

DNS Transport over TCP - Implementation Requirements

J. Dickinson, Sinodun Internet Technologies

R. Bellis, Nominet

A. Mankin and D. Wessels, Verisign Labs

DNS Transport over TCP

- This is a -bis of RFC5966
- Aim of draft is to put TCP on the same footing as UDP for use as a DNS transport
- In support of
 - Privacy efforts
 - Preventing amplification attacks
 - Packet size limitations

DNS Transport over TCP

- Major changes in -bis include:
 - DNS implementations are recommended not only to support TCP but to support it on an equal footing with UDP
 - DNS implementations are recommended to support reuse of TCP connections
 - DNS implementations are recommended to support pipelining and out of order processing of the query stream
 - A non-normative discussion of use of TCP Fast Open

Connection Handling

- One perceived disadvantage to DNS over TCP is the added connection setup latency, generally equal to one RTT.
 - Both clients and servers SHOULD support connection reuse by sending multiple queries and responses over a single TCP connection.
- DNS currently has no connection signalling mechanism. Clients and servers may close a connection at any time. Clients MUST be prepared to retry failed queries on broken connections.

Connection Handling

- To mitigate the risk of unintentional server overload, it is RECOMMENDED that for any given client - server interaction there SHOULD be no more than one connection for
 - regular queries [One for each client application]
 - one for zone transfers
 - one for each protocol that is being used on top of TCP, for example, if the resolver was using TLS.

Query Pipelining

- In order to achieve performance on par with UDP, it is RECOMMENDED that DNS clients pipeline their queries.
 - Do not wait for an outstanding reply before sending the next query.
- DNS servers SHOULD expect to receive pipelined queries. The server should process TCP queries in parallel, just as it would for UDP.

Query Pipelining

- Authoritative servers and recursive resolvers are **RECOMMENDED** to support the sending of responses in parallel and/or out-of-order, regardless of the transport protocol in use.
- Stub and recursive resolvers **MUST** be able to process responses that arrive in a different order to that in which the requests were sent, regardless of the transport protocol in use.
- Recursive resolvers **SHOULD** process TCP queries in parallel and return individual responses as soon as they are available, possibly out-of-order.

TCP Fast Open

- This section is non-normative.
- TCP fastopen [I-D.ietf-tcpm-fastopen] (TFO) allows data to be carried in the SYN packet.
 - It saves up to one RTT compared to standard TCP.
- Currently Linux only. 3.16.0 added IPv6 support.

TCP Fast Open

- TFO Code changes
 - On the client, the call to `connect()` is replaced with a TFO aware version of `sendmsg()` or `sendto()`.
 - On server, set a socket option between the `bind()` and `listen()` calls.

TCP Fast Open

- TFO kernel config
 - change the kernel parameter `net.ipv4.tcp_fastopen` (A bitmap)
 - 1 = client
 - 2 = server

TCP Fast Open - query

The image shows a Wireshark capture of a network packet. The packet list pane shows a sequence of events: a DNS query, a TCP SYN+ACK packet (Fast Open), a DNS query response, another DNS query, and a TCP ACK packet. The selected packet is a TCP segment with the following details:

- [Stream index: 0]
- [TCP Segment Len: 35]
- Sequence number: 0 (relative sequence number)
- [Next sequence number: 35 (relative sequence number)]
- Acknowledgment number: 0
- Header Length: 52 bytes
- ... 0000 0000 0010 = Flags: 0x002 (SYN)
- Window size value: 29200
- [Calculated window size: 29200]
- Checksum: 0xa7d4 [validation disabled]
- Urgent pointer: 0
- Options: (32 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale, Experimental
 - Maximum segment size: 1460 bytes
 - TCP SACK Permitted Option: True
 - Timestamps: TSval 274228253, TSecr 0
 - No-Operation (NOP)
 - Window scale: 7 (multiply by 128)
 - TCP Option - Experimental: fe0cf9892bfff04fd01dd53c1
- [SEQ/ACK analysis]
- Domain Name System (query)
 - [Response In: 4]

The packet bytes pane shows the raw data of the packet, including the IP header, TCP header, and DNS query data.

TCP Options (tcp.options), ... Packets: 16 · Displayed: 16 (100.0%) · Load time: 0:00.000 Profile: Default

TCP Pipelining

Multiple queries in one packet

The image shows a Wireshark capture of a network packet. The packet list pane shows a sequence of packets: a TCP ACK (No. 7), a DNS response (No. 8), a DNS query (No. 9), a TCP ACK (No. 10), a TCP ACK (No. 11), and a DNS response (No. 12). The selected packet (No. 9) is expanded to show its details: Ethernet II, Internet Protocol Version 4, Transmission Control Protocol, and Domain Name System (query). The query is for www.sinodun.com, type A, class IN. The packet bytes pane shows the raw data of the packet, including the IP header and the DNS query.

No.	Time	Source	Destination	Protocol	Length	Info
7	0.000501000	192.168.11.1	192.168.11.3	TCP	66	56822->43211 [ACK] Seq=71 Ack=88 Win=29312 Len=0 TSval=274228253 TSecr=25...
8	0.000527000	192.168.11.3	192.168.11.1	DNS	153	Standard query response 0x0001 A 192.168.11.5
9	0.000624000	192.168.11.1	192.168.11.3	DNS	171	Standard query 0x0004 A www.sinodun.com
10	0.037256000	192.168.11.3	192.168.11.1	TCP	66	43211->56822 [ACK] Seq=175 Ack=176 Win=29056 Len=0 TSval=252757646 TSecr=...
11	0.038073000	192.168.11.1	192.168.11.3	TCP	66	56822->43211 [ACK] Seq=176 Ack=175 Win=29312 Len=0 TSval=274228263 TSecr=...
12	0.038099000	192.168.11.3	192.168.11.1	DNS	327	Standard query response 0x0004 A 192.168.11.5

Frame 9: 171 bytes on wire (1368 bits), 171 bytes captured (1368 bits) on interface 0

- Ethernet II, Src: AsustekC_d5:46:ab (10:bf:48:d5:46:ab), Dst: IntelCor_2a:6b:c7 (68:05:ca:2a:6b:c7)
- Internet Protocol Version 4, Src: 192.168.11.1 (192.168.11.1), Dst: 192.168.11.3 (192.168.11.3)
- Transmission Control Protocol, Src Port: 56822 (56822), Dst Port: 43211 (43211), Seq: 71, Ack: 88, Len: 105
- Domain Name System (query)
- Domain Name System (query)
- Domain Name System (query)
 - [Response In: 12]
 - Length: 33
 - Transaction ID: 0x0004
 - Flags: 0x0120 Standard query
 - Questions: 1
 - Answer RRs: 0
 - Authority RRs: 0
 - Additional RRs: 0
 - Queries
 - www.sinodun.com: type A, class IN

0000 68 05 ca 2a 6b c7 10 bf 48 d5 46 ab 08 00 45 00 h..*k... H.F...E.
0010 00 9d c1 28 40 00 40 06 e1 dd c0 a8 0b 01 c0 a8 ... (@. @.
0020 0b 03 dd f6 a8 cb 01 3f a5 80 f7 81 a9 97 80 18?
0030 00 e5 7f b5 00 00 01 01 08 0a 10 58 64 1d 0f 10Xd...
0040 c6 84 00 21 00 02 01 20 00 01 00 00 00 00 00 00 !.....

Frame (frame), 171 bytes | Packets: 16 · Displayed: 16 (100.0%) · Load time: 0:00.000 | Profile: Default

DNS Transport over TCP

- We are seeking adoption of this draft...