Opportunistic ADoX deployment: The forgotten ‘big win’

Sara Dickinson (Sinodun IT) sara@sinodun.com

Johan Stenstam (Swedish Internet Foundation)
Leon Fernandez (Swedish Internet Foundation)
Philip Homburg (NLnet Labs)
Joe Abley (Cloudflare)

Quick bit of terminology

- For Recursive-Authoritative DoT/Q (ADoX) CLIENT policy:
  - **Opportunistic** (RFC 7435)
    - Encrypted and authenticated transport **when available**
    - **Fall back to clear text** when not
    - No security guarantees but “**Some Protection Most of the Time”**

Quick bit of terminology

- For Recursive-Authoritative DoT/Q (ADoX) CLIENT policy:
  - **Opportunistic** (RFC 7435)
    - Encrypted and authenticated transport **when available**
    - **Fall back to clear text** when not
    - No security guarantees but “**Some Protection Most of the Time**”
  - **Strict** (RFC 8310)
    - Client gets credentials via secure channel
    - Require encrypted and **authenticated** transport to the server
    - **Hard fail** if this is not available

Quick bit of terminology

- For Recursive-Authoritative DoT/Q (ADoX) CLIENT policy:

  - **Opportunistic** (RFC 7435)
    - Encrypted and authenticated transport **when available**
    - **Fall back to clear text** when not
    - No security guarantees but "**Some Protection Most of the Time**"

  - **Strict** (RFC 8310)
    - Client gets credentials via secure channel
    - Require encrypted and **authenticated** transport to the server
    - **Hard fail** if this is not available

Quick bit of terminology

- For Recursive-Authoritative DoT/Q (ADoX) CLIENT policy:
  - **Opportunistic** (RFC 7435)
    - Encrypted and authenticated transport **when available**
    - **Fall back to clear text** when not
    - No security guarantees but "**Some Protection Most of the Time**"

  - **Strict** (RFC 8310)
    - Client gets credentials via secure channel
    - Require encrypted and **authenticated** transport to the server
    - **Hard fail** if this is not available

In this talk, **Op-ADoX (Opportunistic ADoX)** means
"Some Protection Most of the Time"

Op-ADoX: How it started and how it is going...

- **Standards work**
  - **2018**: DPRIVE was re-chartered for recursive to authoritative
  - **2021**: Attempts at Strict (authenticated) mode abandoned
  - **2024**: RFC9539 - Unilateral Probing (Experimental) “blind probing”
    DELEG WG started
  - **2025**: DPRIVE closed

Op-ADoX: How it started and how it is going…

- **Standards work**
  - **2018**: DPRIVE was re-chartered for recursive to authoritative
  - **2021**: Attempts at Strict (authenticated) mode abandoned
  - **2024**: RFC9539 - Unilateral Probing (Experimental) “blind probing”
    DELEG WG started
  - **2025**: DPRIVE closed

- **Deployment**: 8 years on - Opportunistic ADoX deployments
  - 2 Open resolvers
  - 2 root servers, a few TLDs, one big hoster and a few names…

Deployment
has stalled

Op-ADoX - Why not more deployment?

- **Chicken-and-egg** problem is biggest issue
  - **Authoritative** operators
    - Concerned about **performance** of ADoX transports
    - Don’t like on/off nature of **blind probing**
    - Don’t feel **failure/error** paths are well understood
  - **Recursive** operators
    - Won’t enable until **more auths do it**
    - **Lack of implementation** in recursives

Op-ADoX - Why not more deployment?

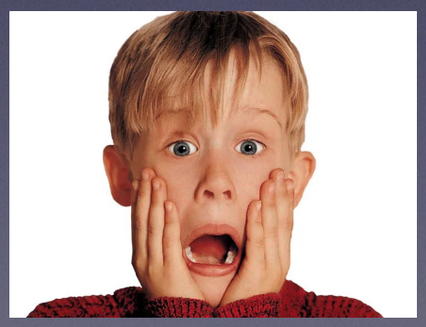
- **Chicken-and-egg** problem is biggest issue
  - **Authoritative** operators
    - Concerned about **performance** of ADoX transports
    - Don’t like on/off nature of **blind probing**
    - Don’t feel **failure/error** paths are well understood
  - **Recursive** operators
    - Won’t enable until **more auths do it**
    - **Lack of implementation** in recursives

IETF WGs overly focussed on solving Strict (the hard problem!)

