Google

TLS Trust Expressions

CA/B Forum F2F 62

asymmetric@google.com

May 2024

Agenda

01 Introduction

02 Current and Future PKI Challenges

03 Overview of Trust Expressions

04 Alternatives

05 Wrapping things up

06 Q&A

Trust Expressions are a mechanism for relying parties to transmit trust signals to subscribers so they can reliably serve a trusted certification path over TLS.

A TLS extension sent in the ClientHello, CertificateRequest, and Certificate messages

Trust Expressions are a mechanism for relying parties to transmit trust signals to subscribers so they can reliably serve a trusted certification path over TLS.

A TLS extension sent in the ClientHello, CertificateRequest, and Certificate messages

Supports both server and client authentication flows

Trust Expressions are a mechanism for relying parties to transmit trust signals to subscribers so they can reliably serve a trusted certification path over TLS.

A TLS extension sent in the ClientHello, CertificateRequest, and Certificate messages

Supports both server and client authentication flows

Trust Expressions are a mechanism for relying parties to transmit trust signals to subscribers so they can reliably serve a trusted certification path over TLS.

In support of a model in which subscribers can confidently provision and serve multiple credentials

A TLS extension sent in the ClientHello, CertificateRequest, and Certificate messages

Supports both server and client authentication flows

Trust Expressions are a mechanism for relying parties to transmit trust signals to subscribers so they can reliably serve a trusted certification path over TLS.

In support of a model in which subscribers can confidently provision and serve multiple credentials

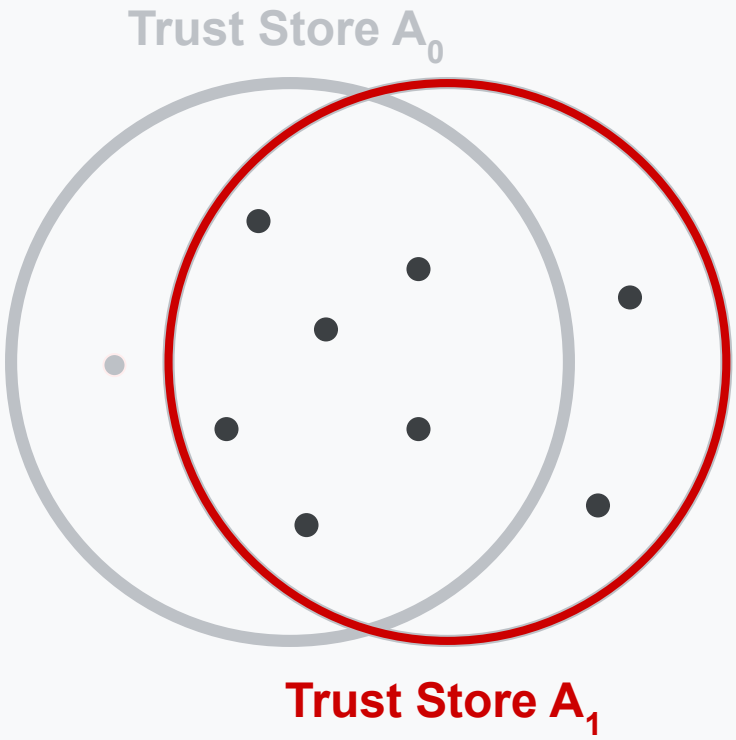
Includes end entity certificates and path to trust anchor

