DS and DNSKEY low TTL experiments

Petr Špaček

2023-09-06

pspacek@isc.org

Outline

- TTL is wrong
- TTL does not behave
- Experimental setup
- Test data set
- Evaluation
- Real-world experience – by Viktor Dukhovni

Time To Live – always wrong

- Too short
  - Outage if servers are down
    - see Cache Me If You Can: Effects of DNS Time-to-Live
- Too long
  - Long outages with bad data
  - Slack.com, .NZ, …

Time To Live – full of surprises

- Supposedly an upper bound
- But …
  RFC 8767: Serving Stale Data to Improve Resiliency
- See also
  Measuring TTL Violation of DNS Resolvers at scale
- See also
  The role of DNS in residential Internet use

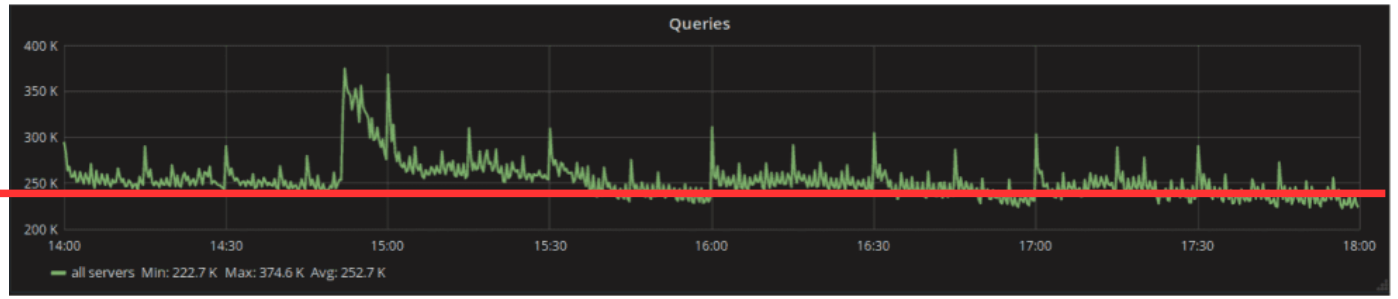
Time To Live – full of surprises #2

- RIPE NCC lowered TTL
  - NS **2 days ⇒ 1 day**
  - DS **1 day ⇒ 1 hour**
  - expecting at least 2x times query load increase …?
  - "There was NO increase in query rates at RIPE NCC's servers"
    - see RIPE NCC's presentation

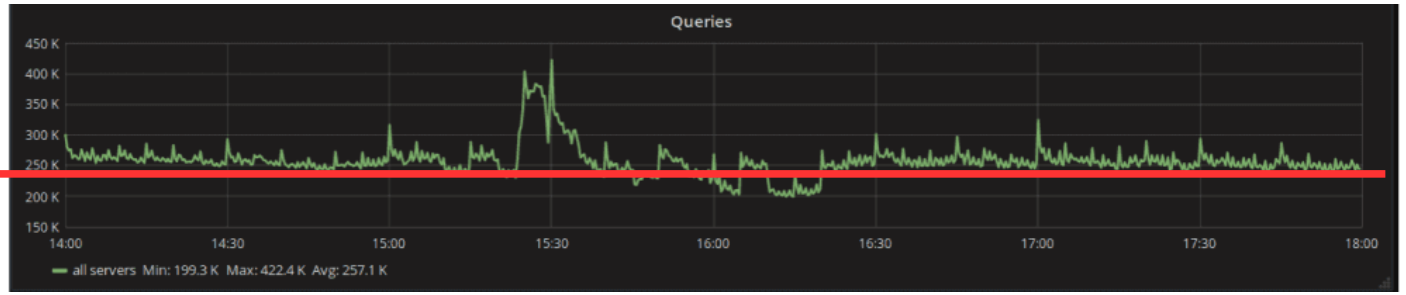
Time To Live – full of surprises #3

- CZ TLD lowered TTL: **5 h ⇒ 1 h**
  - expecting 5x times query load increase …?
  - see article

before - 250 k QPS



after - 250 k QPS !