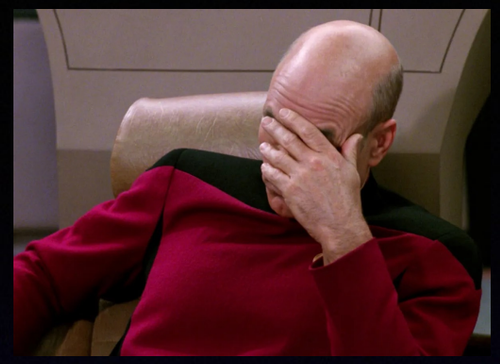
Op-ADoX - Why not more deployment?

- **Chicken-and-egg** problem is biggest issue
  - **Authoritative** operators
    - Concerned about **performance** of ADoX transports
    - Don’t like on/off nature of **blind probing**
    - Don’t feel **failure/error** paths are well understood
  - **Recursive** operators
    - Won’t enable until **more auths do it**
    - **Lack of implementation** in recursives

BUT! Bigger, more basic issues around just **deployment of the transport!**



Hindsight facepalm!

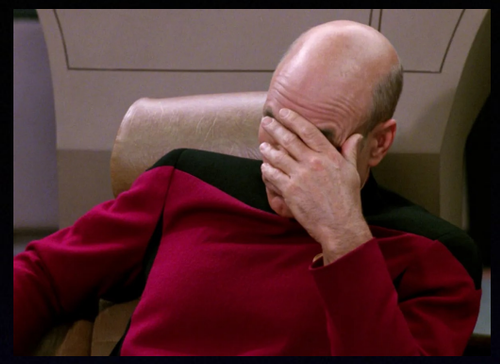


- Standards work
  - **2018**: DPRIVE was re-chartered for recursive to authoritative
  - Didn't refine with the **threat and landscape model**

We should have created 2 Goals at this stage:

- One to start **Opportunistic** roll out asap
- One to take time to figure out **Strict** mode

Hindsight facepalm!



- Standards work
  - **2018**: DPRIVE was re-chartered for recursive to authoritative
  - Didn't refine with the **threat and landscape model**

We should have created 2 Goals at this stage:

- One to start **Opportunistic** roll out asap
- One to take time to figure out **Strict** mode

NOT TOO LATE! Still time to actively
de-coupled transport deployment from deployment of Strict

Reasons to work On Op-ADoX

1. Op-ADoX protects end users

- Immediate protection from **passive surveillance**

1. Op-ADoX protects end users

- Immediate protection from **passive surveillance**
- Attacks must now **downgrade** or **MitM** connection(s)
  - Not only for **Surveillance**
  - **Injection attacks** on (non-)DNSSEC signed data

Attacks not defeated but SO MUCH HARDER!

1. Op-ADoX protects end users

- Immediate protection from **passive surveillance**
- Attacks must now **downgrade** or **MitM** connection(s)
  - Not only for **Surveillance**
  - **Injection attacks** on (non-)DNSSEC signed data

Attacks not defeated but SO MUCH HARDER!

- **Post-quantum** moves faster in TLS/QUIC than DNSSEC

2. Iterative, operator led effort

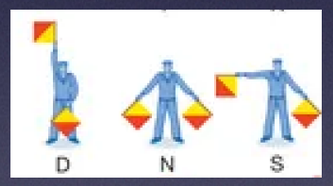
- **Operator led**
  - **No big changes** to parent zones or protocol
  - Operators can deploy as required, with **minimal risk** to service

- **Software implementation** required
  - **Easier** and quicker than registry updates
  - Deployment ease directly **reduces costs**