02

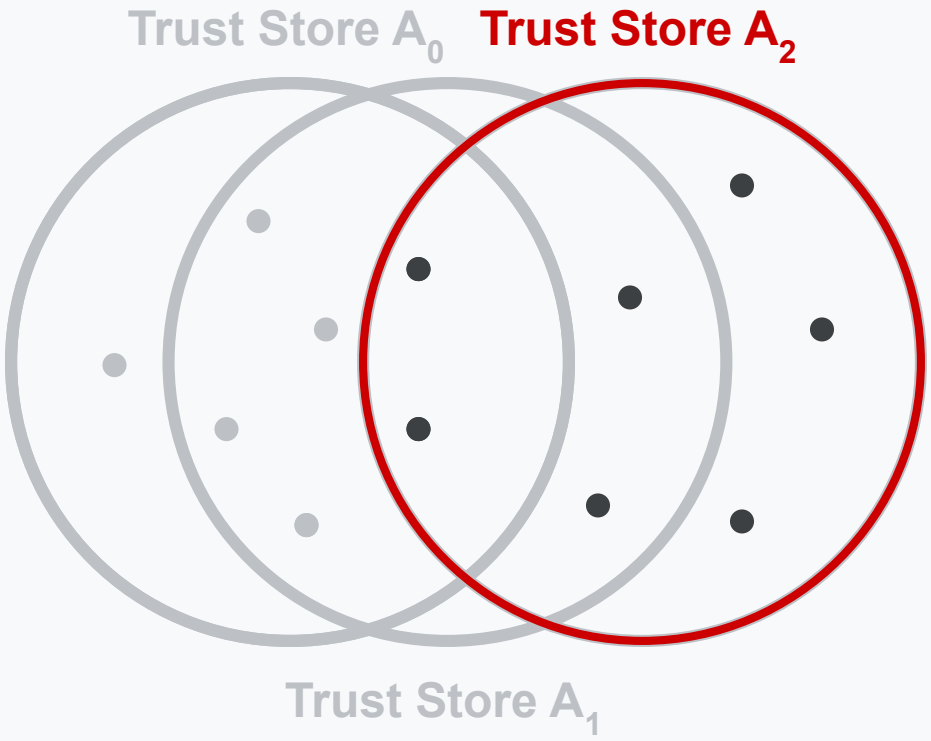
Current and Future PKI Challenges

Challenges with Trust Stores: Divergence

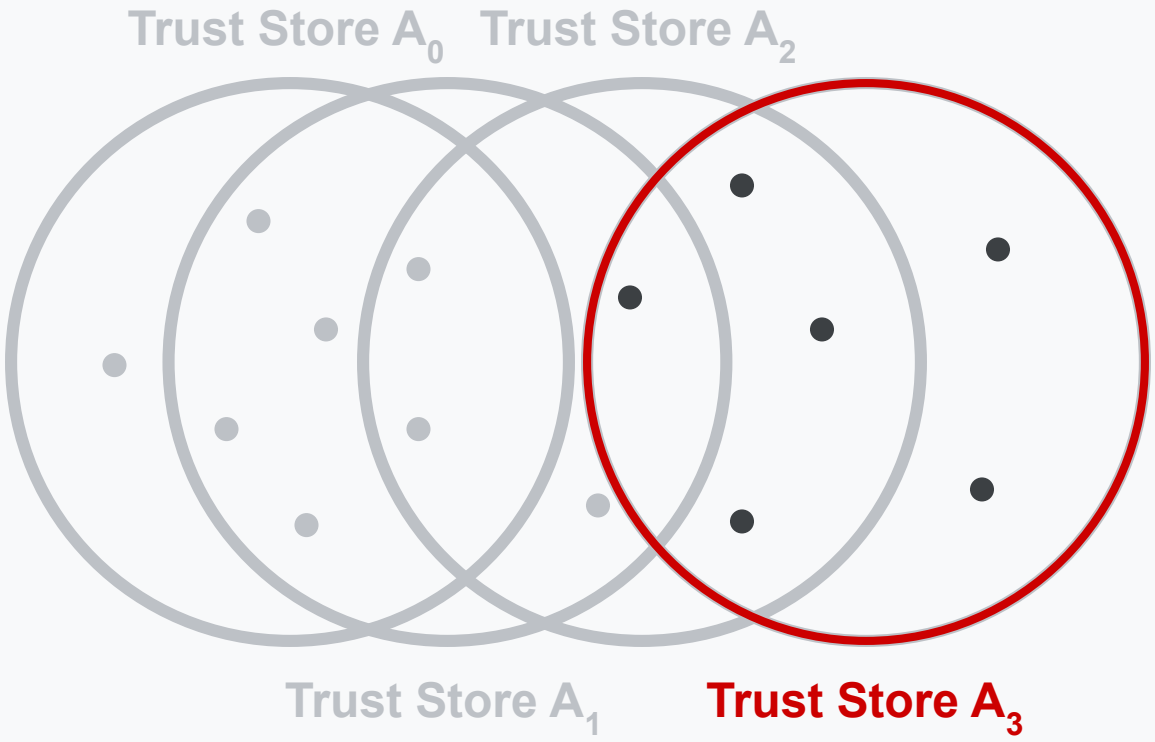
Challenges with Trust Stores: Divergence