Lab Experiment

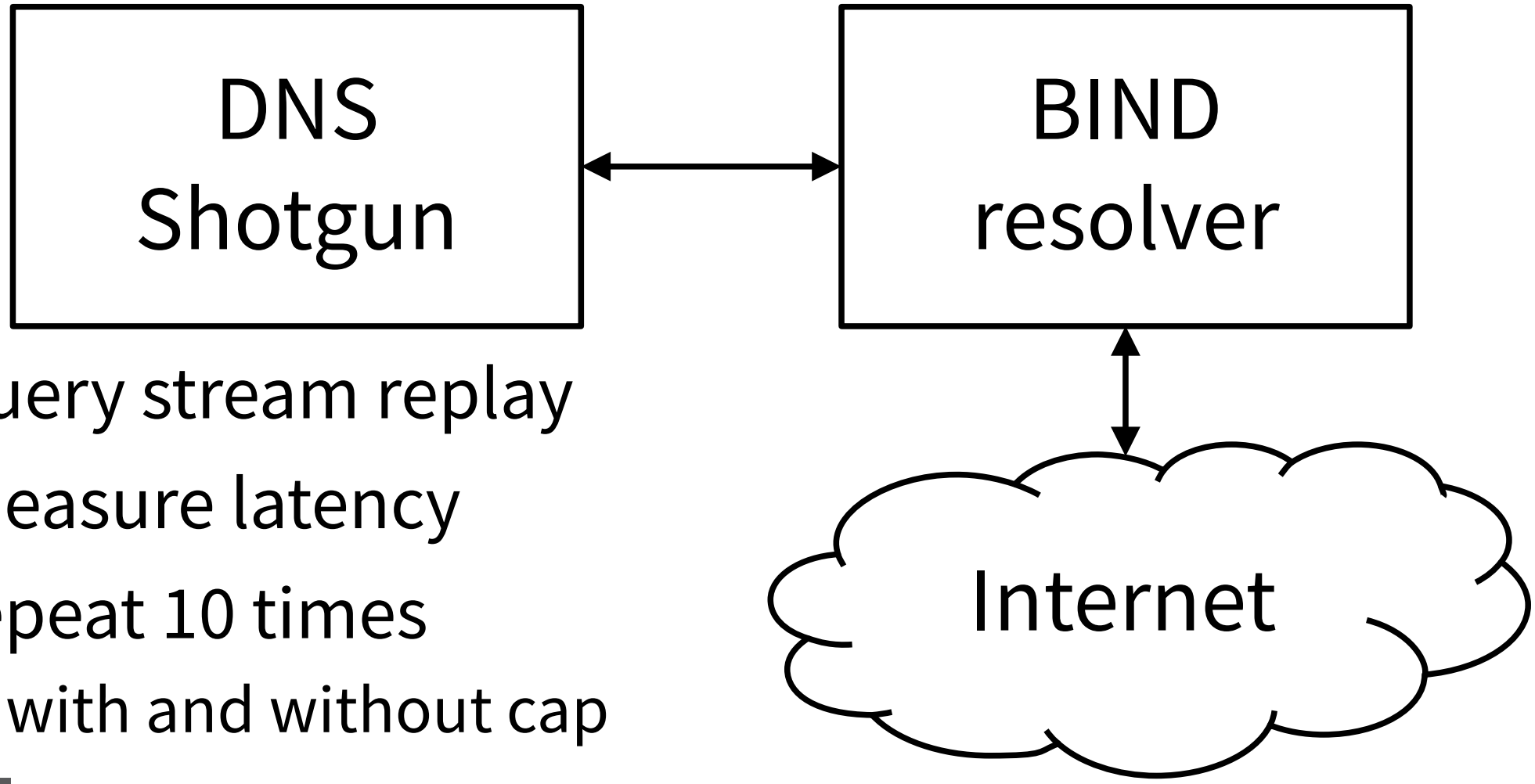
Lab Experiment

- **DS & DNSKEY** TTL cap = 5 minutes
  - other RR types unaffected
- Enforce cap inside resolver side
  - but unlimited cache size
- Evaluate
  - answer latency
  - queries to auths

BIND v9.18.15 hack – TTL cap

```
diff --git a/lib/dns/resolver.c b/lib/dns/resolver.c
index b09813d444..e45d062369 100644
--- a/lib/dns/resolver.c
+++ b/lib/dns/resolver.c
@@ -6255,6 +6255,12 @@ cache_name(fetchctx_t *fctx, dns_name_t *name, dns_message_t
*message,
                /*
                 * Enforce the configure maximum cache TTL.
                 */
+               if ((rdataset->type == dns_rdatatype_ds ||
+                    rdataset->type == dns_rdatatype_dnskey) &&
+                   rdataset->ttl > 5 * 60)
+               {
+                       rdataset->ttl = 5 * 60;
+               }
                if (rdataset->ttl > res->view->maxcachettl) {
                        rdataset->ttl = res->view->maxcachettl;
                }
```

