DNS Zone Transfer-over-TLS (XoT)

draft-ietf-dprive-xfr-over-tls

Sara Dickinson
Willem Toorop
Shivan Sahib
Pallavi Aras
Allison Mankin

XoT - Background

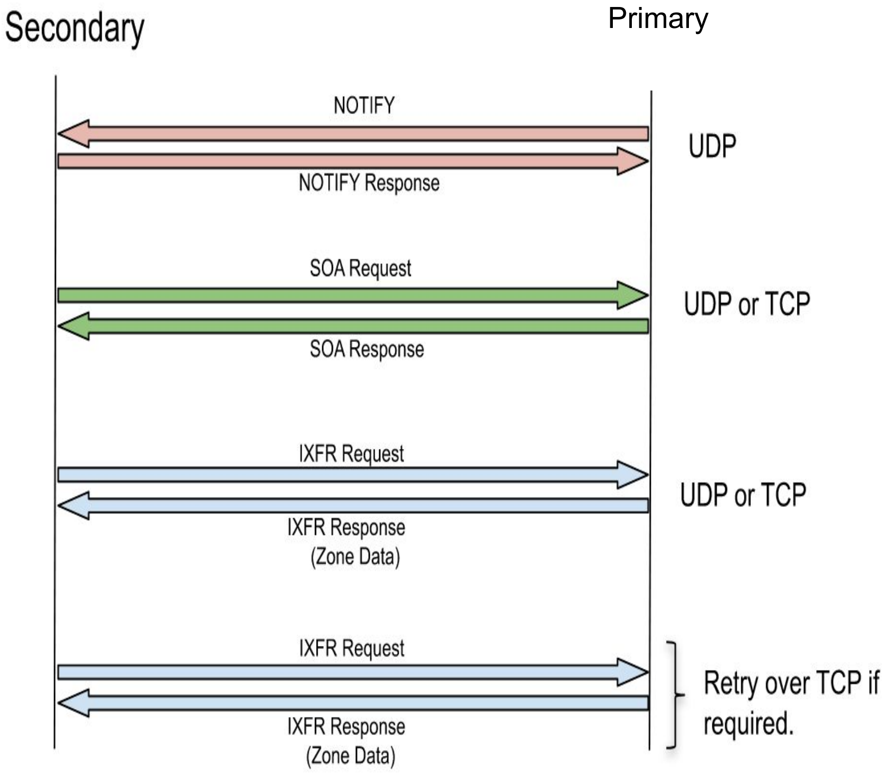
What is XoT?

- Encryption of DNS zone transfer (AXFR & IXFR) using TLS as a transport

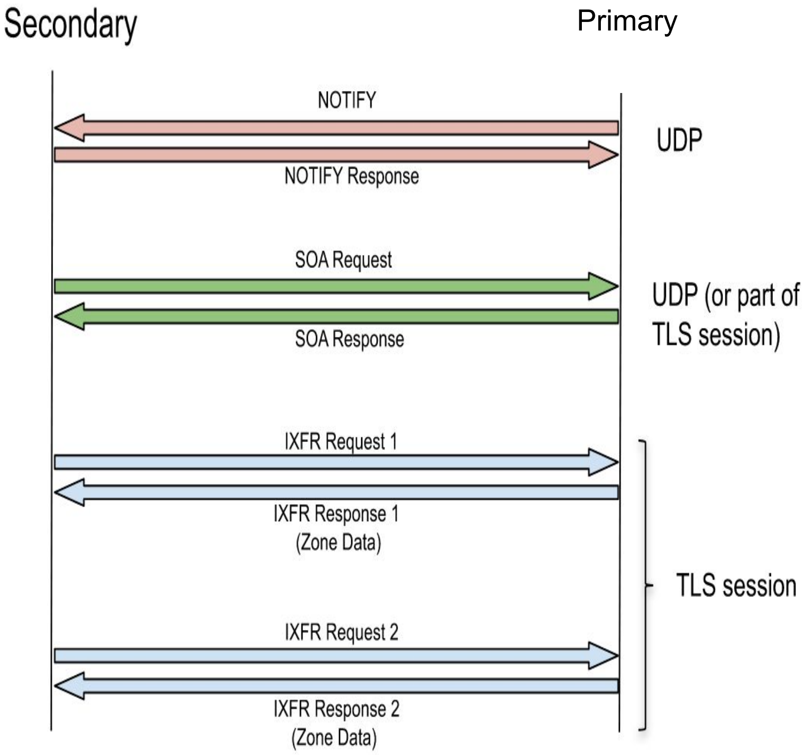
Use cases

- **Confidentiality**: Encrypting zone transfers will defeat zone content leakage that can occur via passive surveillance
- **Authentication**: Use of single or mutual TLS authentication can complement TSIG/ACLs
- **Performance**: Current usage of TCP for XFR is suboptimal in most cases

