

OpenTelemetry Tracing for your DNS

Observability using standards

Peter van Dijk and Otto Moerbeek

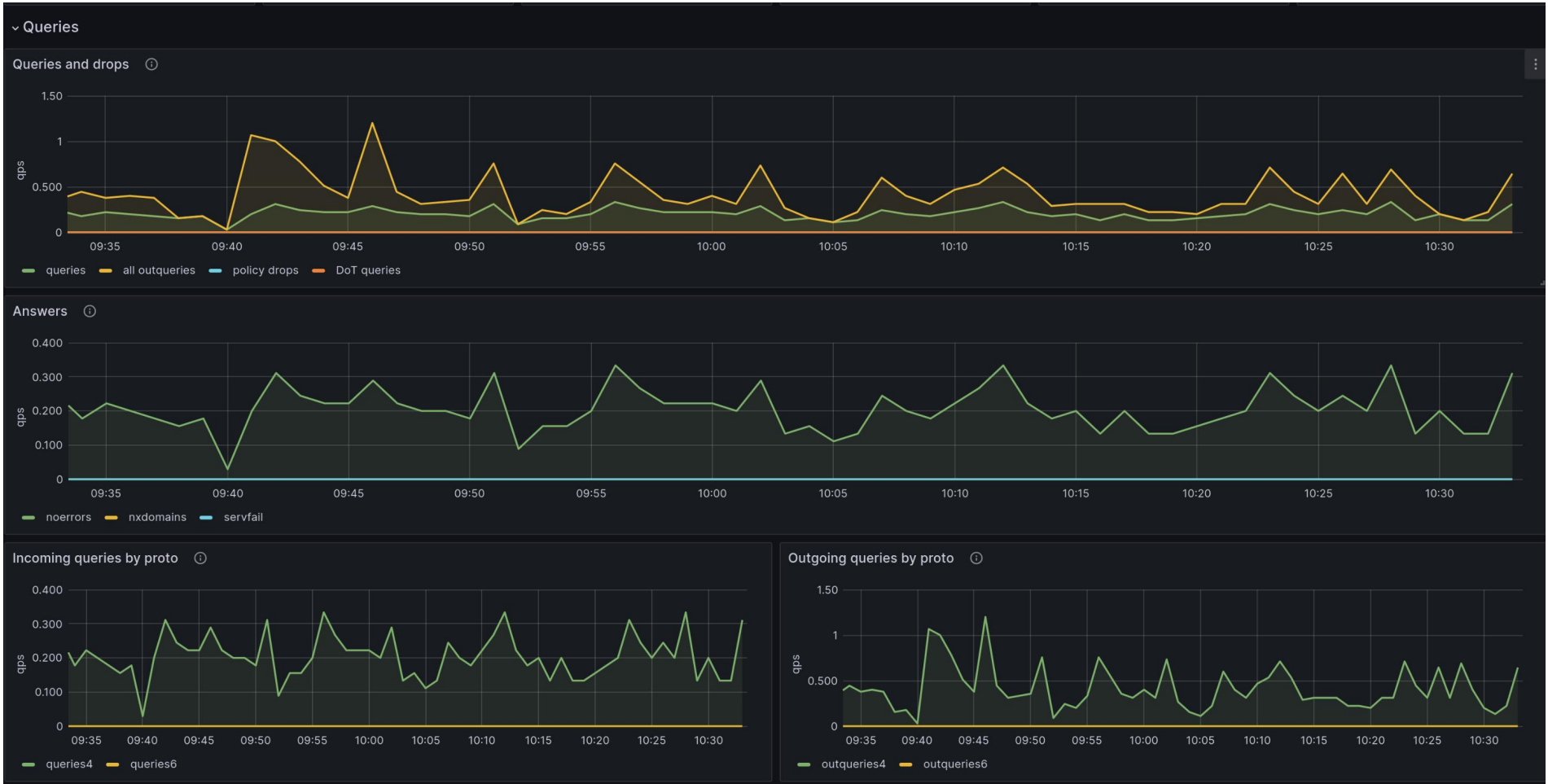
OARC-45 8th of October 2025

POWERDNS 

Stay Open.