Experimental setup

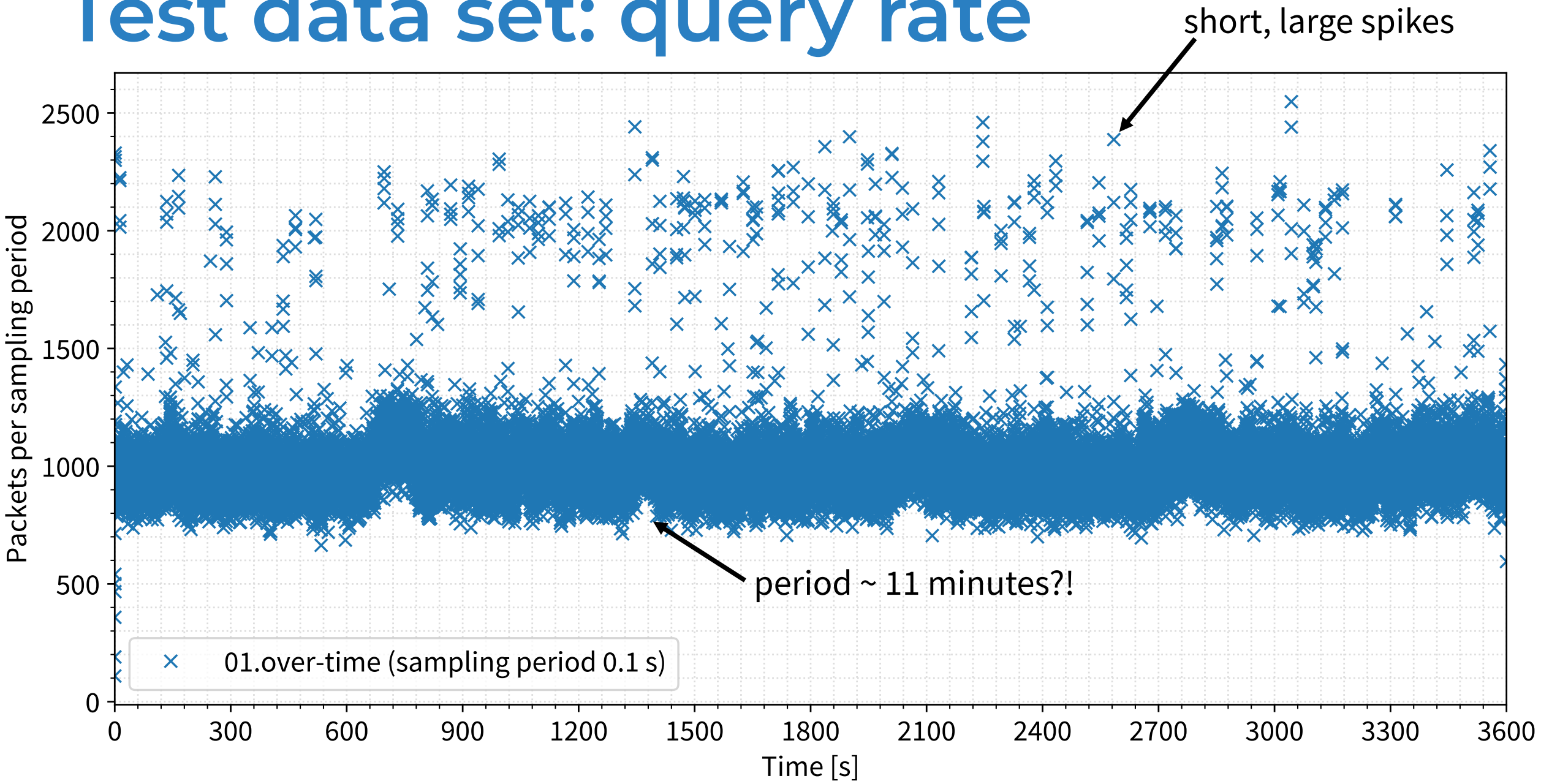


- query stream replay
- measure latency
- repeat 10 times
  - with and without cap

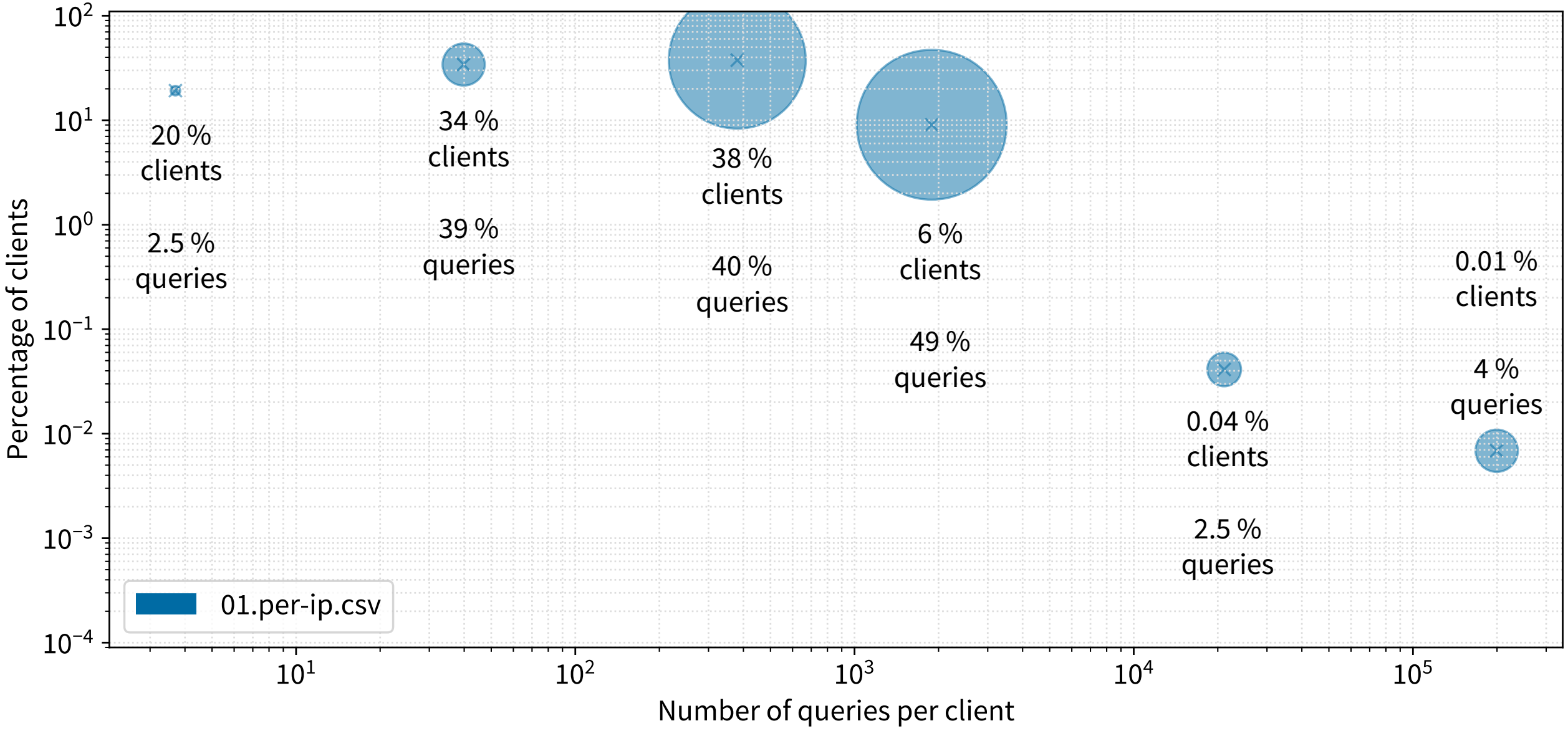
Test data set #1

- Real traffic capture
  - **Anonymized**
  - 1 real telco resolver, 1 hour of traffic
  - Telco in northern Europe, March 2023
  - Mix of landline & mobile clients
- DNSSEC in nordics* – signed domains
  - .NO ~ 60 %
  - .SE ~ 55 %
  - .NU ~ 51 %