IXFR : Existing mechanisms vs IXoT

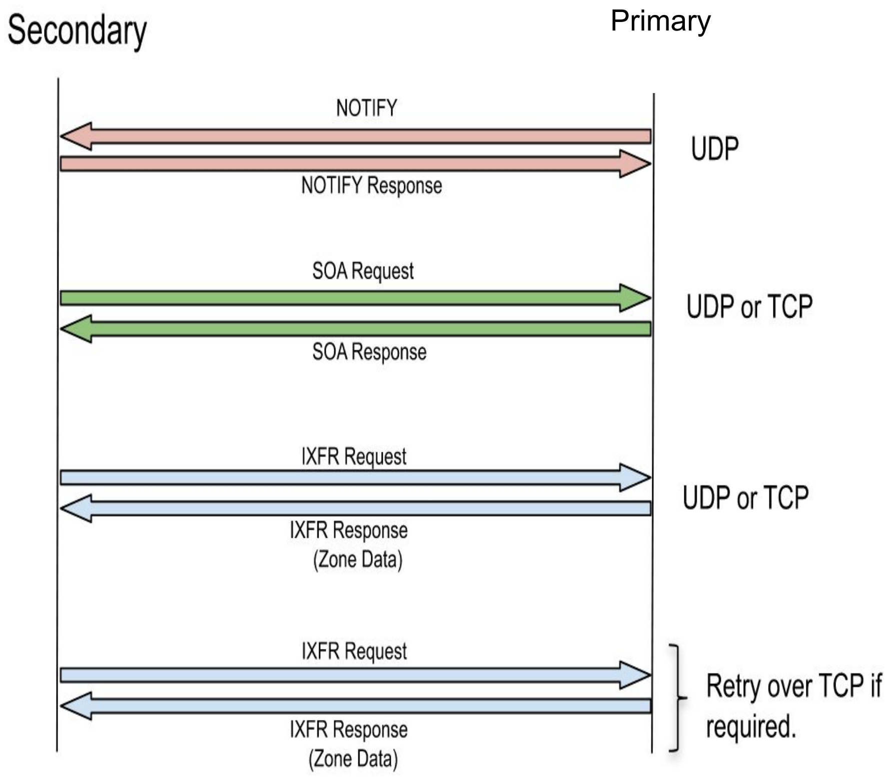


Existing



XOT-Based IXFR

IXFR : Existing mechanisms vs IXoT



Secondary

Primary

NOTIFY

NOTIFY Response

UDP

SOA Request

SOA Response

UDP or TCP

IXFR Request

IXFR Response
(Zone Data)

