DNS Sessions
- where next?

Sara Dickinson      sara@sinodun.com
OARC 27              San Jose, Sep 2017
Overview

• A tour of the evolution of doing DNS over session based protocols
• Recent use cases for DNS Sessions
• Trade-offs encountered when using DNS sessions
• Summarise some of the recent research
A History of DNS Sessions (TCP)
Happy Birthday DNS

- Nov 1987 - RFC1034 and RFC1035 published!
Happy Birthday DNS

- Nov 1987 - RFC1034 and RFC1035 published!

“The DNS assumes that messages will be transmitted as datagrams or in a byte stream carried by a virtual circuit”
RFC1035
(on transport)

“While virtual circuits can be used for any DNS activity, **datagrams are preferred** for queries due to their lower overhead and better performance.”
RFC1035
(on transport)

“While virtual circuits can be used for any DNS activity, datagrams are preferred for queries due to their lower overhead and better performance.”

“Zone refresh activities must use virtual circuits because of the need for reliable transfer.”
RFC 1035
(on transport)

“While virtual circuits can be used for any DNS activity, **datagrams are preferred** for queries due to their lower overhead and better performance.”

“**Zone refresh activities must use virtual circuits** because of the need for reliable transfer.”

“Messages carried by UDP are restricted to **512 bytes** (not counting the IP or UDP headers).”
RFC1035
(on server connections)

• “The server should support **multiple connections**

• The server should assume that the **client will initiate connection closing**…

• If the server needs to close a dormant connection to reclaim resources, it should wait until the connection has been idle for a period on the **order of two minutes**.”
RFC 1035
(on server connections)

• “The server should support **multiple connections**

• The server should assume that the **client will initiate connection closing**…

• If the server needs to close a dormant connection to reclaim resources, it should wait until the connection has been idle for a period on the **order of two minutes**.”
RFC1035
(on server connections)

- “The server should support **multiple connections**
- The server should assume that the **client will initiate connection closing**…
- If the server needs to close a dormant connection to reclaim resources, it should wait until the connection has been idle for a period on the **order of two minutes**.”

Client in control
“Specifically, a DNS resolver or server that is sending a non-zone-transfer query MUST send a UDP query first.”
RFC1123

“Specifically, a DNS resolver or server that is sending a non-zone-transfer query MUST send a UDP query first.”

- “Thus, resolvers and name servers should implement TCP services as a backup to UDP today, with the knowledge that they will require the TCP service in the future.”
DNSSEC
RFCs 4033, 4034, 4035

- Response to security issues in the DNS
- This means larger answers
- Uh, oh… 512 bytes over UDP won’t do…
• Response to security issues in the DNS
• This means larger answers
• Uh, oh… 512 bytes over UDP won’t do…
Extension Mechanisms for DNS (EDNS(0))

RFC 6891

“This document describes backward compatible mechanisms for allowing the protocol to grow.”
Extension Mechanisms for DNS (EDNS(0))

RFC6891

“This document describes backward compatible mechanisms for allowing the protocol to grow.”

- Advertise buffer sizes larger than 512 bytes
- Not as simple a path as hoped….
DNS Service Discovery

- Polling does not scale, need async publish/subscribe
- I-D: DNS Long-lived queries
  - EDNS(0) option over UDP to set up a “lease” (via HS)
  - Server sends “gratuitous DNS response” on changes
- Draft didn’t progress but solution was deployed
TCP, where art thou?

- 1997: RFC2136 DNS Update “requestors who require an accurate response code must use TCP.”
- 2008: Kaminsky attack (UDP is vulnerable to cache poisoning)
  - Source port randomisation, DNSSEC (Larger answers….)
- 2010: Response Rate Limiting (RRL) proposed.
  - TCP can be a fallback when under attack
DNS can always fallback to TCP, right?

I can’t answer on TCP

You crazy - the middle-boxes will probably kill ya!
Not so much…

- Implementations: No support at all or typically…
- Client does one-shot TCP => Poor performance
- Server conns (5-20) weak => TCP was DDoS’able
- Operations: Middle-boxes block TCP on port 53
  - Some operators disable/block it
  - Some use only for zone transfers
“A resolver SHOULD send a UDP query first, but MAY elect to send a TCP query instead” (TC=1, already connected)
DNS Transport over TCP - Implementation Requirements

RFC 5966

“A resolver SHOULD send a UDP query first, but MAY elect to send a TCP query instead” (TC=1, already connected)

- “Support for TCP is henceforth a REQUIRED part of a full DNS protocol implementation.”
- Described ‘implicit’ persistent sessions but no specifics, improved server advice, clarified response re-ordering
TCP Performance & Persistence

- NSD4 TCP Performance figures, PowerDNS blog
- One shot tools => 1/10th query performance of UDP
- First proposal for EDNS(0) Keepalive (signal capability, specified idle time, server can request connection close)
- DNSSEC, Reflection attacks
Recent Use Cases

UDP WAS OUR ONLY HOPE

NO, THERE IS ANOTHER....
DNS Privacy
(stub to recursive)
DNS Privacy
(stub to recursive)

RFC7258:
Pervasive Monitoring is an attack

Snowdon Revelations


Pervasive Monitoring is an attack
DNS Privacy
(stub to recursive)

RFC 7258: Pervasive Monitoring is an attack

DPRIE WG: We are going to need sessions....

Snowdon Revelations


Pervasive Monitoring is an attack
DNS Privacy
(stub to recursive)

RFC7258: Pervasive Monitoring is an attack

DPRIEV WG: We are going to need sessions….