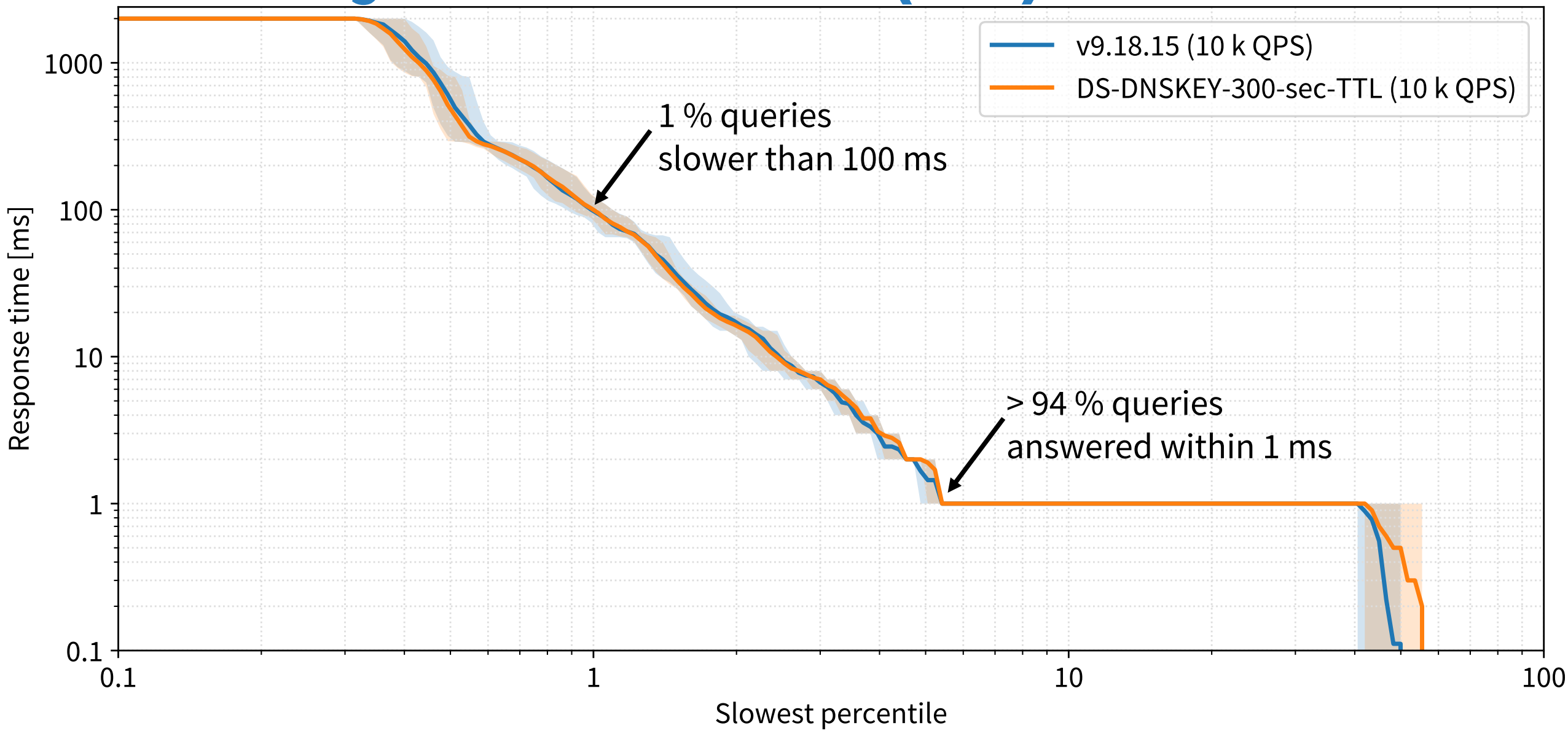
Test data set: query rate



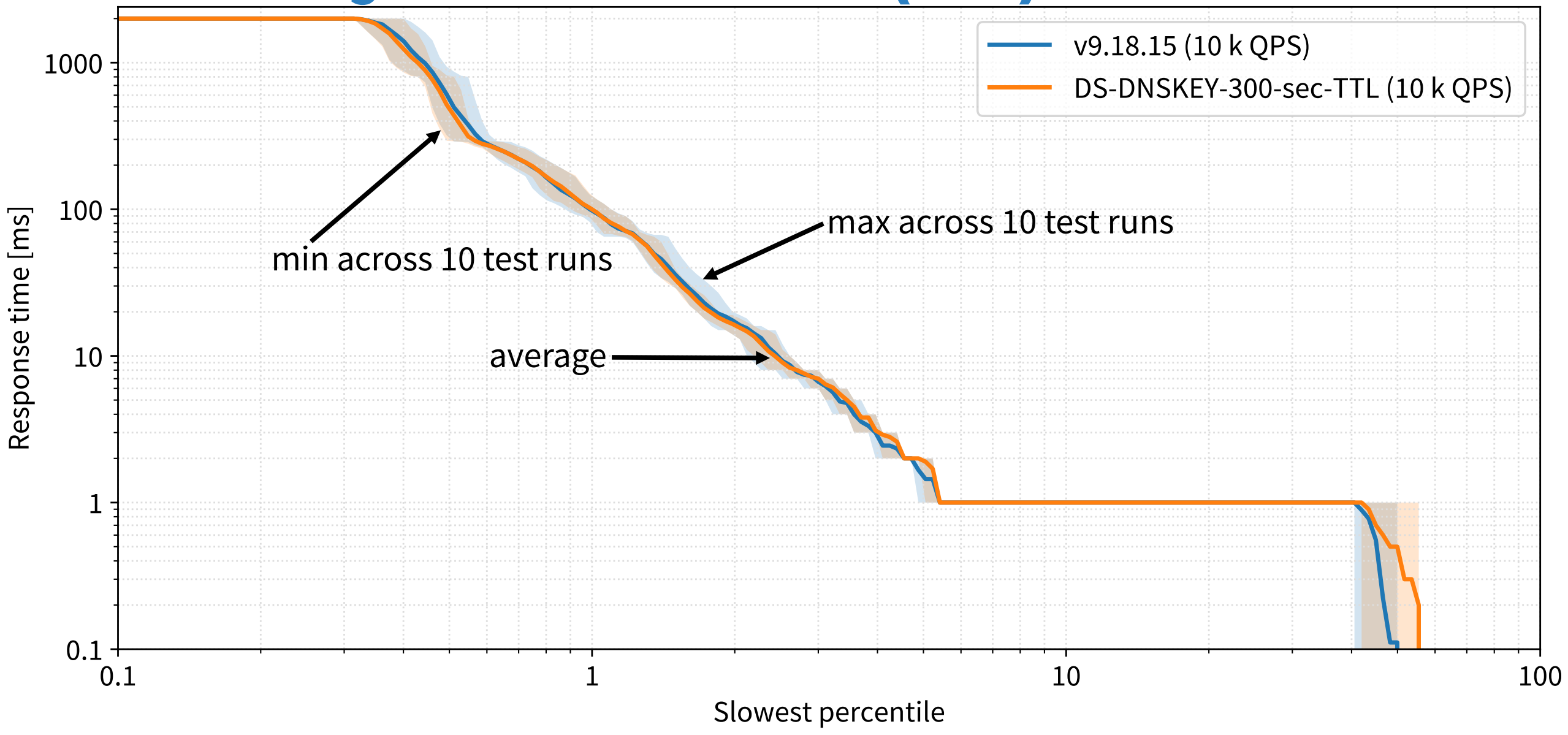
Test data set: query distribution



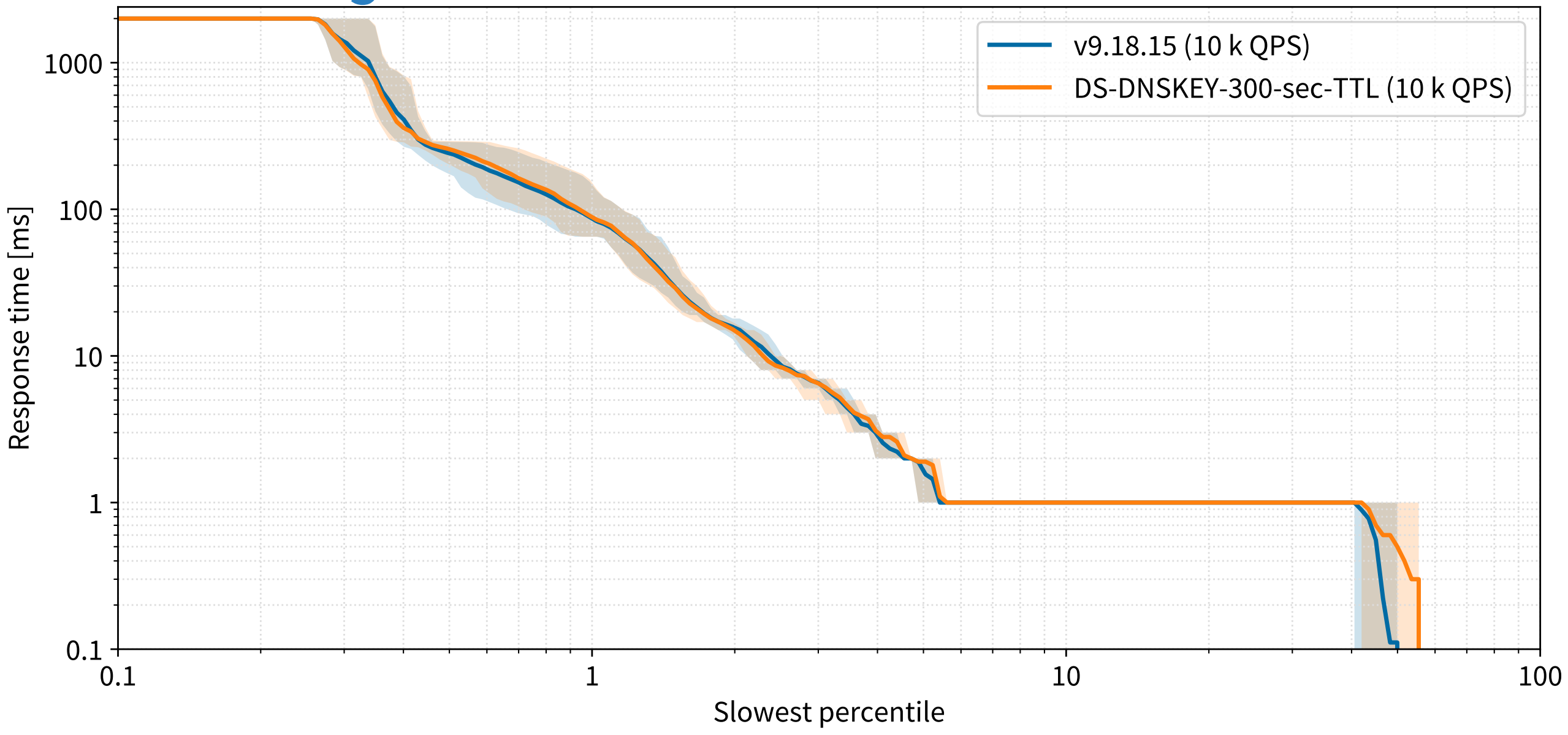
Latency: whole test (1 h) – telco



Latency: whole test (1 h) – telco