Challenges with Trust Stores: Divergence

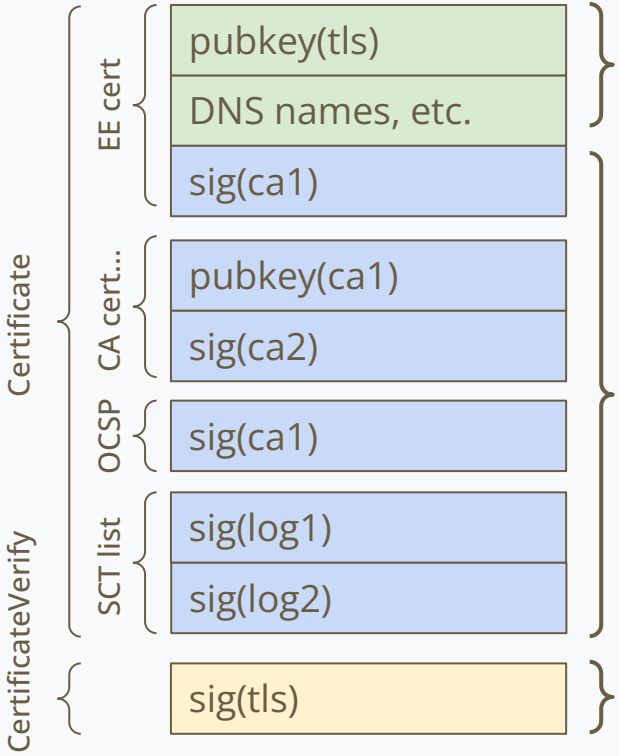


Challenges with Trust Stores: Divergence



Challenges with PQC in TLS: Size

Keys and Signatures in TLS handshake



Certificate and CertificateVerify contain:

A TLS key and application identifier

Proof from the PKI that identifiers match the key

Using the TLS key

Proof is determined by the PKI

Key, identifiers, and usage are determined by the protocol

Challenges with PQC in TLS: Size

≤0.4 Kb

EdDSA (Ed25519)

EdDSA, like ECDSA, is a highly performant signature scheme with very small keys and signatures. This algorithm is not secure against a CRQC.

- Pubkey: 32 bytes
- Sig: 64 bytes
- Sign*: 42K cycles
- Verify*: 130K cycles

≤1.8 Kb

RSA 2048

RSA is a signature scheme with tolerably small keys and signatures and fast signature verification. This algorithm is not secure against a CRQC.

- Pubkey: 272 bytes
- Sig: 256 bytes
- Sign*: 27M cycles
- Verify*: 27K cycles

≤14.8 Kb

ML-DSA-44

ML-DSA-44 is the smallest parameter set for the ML-DSA (nee Dilithium) signature algorithm. Signatures and keys are still extremely large.

- Pubkey: 1,312 bytes
- Sig: 2,420 bytes
- Sign*: 333K cycles
- Verify*: 118K cycles

≤28.3 Kb

ML-DSA-87

ML-DSA-87 is the largest parameter set for the ML-DSA (nee Dilithium) signature algorithm. Signatures and keys are prohibitively large.

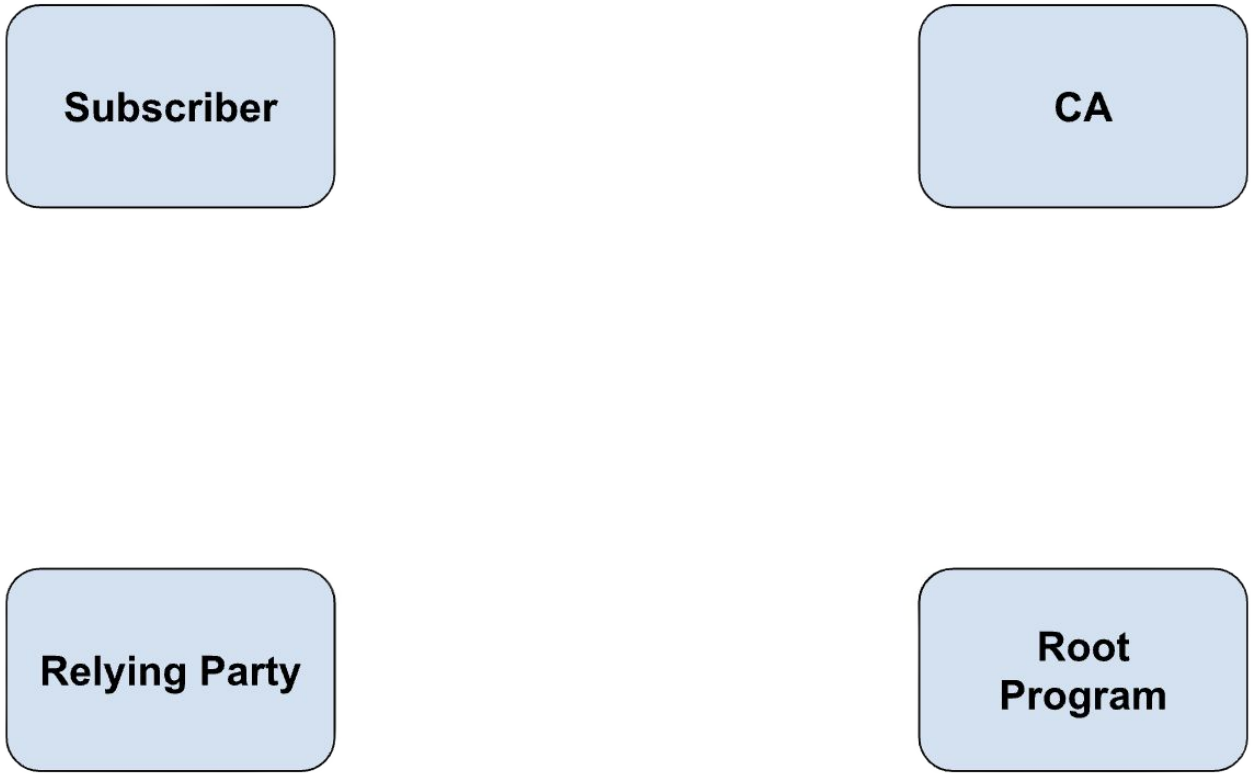
- Pubkey: 2,592 bytes
- Sig: 4,627 bytes
- Sign*: 642K cycles
- Verify*: 279K cycles