2. Iterative, operator led effort

- **Operator led**
  - **No big changes** to parent zones or protocol
  - Operators can deploy as required, with **minimal risk** to service

- **Software implementation** required
  - **Easier** and quicker than registry updates
  - Deployment ease directly **reduces costs**



- Work on **Signalling mechanism** for Op-ADoX in existing namespace
  - I-D: Authoritative DNS Transport Signalling (There is code!)
  - Even if only SVCB is signed, allows **experimentation** & incremental deployment
  - Start with **hint**, traffic % (exp. code point), [active probing/priming]
  - We can start work on this now **learn** what works and what is missing

3. Op-ADoX will be around a long time….

- Lets be realistic about Strict (hard-failures) for privacy reasons
  - Want and need **Early Strict adopters** and **Islands of Strict**
  - BUT likely **not mainstream for years to come**
    1. Requires entire resolution chain to be protected
    2. Unlike DNSSEC, resolvers may adopt risk-averse “best effort privacy”

3. Op-ADoX will be around a long time….

- Lets be realistic about Strict (hard-failures) for privacy reasons
  - Want and need **Early Strict adopters** and **Islands of Strict**
  - BUT likely **not mainstream for years to come**
    1. Requires entire resolution chain to be protected
    2. Unlike DNSSEC, resolvers may adopt risk-averse “best effort privacy”

- **Existing namespace** deserves attention!
  - Deploy ADoX and develop signalling here rather than back-port it
  - Some zone owners may never have access to Strict mechanisms

3. Op-ADoX will be around a long time….

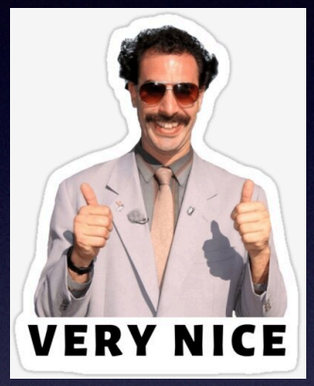
- Lets be realistic about Strict (hard-failures) for privacy reasons
  - Want and need **Early Strict adopters** and **Islands of Strict**
  - BUT likely **not mainstream for years to come**
    1. Requires entire resolution chain to be protected
    2. Unlike DNSSEC, resolvers may adopt risk-averse “best effort privacy”

- **Existing namespace** deserves attention!
  - Deploy ADoX and develop signalling here rather than back-port it
  - Some zone owners may never have access to Strict mechanisms