OX

Statistics? We have statistics



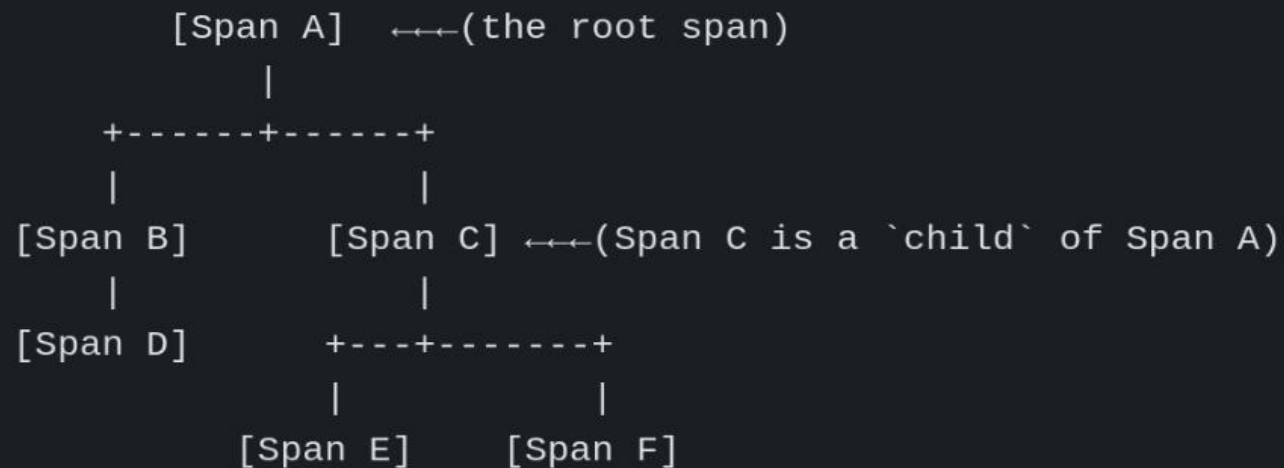
OpenTelemetry Trace Information

Structured using *Spans* and easy to correlate.

From <https://opentelemetry.io/docs/specs/otel/overview/#traces>:

For example, the following is an example **Trace** made up of 6 **Spans**:

Causal relationships between Spans in a single Trace



OpenTelemetry Trace Information

Structured using *Spans* and easy to correlate.

From <https://opentelemetry.io/docs/specs/otel/overview/#traces>:

Sometimes it's easier to visualize **Traces** with a time axis as in the diagram below:

Temporal relationships between Spans in a single Trace

--|-----|-----|-----|-----|-----|-----|-----|-> time

[Span A.....]

[Span B.....]

[Span D.....]

[Span C.....]

[Span E.....]

[Span F..]

Example Span

In JSON representation

```
{
  "trace_id": "69fl8J6/Jk0wG9q6jB0ikg==",
  "span_id": "XZSWxgWu3uM=",
  "parent_span_id": "ZersYXD33JE=",
  "name": "SyncRes",
  "start_time_unix_nano": "1756795915925147000",
  "end_time_unix_nano": "1756795916160608000",
  "attributes": [
    {
      "key": "result",
      "value": {
        "int_value": "0"
      }
    }
  ]
},
```

Which Trace?

Which Span?