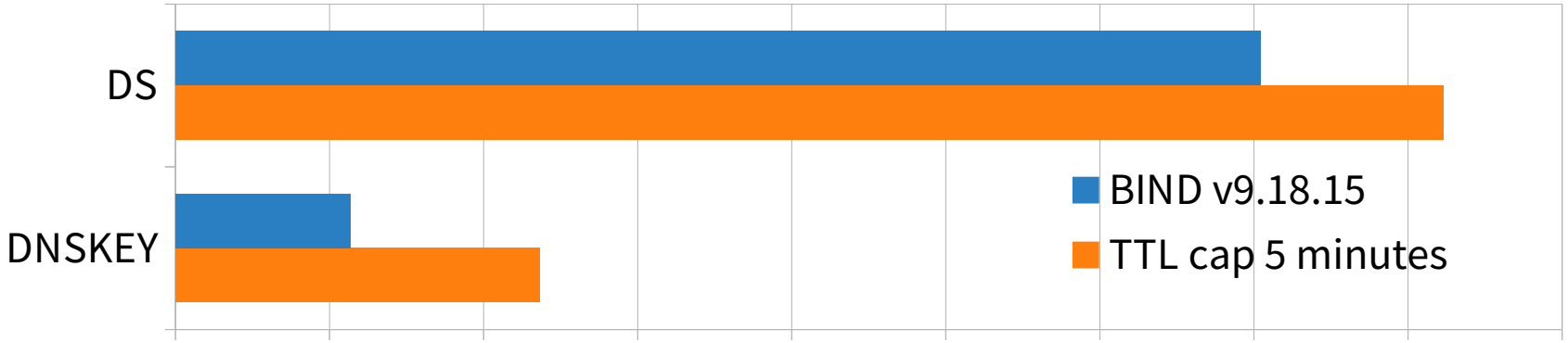


Latency: 05:00 – 10:00 minutes



Queries upstream

Resolver	**DS** queries	**DNSKEY** queries	**All** queries
BIND v9.18.15	70 411	11 372	3 241 806
+ TTL cap 5 minutes	82 311 (+ 17 %)	23 653 (+ 108 %)	3 259 009 (+ 0.5 %)