Proving Op-ADoX is viable paves the path for Strict

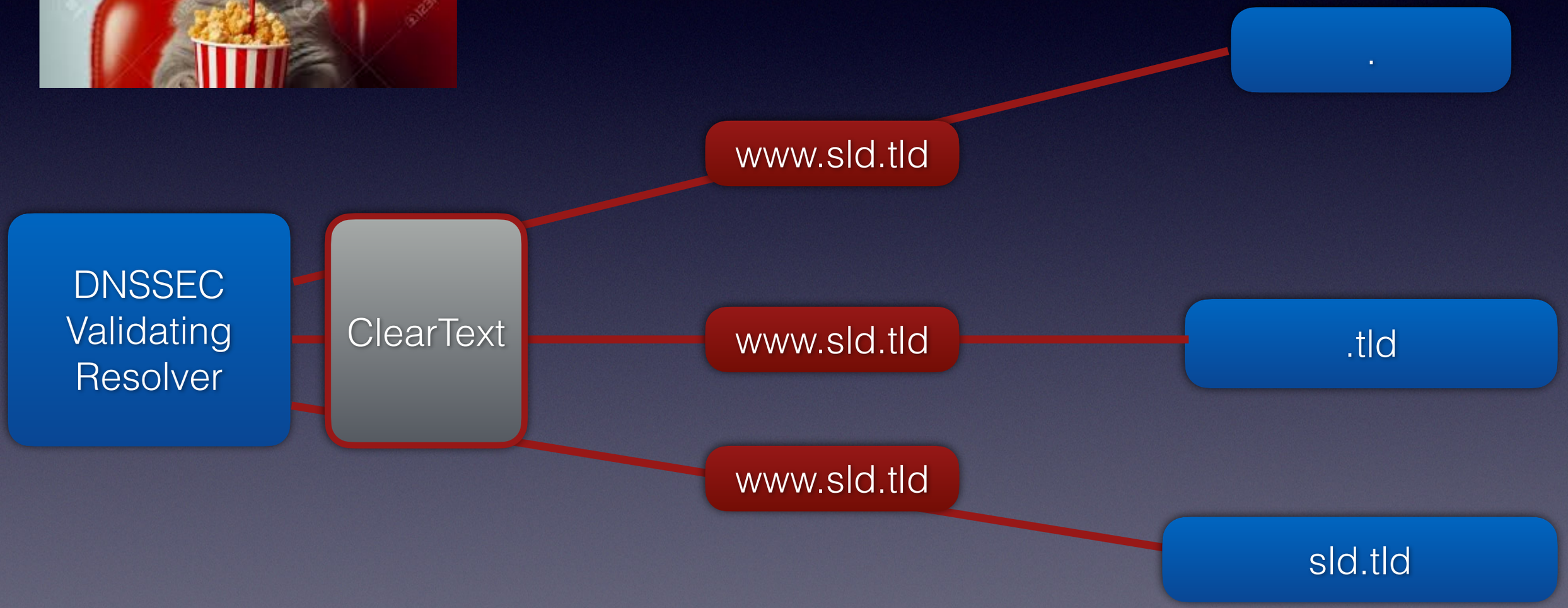
Thought experiment

- **Utopian future - 100% traffic protected by Strict**



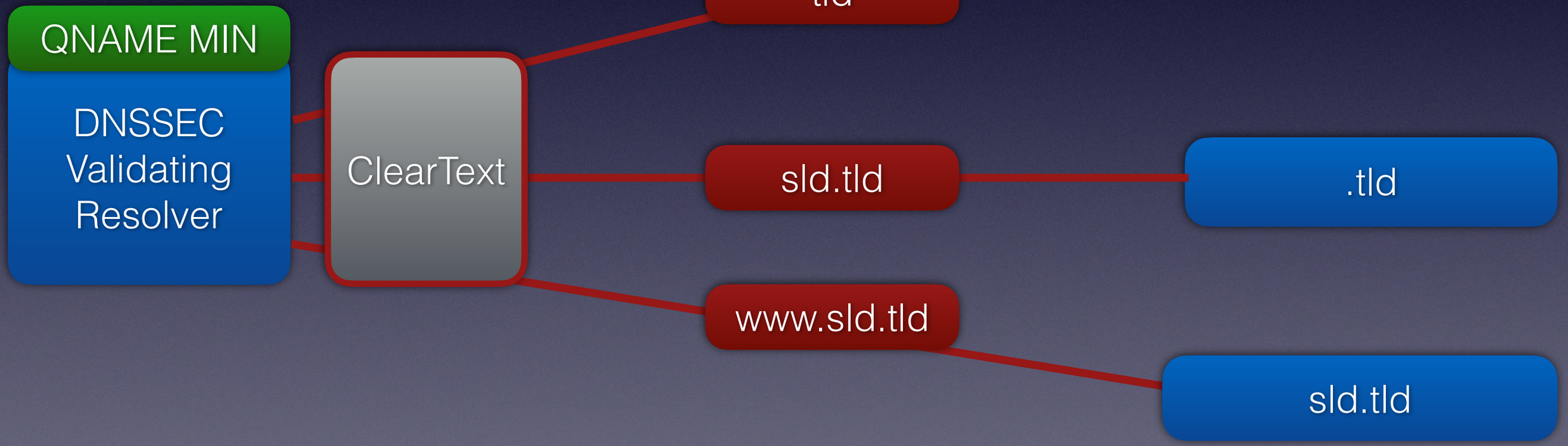
Thought experiment

- **Today, virtually all traffic clear text**



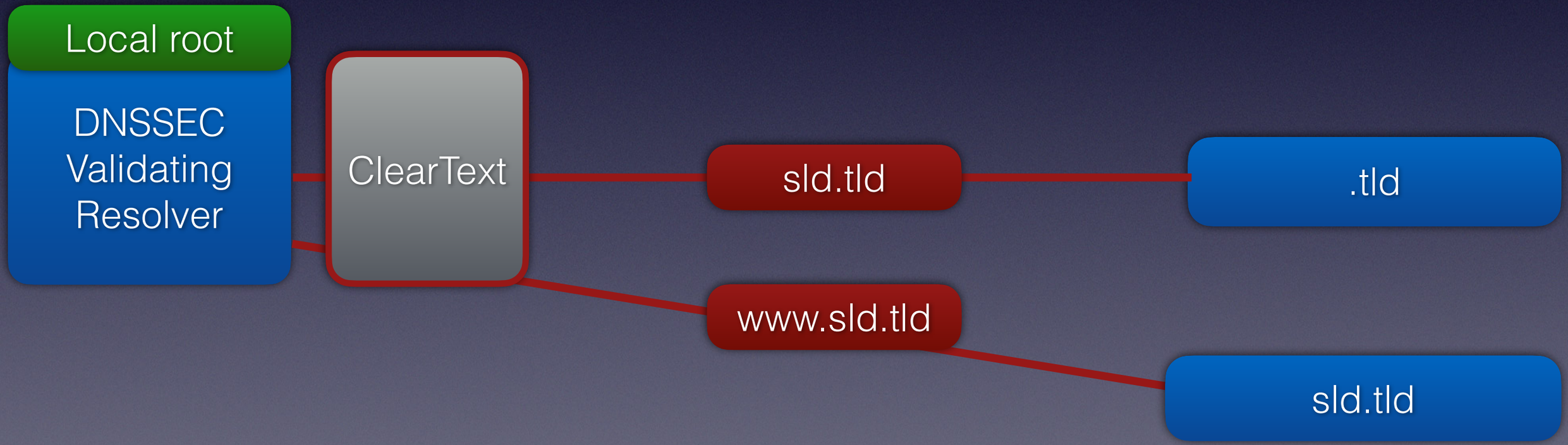
Thought experiment

- **Today, we also have mitigations that protect primarily root traffic**



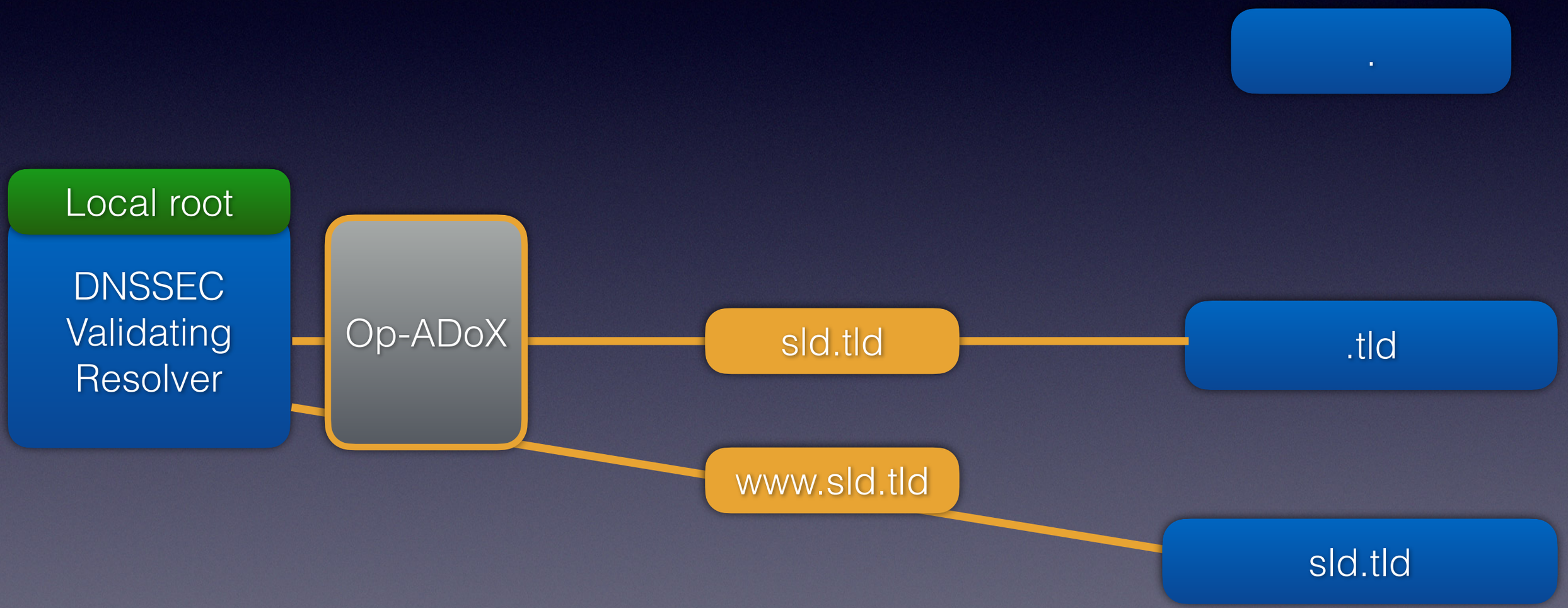
Thought experiment

- **Today, we also have mitigations that protect primarily root traffic**



Thought experiment

- **Op-ADoX at biggest operators will protect significant % of traffic**



Thought experiment

- **Op-ADoX at biggest operators will protect significant % of traffic**

Local root

DNSSEC Validating Resolver

Op-ADoX

sld.tld

.tld

www.sld.tld

sld.tld

<u>Most of the Traffic</u> has Some Protection Most of the Time

BIG WIN FOR END USERS!

.

Thought experiment

- **Op-ADoX at biggest operators will protect significant % of traffic**

Local root

DNSSEC Validating Resolver

Op-ADoX