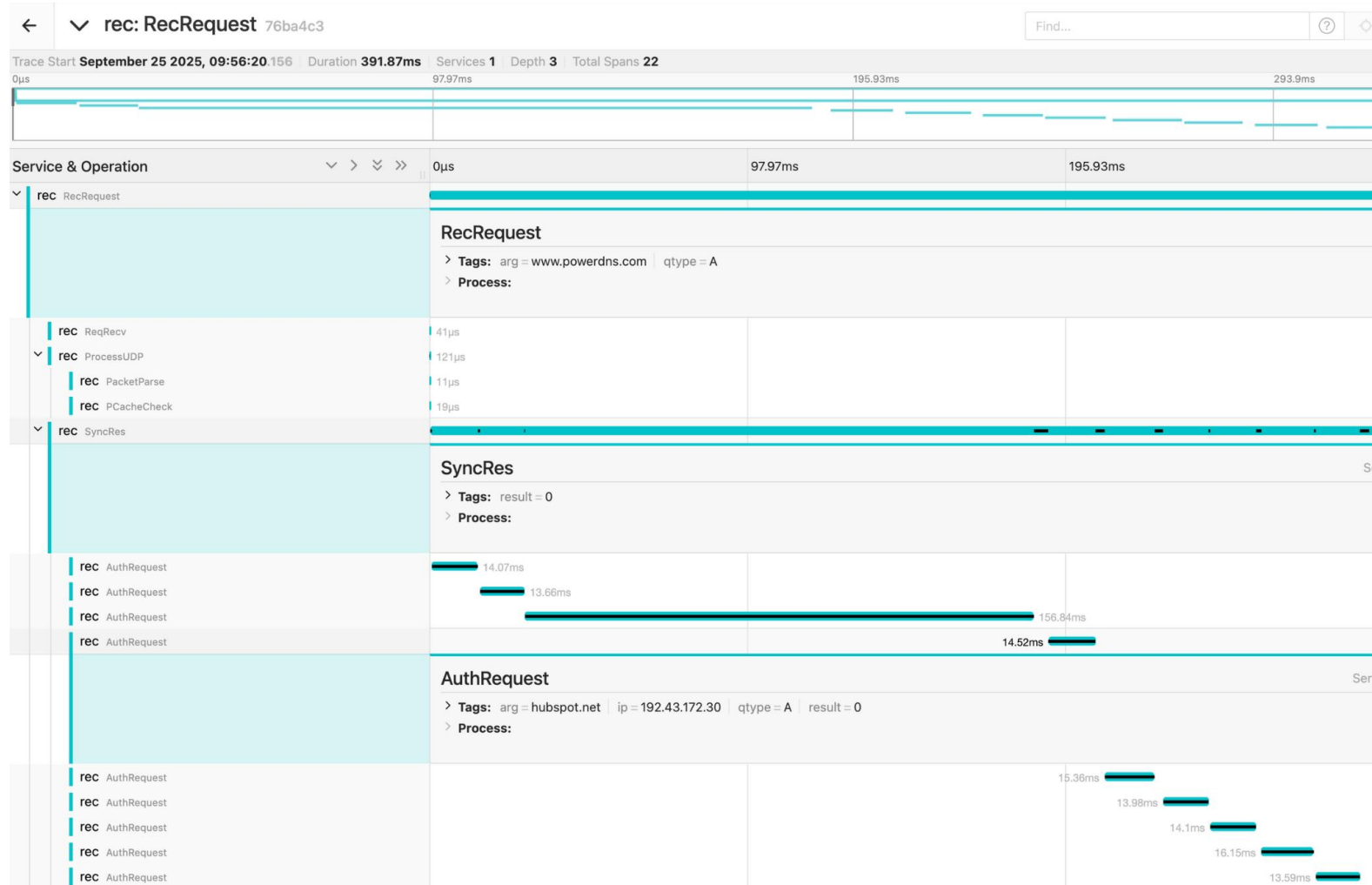
What's my parent?