I-D: EDNS(0) Keepalive

Snowdon Revelations

DNS Privacy
(stub to recursive)

RFC7258: Pervasive Monitoring is an attack

DPRIVE WG: We are going to need sessions....

I-D: EDNS(0) Keepalive

Snowdon Revelations

RFC7858: DNS-over-TLS
DNS Service Discovery

- 2013: IETF created a [DNS-SD working group](#)
- 2015: I-D: [DNS Push Notifications](#) (evolution of LLQ)
  - Uses TCP/TLS
  - Persistent connections, EDNS(0) Keepalive
  - 2 new OPCODEs for Sub/UnSub
Increasing DDoS Attacks on the DNS

- 2013: Spamhouse
- 2016: Dyn - Primary target is UDP but attacked TCP too.
  - Operators need a range of defence mechanisms….
  - Persistent TCP sessions could be part of that
KSK rollover

- DNSSEC in action
- DNSKEY responses from the root will peak over 1280 bytes during the rollover (including right now)
- ICANN: “Make sure your servers can query over TCP (especially over IPv6)”
Latest Specs for DNS Sessions

Nobody puts TCP in a corner
Domain Name System (DNS) Cookies

RFC7873

• “This document describes DNS Cookies, a lightweight DNS transaction security mechanism specified as an OPT option.”

• “The protection provided by DNS Cookies is similar to that provided by using TCP for DNS transactions.”

• Pseudo-session?
“Stub resolvers and recursive resolvers MAY elect to send either TCP or UDP queries depending on local operational reasons.”
DNS Transport over TCP - Implementation Requirements

RFC7766 (RFC5966-bis)

“Stub resolvers and recursive resolvers MAY elect to send either TCP or UDP queries depending on local operational reasons.”

• Optimised performance of connections (pipelining, TFO)

• “In essence, TCP ought to be considered a valid alternative transport to UDP, not purely a retry option.”
I-D: DNS Transport over TCP - Operational Requirements

- Response to ongoing confusion amongst students, operators about use of TCP (John Kirstoff RFC review)
- Companion to RFC7766 - look at operational aspects
- Updates server resource limitation, TCP filtering (DNS Wedgie)
Future of DNS Sessions?
EDNS(0) Keepalive doesn’t cut it

- 2016: EDNS(0) Keepalive just not sufficient - Why not?
  - EDNS(0) is defined as *per-message*
  - Client signalling it tied to real (or empty) messages
  - Server can only use EDNS(0) if query contained EDNS(0)
  - Server cannot initiate communication
Hello ‘Session Signalling”

- **I-D: Session Signalling**: Generalised model of ‘signalling’ under one new **OPCODE**
- TLV format (not EDNS(0))
- Clients and servers exchange SS messages to create a ‘session’
  - Keepalive traffic (inactivity and keepalive timeouts)
  - Server initiated messages
  - (Push subscriptions occur within the session using TLVs)
Goodbye “SS”, Hello “DNS Stateful Operations”

• Service Discovery - other use cases became apparent (not pure signalling)

• Push: Servers “push” updated data directly to client…..

• mDNS SD Relay

• Other use cases? (server capability, alternate servers, etc.)

• DSO Draft: Renamed, updated, need review to move forward
What is meant by a session for DSO?

- **CONNECTION:**
  - “a bidirectional byte stream of reliable, in-order messages”

- **SESSION:**
  - The connection is persistent and relatively long-lived
  - Either end of the connection may initiate messages to the other.
What is meant by a session for DSO?

- **CONNECTION:**
  - “a bidirectional byte stream of reliable, in-order messages”

- **SESSION:**
  - The connection is persistent and relatively long-lived
  - Either end of the connection may initiate messages to the other.

<table>
<thead>
<tr>
<th>Protocol</th>
<th>SESSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDP (even with cookies)</td>
<td>N</td>
</tr>
<tr>
<td>TCP</td>
<td>Y</td>
</tr>
<tr>
<td>TLS</td>
<td>Y</td>
</tr>
<tr>
<td>QUIC</td>
<td>N (but…)</td>
</tr>
</tbody>
</table>
Session overheads

- **Server** state is proportional to number of connections
  - Off-load overhead to a proxy (understand DNS?)

- **Clients** have more failure modes, server selection can be more complex

- **Connection re-use** depends on traffic (bursty)
  - Client RTT is amortised for N queries as \( (1+N)/N \)
    => 10 queries, 1.1 RTT (TCP Fast Open)
TCP Investigations

- 2015: Academic research: Heideman, IEEE
- 2016: Recursive perspective: Damas, RIPE 71
- 2017: Authoritative perspective (DITL): Včelák, OARC26
  - Google resolvers did 3 queries per TCP session
- ICANN in prep for KSK roll (TCP can be better for retries)
nic.at Research

- **Alexander Mayrhofer @ JSCA 17**: TLS/TCP Cost Simulation

- Authoritative data, very sensitive to idle timeout
nic.at Research

• Alexander Mayrhofer @ JSCA 17: TLS/TCP Cost Simulation

• Authoritative data, very sensitive to idle timeout

Rough UDP vs TLS cost $\times 8$
Next steps

• Work to do on DNS Stateful Operations

• More analysis of traffic patterns (rec and auth)

• More rigorous benchmarking
  • DNSPERF TCP patch, but need custom tool
  • Open Tech Fund funding for DNS-over-TLS benchmarking (DNS Privacy project)
Thank you!

Any Questions?