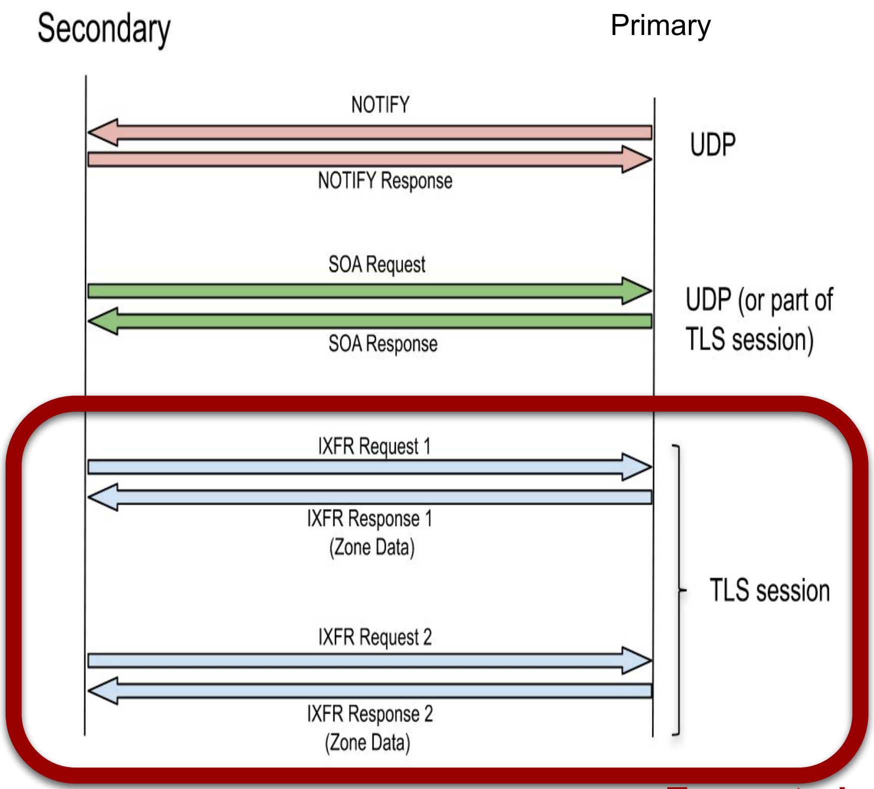
UDP or TCP

IXFR Request

IXFR Response
(Zone Data)

Retry over TCP if required.

Existing



Secondary

Primary

NOTIFY

NOTIFY Response

UDP

SOA Request

SOA Response

UDP (or part of TLS session)

IXFR Request 1

IXFR Response 1
(Zone Data)

IXFR Request 2

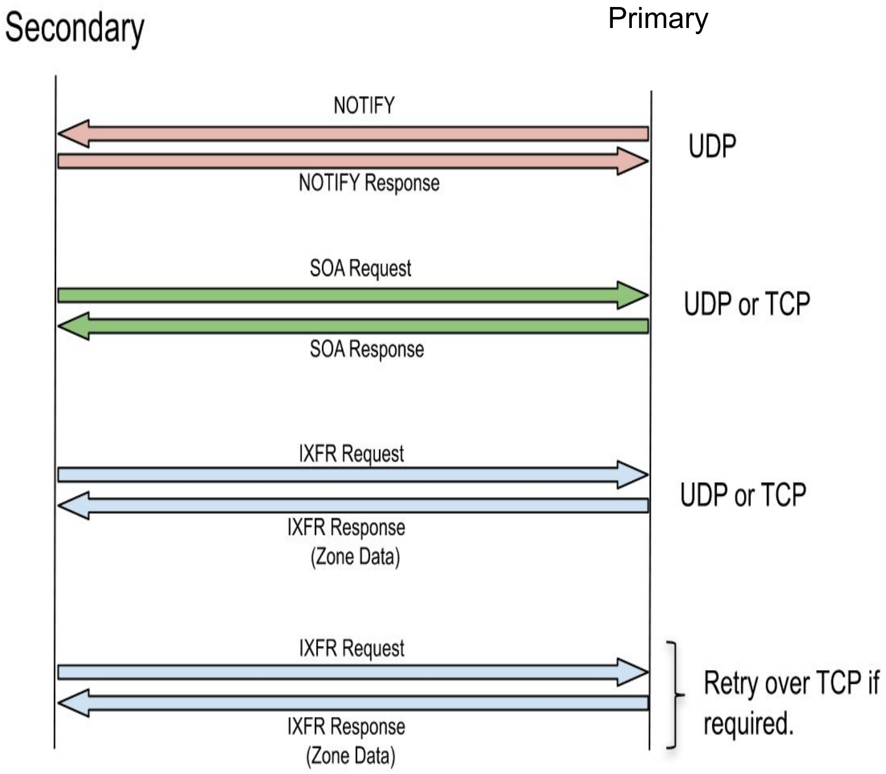
IXFR Response 2
(Zone Data)

