Trust Anchor IDs

draft-ietf-tls-trust-anchor-ids

Status Update

draft-ietf-tls-trust-anchor-ids-00 published

Plan

Collect feedback to organize discussion

Editorial changes to clarify initial design as needed

Work through decision points—tradeoffs considered in initial design, alternatives, etc.

Goal now is to collect decision points, get folks thinking. Detailed discussion to follow

Editorial Changes

Clarify description of current starting point:

- "Trust Anchor Identifiers" → “Trust Anchor IDs” (Issue #92)
- Define semantics of empty `trust_anchors` extension (Issue #91)
- Handshake failure vs opportunistic certificate (Issue #63)
- Improve guidance on what is advertised in ClientHello (Issue #99)
- ECH interaction (Issue #98)

File issues (https://github.com/tlswg/tls-trust-anchor-ids/issues) or mail if anything else is unclear

Provisioning Certificate Metadata

Current design

Maintain a *CertificatePropertyList* alongside certificate chain

Constructed by CA, passed through provisioning pipeline to TLS library.

Extensible to support other properties — only plumb this once (possible future uses like stapled SCTs)

Questions

Trade-offs between in-band vs out-of-band metadata (just one? both?)

Bikeshed syntax? Current draft is TLS-style with registry of 16-bit codepoints. Use ASN.1 attributes instead?

DNS Dependency

Current design

Support large (many CAs) PKIs with server-offer / client-select

Server lists available certificates (by trust anchor) in HTTPS/SVCB, client selects

DNS usage aligns with other IETF work (draft-ietf-tls-wkech, draft-ietf-tls-key-share-prediction), but adds deployment constraint.

Questions

Is this the design we want? Do we want less of a DNS dependency?

More compact ClientHello encodings? (https://github.com/bwesterb/go-ncrlite?)

Lean on retry flow to reduce the dependency?

Retry Flow

Current design

Servers send available alternate certificates alongside certificate

Clients can retry and request another if unacceptable, currently (like ECH) over another connection

Feedback from clients: useful for handling more complex policies

Questions

In-handshake retry? (Issue #53)

More efficient, but more complex

Which side initiates the retry impacts previous discussion

Identifiers

Current design

IDs are based on OIDs

- Reasonably short (once relative to RFC 9371 prefix)
- Easy to allocate

Question

Do we want some other ID scheme?

Proposal

Decide this last, may by determined by decision points

E.g. if different ClientHello encoding wants integers

Early Prototyping

Starting some early prototypes to gather implementation experience

Prototype draft-00 implementation in BoringSSL

Aiming to experiment between participating clients and participating servers

ID assignment tracking table added to GitHub

https://github.com/tlswg/tls-trust-anchor-ids/blob/main/assignments.md

Next Steps

Work through decision points

Other topics of interest?

File bugs at https://github.com/tlswg/tls-trust-anchor-ids/issues

Or mail the list