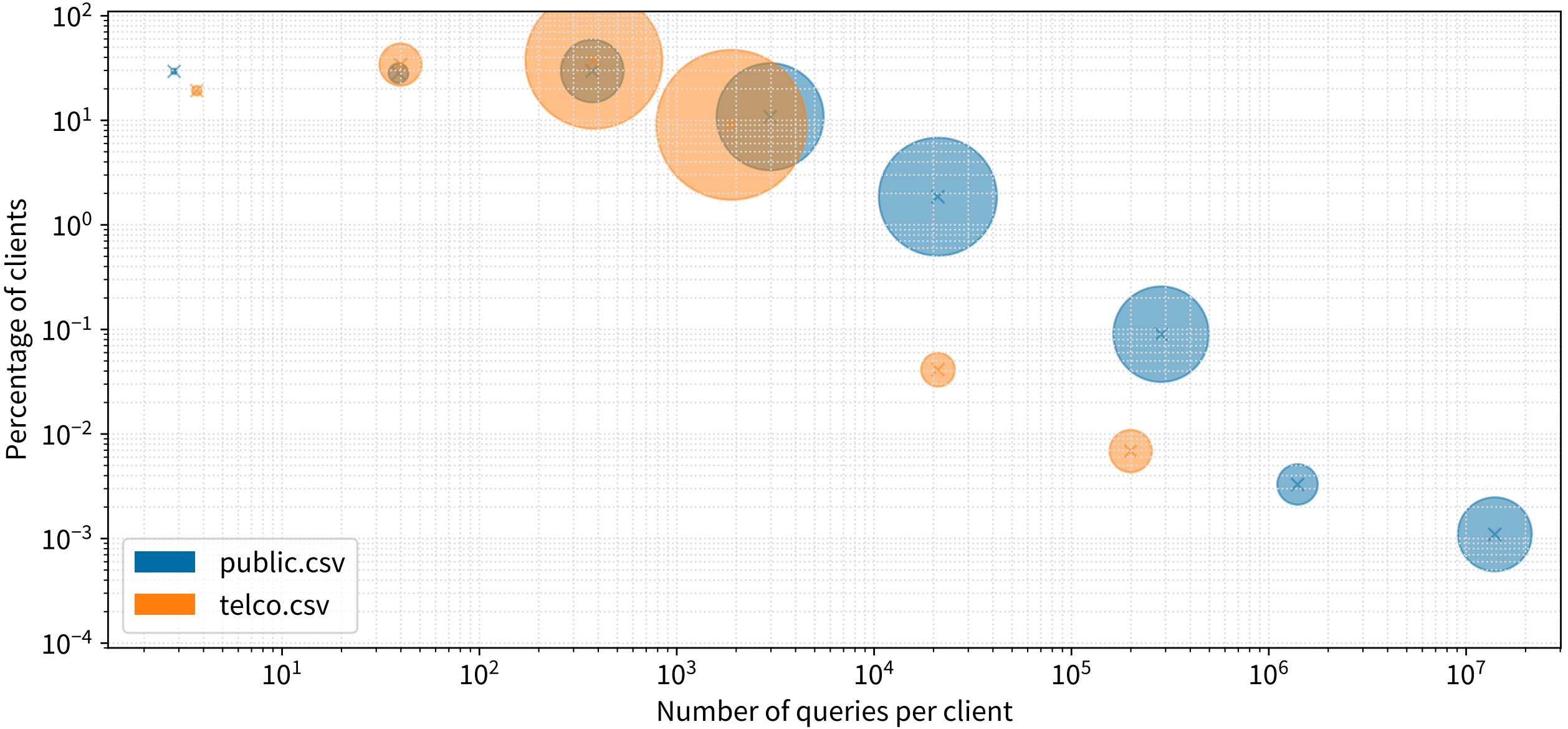


Number of queries sent to upstream servers

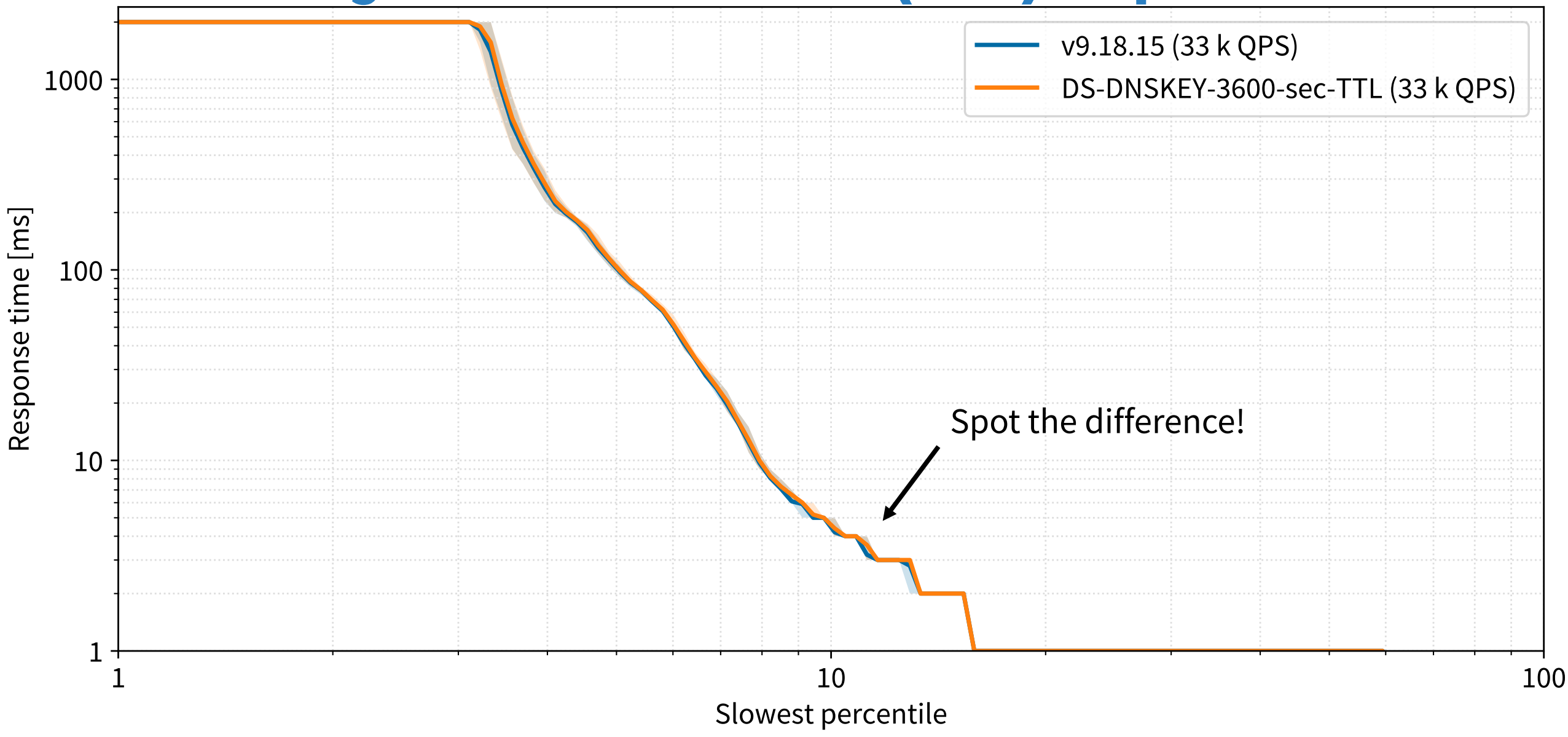
Test data set #2

- Real traffic capture
  - **Anonymized**
  - 1 PoP, Washington D.C. area, 1 hour of traffic
  - Public resolver
  - Composition of clients unknown
- US – signed domains
  - .COM ~ 4 %
  - .NET ~ 5 %
  - .GOV ~ 8 %

Test data set: query distribution



Latency: whole test (1 h) – public



+20 % DS queries – significant?

TLD	Signed	DS TTL	**current** DS volume	**projected** DS volume
CZ	57 %	1 h	8 %	~ + 2 %
GOOG	7 %	1 h	30 %	~ + 6 %
NL	59 %	1 h	9 %	~ + 2 %

+108 % DNSKEY queries – significant?

Operator	Signed	DNSKEY TTL	DNSKEY volume
NS1	0.1 %	1 h	0.3 %
GoDaddy	?	1 h ?	4 %
deSEC	59 %	1 h	8 %

Steady state: Suối tranh

What we provision for: Dray Nur

Lab Experiment Evaluation

- 5 minute DS TTL @ TLD
  - **No** impact on answer latency
  - **Minimal** impact on TLD steady-state
- 5 minute DNSKEY TTL @ apex
  - Surprisingly still no latency impact
- 0.5 % increase in auth query volume