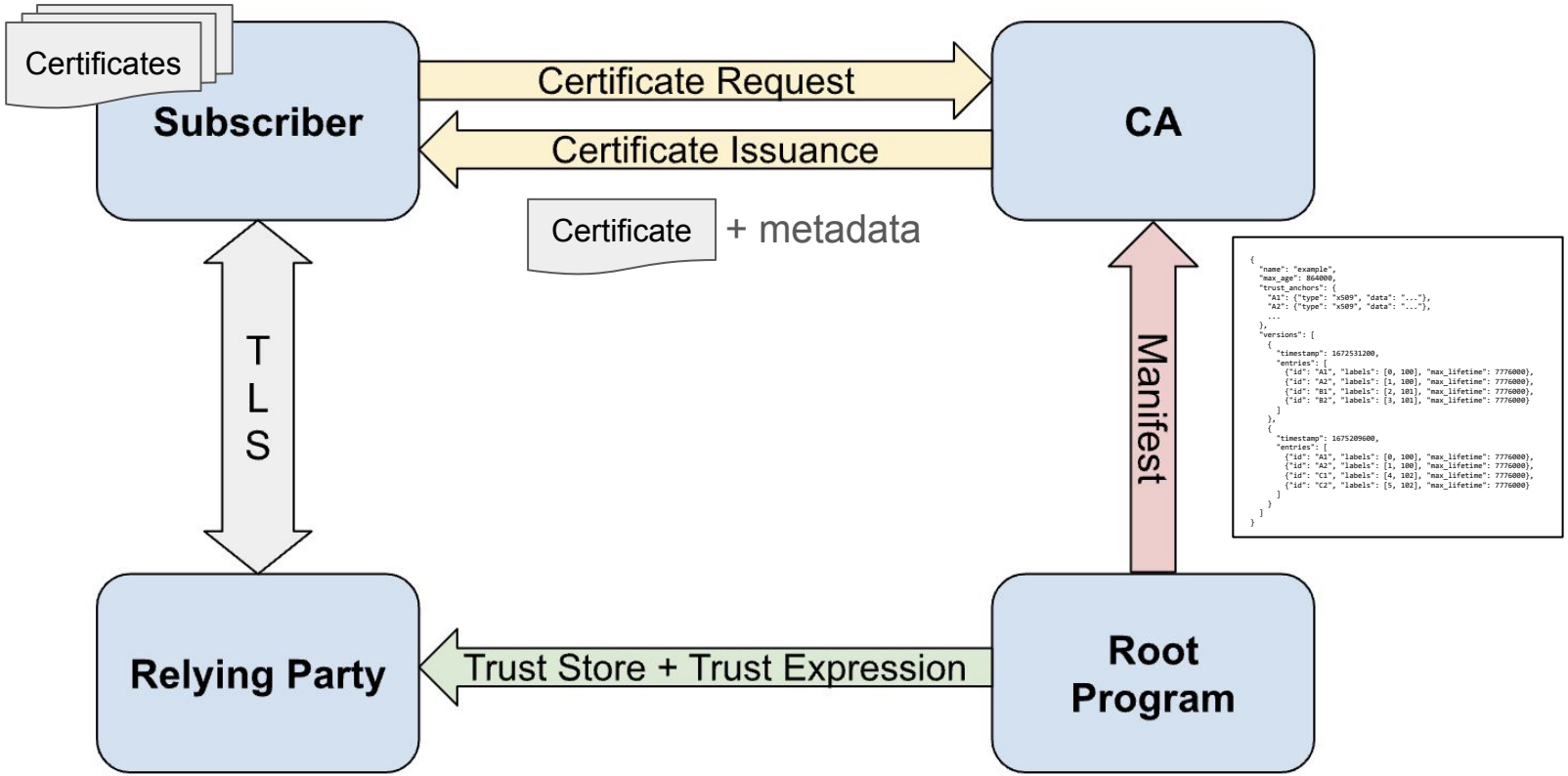
* cycles based on 2.5GHz processor