TLS session

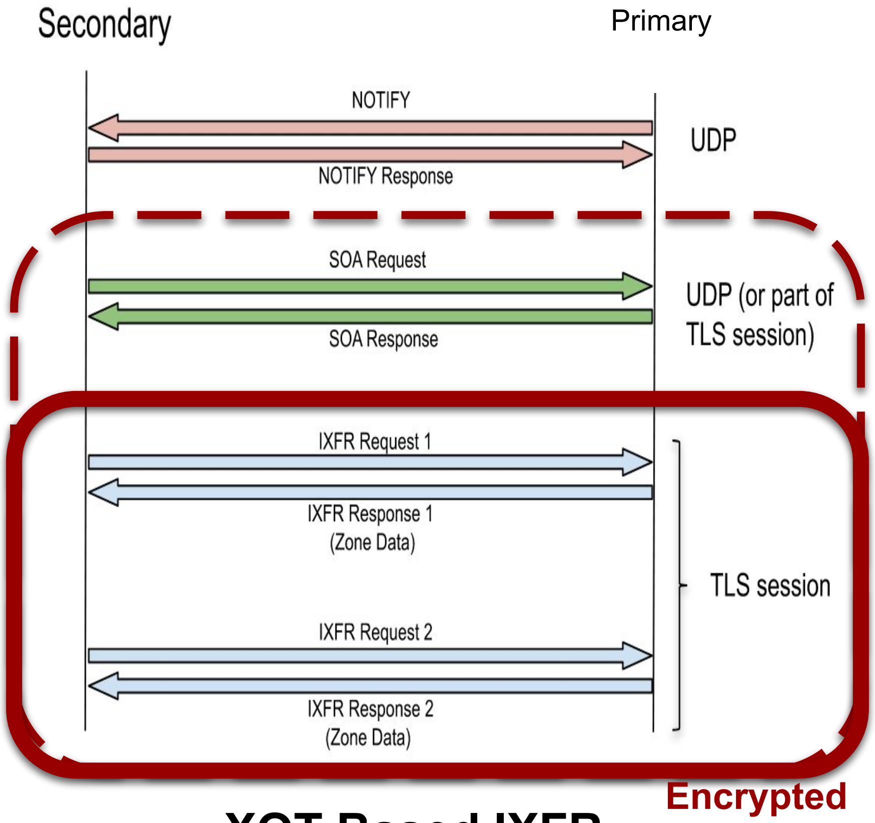
Encrypted

XOT-Based IXFR

IXFR : Existing mechanisms vs IXoT



Existing



XOT-Based IXFR

Current status

- **Draft adopted** by WG in Nov 2019

- -02 was presented at IETF 108
    - **Got a lot of feedback**, particularly on the proposed use of ALPN (not supported)

- -03 version (Oct 2020)
    - Incorporates that feedback
        - Provide details in next few slides
    - Addressed a lot of open questions
    - *Looking for comments today (or on the list) on how close we are to WGLC*

-03 updates (Oct 2020)

- **Terminology**: XFR-over-TCP, XoT, IXoT and AXoT
- **Main elements of draft structure:**
  - Use cases/threat model
  - Existing XFR mechanisms, limitations and data leakage

  - Updates to existing specifications

  - XoT specification

  - Authentication mechanisms
  - Group policies for XoT transfers

-03 draft updates (Oct 2020)

- Use cases/threat model
  - Clarify that **threat considered is exposure of zone contents**, do not try to obfuscate the existence of a zone or that zone transfers are happening

- Updates to existing specs (more detail added)
  - **This draft now updates both RFC1995 (IXFR) and RFC5936 (AXFR)** (in the light of RFC7766 - ‘TCP Implementation Requirements for DNS’)
  - Clarifies how TCP connection reuse SHOULD be done, e.g.
    - **Persistent connections** and EDNS0 Keepalive to manage idle timeouts
    - Clients should pipeline XFR requests, use same connection for IXFR and AXFR
  - **Updates RFC7766** with regard to concurrent connections of different transports (treat TCP and TLS the same)

-03 draft updates (Oct 2020)

- XoT specification
  - **Authentication** - client MUST authenticate server using Strict DoT,
    server MUST authenticate client using mTLS (or IP based ACL)
    (the later section providing rationale for this approach has been updated)
  - Discuss **TLS connection handling** by primary
    - Potential concerns for authoritative servers now listening on TLS
      - Make clear that support for XoT is distinct from any form of ADoT
    - Outline how **Extended DNS Error codes** can be used to signal why none-XFR traffic might be refused on TLS connections
    - **Appendix which outlines operational and policy options** available to manage which TLS connections are accepted and which queries are answered
      - e.g. using proxies, requiring SNI, requiring TSIG, response policy, etc.

-03 draft updates (Oct 2020)

- **Remaining open questions**
    - Largely around new Extended DNS Error codes
        - What servers should return if they REFUSE non-XFR traffic on TLS connections
        - Declining XFRs because quota on concurrent transfers reached?

- **Latest on implementation work**
    - Patch to *NSD* to implement XFR-over-TCP connection reuse by default as a secondary, with a fixed idle timeout (EDNS0 Keepalive is a WIP)
    - Patch to *NSD* to use XoT as a secondary, tested against a TLS proxy
    - *BIND* are implementing DoT (announced an initial code update last week),
      => interop testing
    - Other implementers interested in working on this?

Moving forward

Moving forward

Reviews please!

Reviews please!

Reviews please!