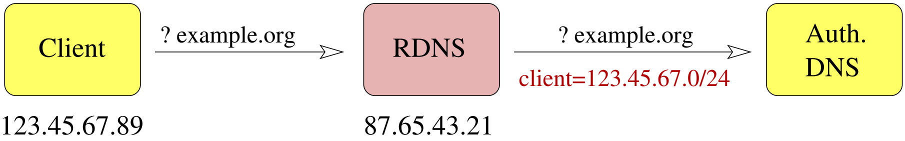
query	RR	TTL	client subnet	data
www.example.org	A	1384360199	130.149.0.0/16	93.184.216.119
www.example.org	A	1384360012	141.23.42.0/16	93.184.216.119
...	...	...	...	...

new row in the q–tuple!

- The scope returned is applied as netmask
- A caching resolver saves this network prefix with the answer
- Clients in the same 'subnet' get the cached answer
- Other clients trigger a new request with their subnet

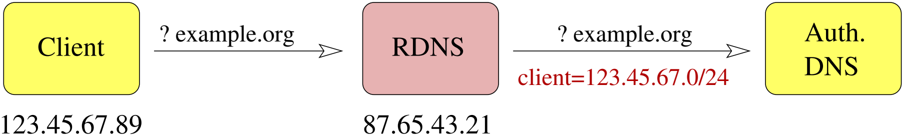
(Ab)using ECS for Measurements

Intended use of ECS:



(Ab)using ECS for Measurements

Intended use of ECS:



Client

123.45.67.89

? example.org

RDNS

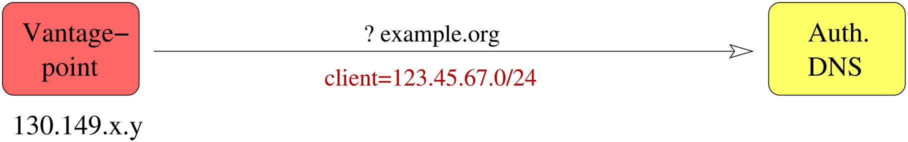
87.65.43.21

? example.org

client=123.45.67.0/24

Auth. DNS

Doing our measurements:



Vantage–point

130.149.x.y

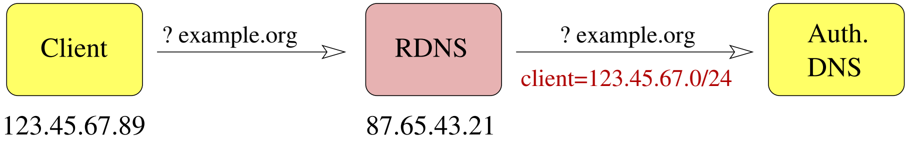
? example.org

client=123.45.67.0/24

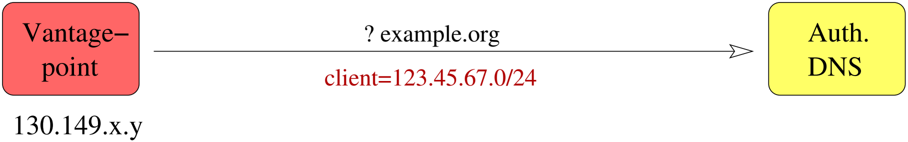
Auth. DNS

(Ab)using ECS for Measurements

Intended use of ECS:



Doing our measurements:



⇒ We can impose every client 'location'.

ECS as a Measurement Tool

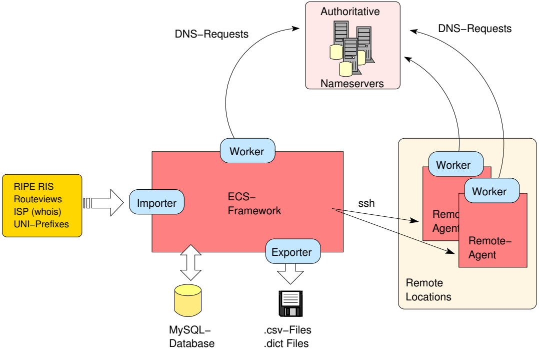
- Using arbitrary client subnet information, we can impose every client 'location'
- This gives us the opportunity to
  - find the location of CDN caches within ISPs,
  - observe the growth of CDN footprints,
  - infer client-to-server mappings (to some extend),
  - analyze dynamic changes by repeated measurements.
- As demonstration we present a subset of our experiments, using Google as example.

Measurements

- One vantage point[1] for *any arbitrary* Client IP/prefix
- We use all network prefixes from RIPE RIS
  (sanity check using Routeviews)
- We compare with Client Subnets derived from:
  popular resolvers, subnets of an ISP, educational networks
- Measurement targets:
  Google/YouTube, MySqueezebox, Edgecast and others
- Data to look at:
  A-records (servers) and scope (caching) returned

[1] we checked from four different locations

Framework used



- Python, mysql, Cymru bulk-interface for AS-lookups
- About 60 Million DNS results, 70 GB data in total
- Performance: 50 DNS requests/sec, full experiment: 2-3 days
- Analysis: typically less than a day

Comparing sources for Client Subnets

	Prefix set	Server IPs	Sub-nets	AS	Countries
Google (03/26/13)	RIPE	6,340	329	166	47
	RV	6,308	328	166	47
	PRES	6,088	313	159	46
	ISP	207	28	1	1
	ISP24	535	44	2	2
	UNI	123	13	1	1