goog. TLD
Real-World Experience

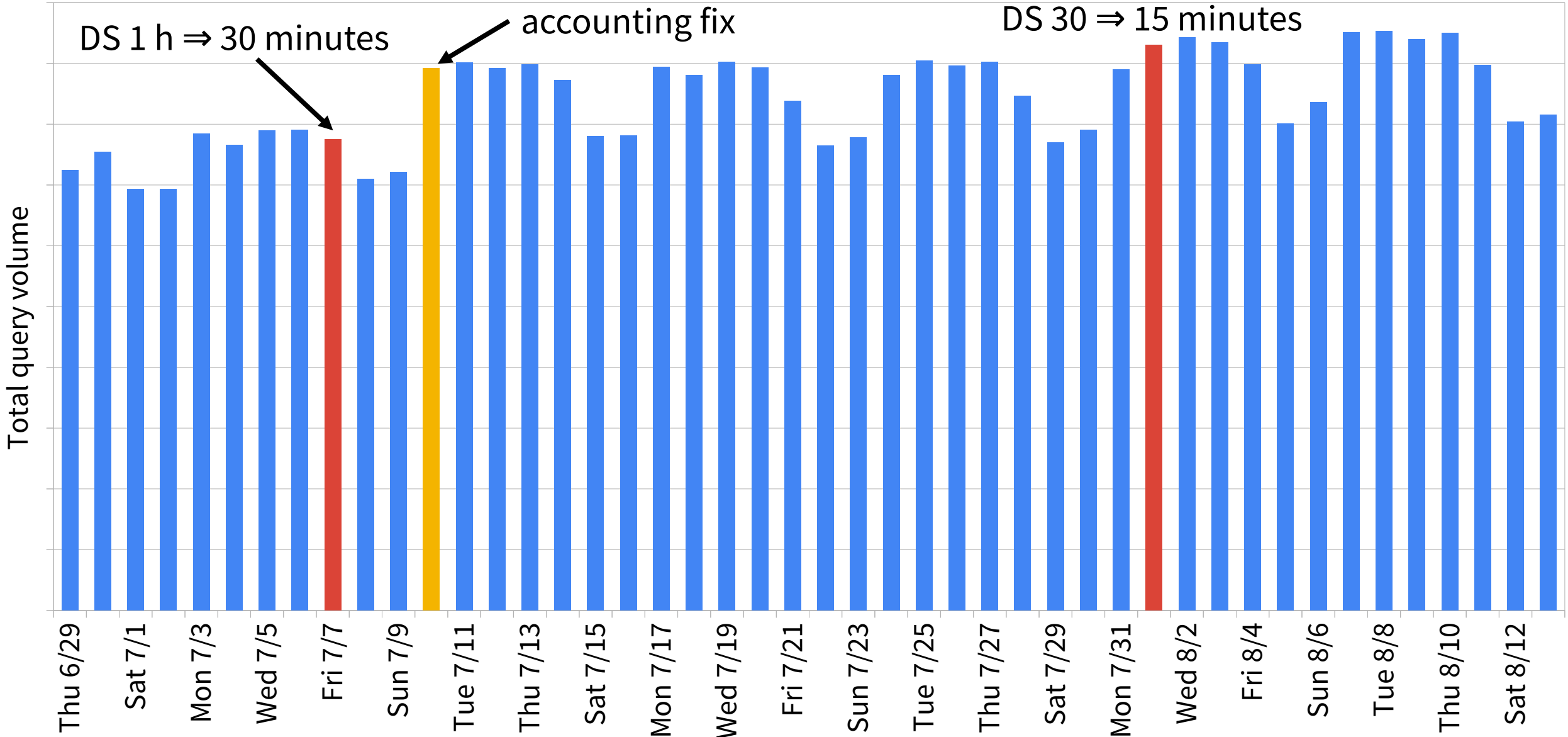
data provided by

Viktor Dukhovni
dviktor@google.com

goog. TTL change

- DS TTL @ TLD
  - 1 h ⇒ 30 minutes ⇒ 15 minutes
  - Already deployed!
- DNSKEY TTL @ 2LD
  - 1h – **unchanged**
  - for domains on Google DNS servers

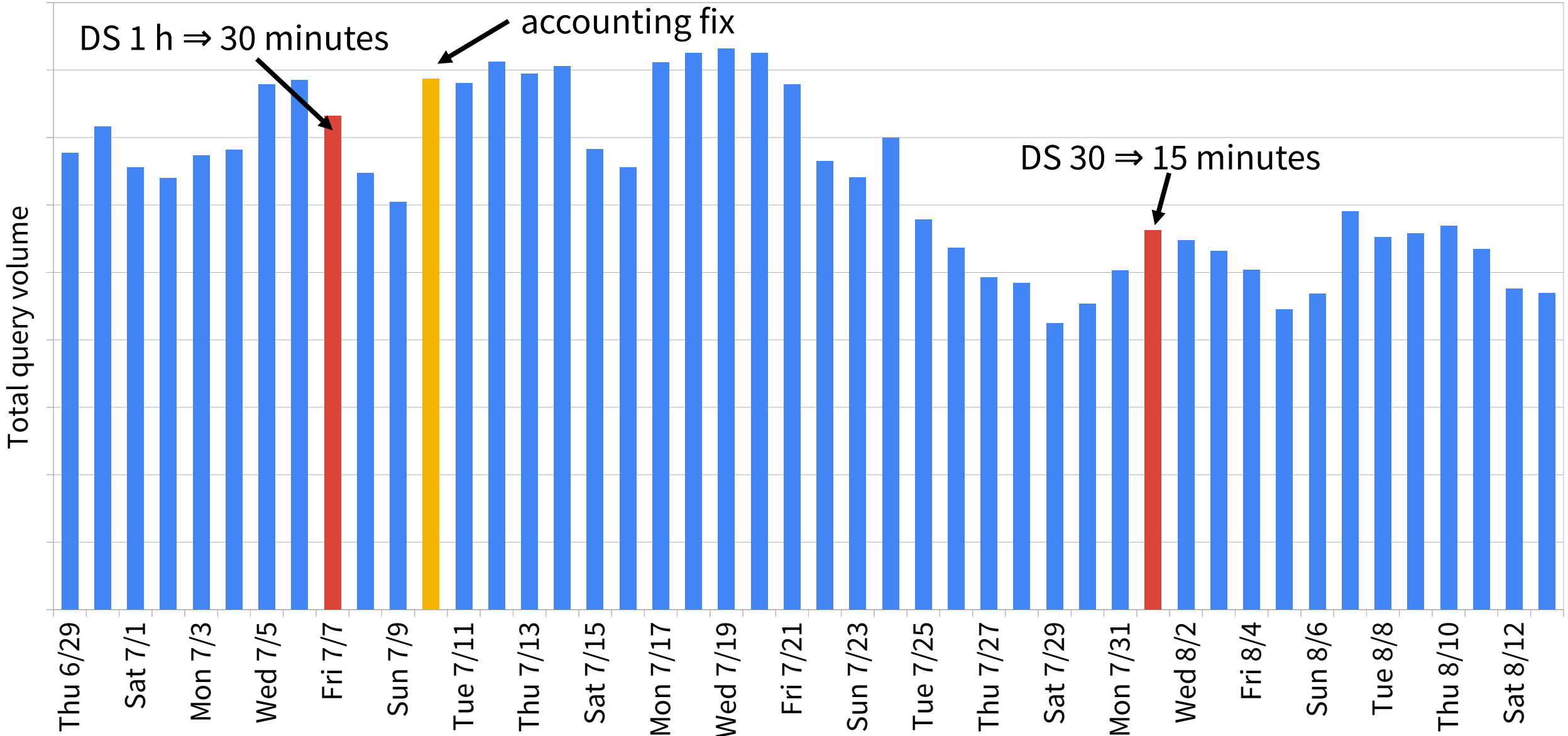
goog. TLD traffic before & after



goog. TLD traffic before & after

DS TTL	DS volume
1 h	29 %
30 m	30 %
15 m	33 %

goog. 2LD traffic before & after



goog. 2LD traffic before & after

DS TTL	DNSKEY TTL	DNSKEY volume
1 h	1 h	3 %
30 m	1 h	3 %
15 m	1 h	4 %

Conclusions – for real

- 15 minute DS TTL @ TLD
  - **No** impact on answer latency
  - **Minimal** impact on TLD steady-state
  - **No** impact on 2LD
- Much faster recovery? Why not?

Thank you!

- Main website: https://www.isc.org
- Software downloads: https://www.isc.org/download or https://downloads.isc.org
- Presentations: https://www.isc.org/presentations
- Main GitLab: https://gitlab.isc.org