Standard attributes

App specific attributes

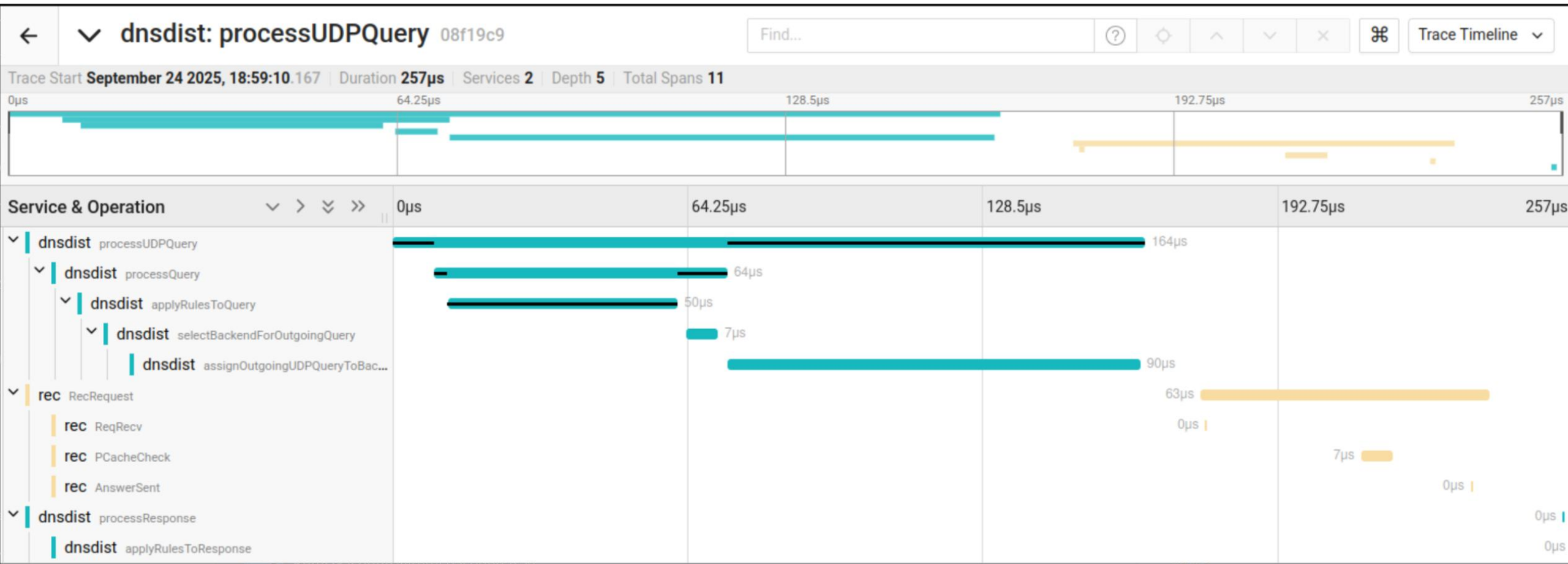
What does it look like: PowerDNS Recursor

Using off the shelf OT Trace collector and viewer: jaeger



dnscat + Recursor

Structured using *Spans* and easy to correlate.



Protobuf logging

<https://github.com/PowerDNS/dnsmessage/blob/master/dnsmessage.proto>

```
187     optional HTTPVersion httpVersion = 24;           // HTTP version used for DNS over HTTP
188     optional uint64 workerId = 25;                   // Thread id
189     optional bool packetCacheHit = 26;               // Was it a packet cache hit?
190     optional uint32 outgoingQueries = 27;            // Number of outgoing queries used to answer the query
191     optional uint32 headerFlags = 28;                // Flags field in wire format, 16 bits used
192     optional uint32 ednsVersion = 29;                // EDNS version and flags in wire format, see https://ww
193 + optional bytes openTelemetryData = 30;            // Protobuf encoded Open Telemetry Data, see https://git
```


DNSTAP?

<https://github.com/dnstap/dnstap.pb/blob/main/dnstap.proto>

```
// Extra data for this payload.  
// This field can be used for adding an arbitrary byte-string annotation to  
// the payload. No encoding or interpretation is applied or enforced.  
optional bytes      extra = 3;
```

Use "extra"? Or add a specific field?

Passing TraceID and SpanID in DNS packets

EDNS0 to the rescue to make correlation very easy

See very early draft: <https://powerdns.github.io/draft-edns-otel-trace-ids/draft-edns-otel-trace-ids.html>

3. Wire Format

```

      0               1
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-----+-----+
0: |          OPTION-CODE (TBD1)          |
+-----+-----+
2: |    OPTION-LENGTH (18 or 26)    |
+-----+-----+
4: |  VERSION (0) |  RESERVED  |
+-----+-----+
6: |    TRACE ID (16 octets)    /
  /                               /
22: |-----+-----+
   | SPAN ID (optional, 8 octets) /
   /                               /
+-----+-----+

```

Alternatively:

- PROXYv2 TLV
- HTTP header (where applicable)

Questions?