- RIPE RIS and Routeviews give nearly identical results
- The 280k most popular resolvers, as seen by a CDN, yield similar results – but dataset is not freely available
- Mapping to GGCs is working, as can been seen at the UNI and ISP datasets

Looking at the A-Records of Google

- Resolving www.google.com via `ns1.google.com`
- Using all network prefixes from RIPE RIS as client subnets
- Different synchronized vantage points (plausibility check)

Date (RIPE)	IPs	Sub nets	ASes	Countries
2013-03-26	6340	329	166	47
2013-03-30	6495	332	167	47
2013-04-13	6821	331	167	46
2013-04-21	7162	346	169	46
2013-05-16	9762	485	287	55
2013-05-26	9465	471	281	52
2013-06-18	14418	703	454	91
2013-07-13	21321	1040	714	91
2013-08-08	21862	1083	761	123

see also:

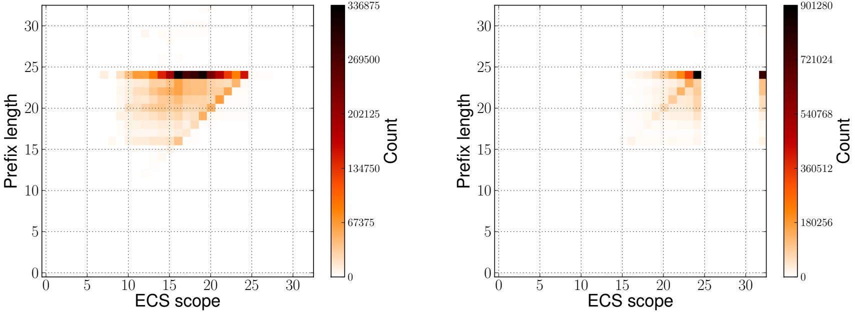
Calder et al.: Mapping the Expansion of Google's Serving Infrastructure, IMC2013

Looking at the A-Records of Google

Selected results from combined experiments:

- We see GGC (Google Global Cache edge servers) in various ISP networks
- ISPs are not allowed to advertise the GGC (we are)
- We observe a huge increase in the footprint, also for YouTube
- Results from different vantage points show redirection of clients and prefixes (load balancing the GGCs?)
- Most of the time clients are served from caches in their respective AS
- A records from the different vantage points mostly overlap, both for Google and YouTube

Comparing Google and Edgecast Scopes



Edgecast (left) aggregates while Google (right) returns more specific scopes.

Conclusion

- ECS gives better performance for clients
- Tradeoff for DNS providers and CDNs:
  it reveals internal information
- Researchers (and competitors) can investigate:
  global footprint, growth-rate, user-to-server mapping, ...
- Filtering of queries was not yet observed
  (e.g. based on number of client prefixes per source IP)
- Information gathered could be used e.g., for DDoS against all nodes of a CDN
- Future Adopters and the community should be aware

Contact:

Florian Streibelt <florian@inet.tu-berlin.de>

Related publication:

Unintended Consequences: Exploring EDNS-Client-Subnet Adopters in your Free Time

Internet Measurement Conference, October 2013

http://conferences.sigcomm.org/imc/2013/

Authors:

Florian Streibelt, Jan Böttger, Nikolaos Chatzis, Georgios Smaragdakis, Anja Feldmann

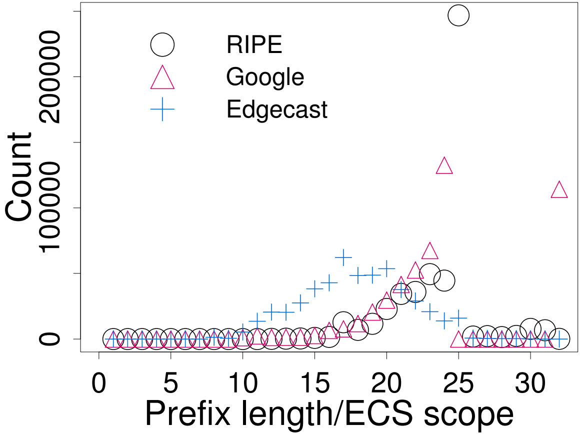
The software and raw data will be published in late November 2013.

http://projects.inet.tu-berlin.de/projects/ecs-adopters/wiki

Image sources:

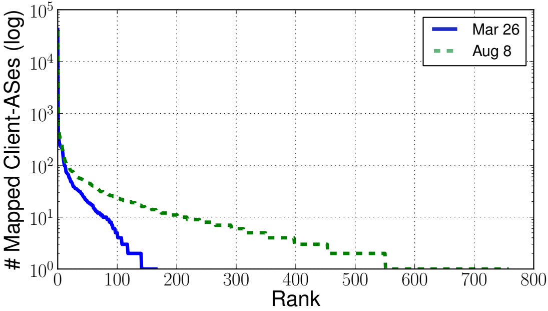
own work and http://openclipart.org/

RIPE RIS prefix length vs. ECS-scopes



Prefix length and scope distribution do not match and differ between adopters, also note the /32s!

Client and AS mappings



In August we see more ASes served from more than one ’server-AS’.