Valid, achievable, cost-effective intermediate target state

BIG WIN FOR END USERS!

.

sld.tld

.tld

www.sld.tld

sld.tld

What else could we do?

- Implementation - DELEG aware resolvers should also do Op-ADoX (cart/horse?)
- We need a BCP - Work has started in OARCs new BCP framework
- Deployment Monitoring - working with OARC on periodic reporting

GOAL: Create confidence in deploying encrypted transports!!

What else could we do?

- Implementation - DELEG aware resolvers should also do Op-ADoX (cart/horse?)
- We need a BCP - Work has started in OARCs new BCP framework
- Deployment Monitoring - working with OARC on periodic reporting

GOAL: Create confidence in deploying encrypted transports!!

- End-user signalling of desired/required upstream privacy
  - EDNS0 option? Server returns information about transport used?

These goals still stand

- 3 Goals of RFC 9539 (2022):
  - **Protection** from **passive attackers** for recursive to authoritative DNS queries.
  - **A road map** for gaining real-world experience at scale with encrypted transports.
  - **A bridge to** some possible future protection.

These goals still stand

- 3 Goals of RFC 9539 (2022):
  - **Protection** from **passive attackers** for recursive to authoritative DNS queries.
  - **A road map** for gaining real-world experience at scale with encrypted transports.
  - **A bridge to** some possible future protection.

- **Immediate next steps**
  - Engagement and interaction with major DNS operators
  - Interop testing of Op-ADoX transport signalling across multiple implementations

These goals still stand

- 3 Goals of RFC 9539 (2022):
  - **Protection** from **passive attackers** for recursive to authoritative DNS queries.
  - **A road map** for gaining real-world experience at scale with encrypted transports.
  - **A bridge to** some possible future protection.

- **Immediate next steps**
  - Engagement and interaction with major DNS operators
  - Interop testing of Op-ADoX transport signalling across multiple implementations

SHORT TERM BIG WIN:
Focus on achievable improvements to end-user privacy with limited investments in both time and resources

Thank you!

Questions?