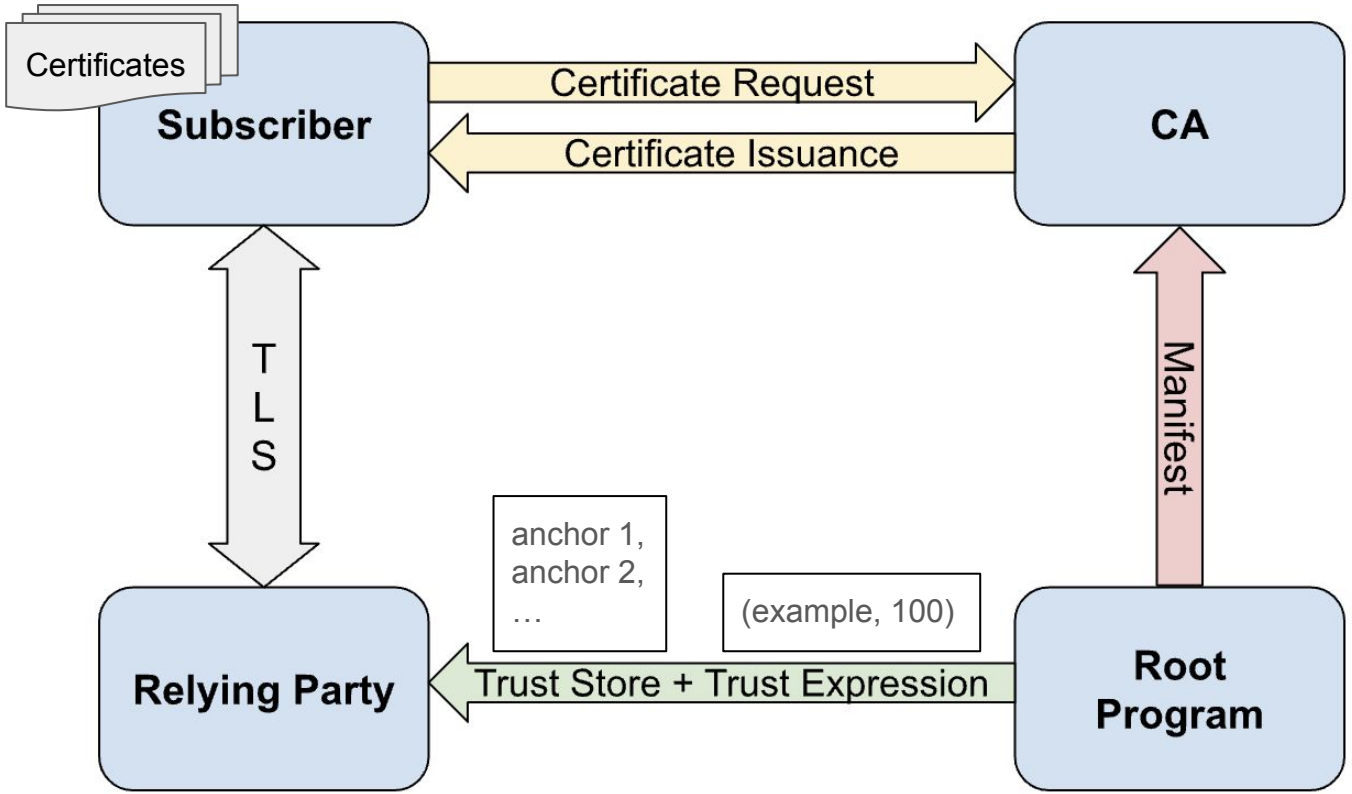
Different root programs may address these challenges in different ways!

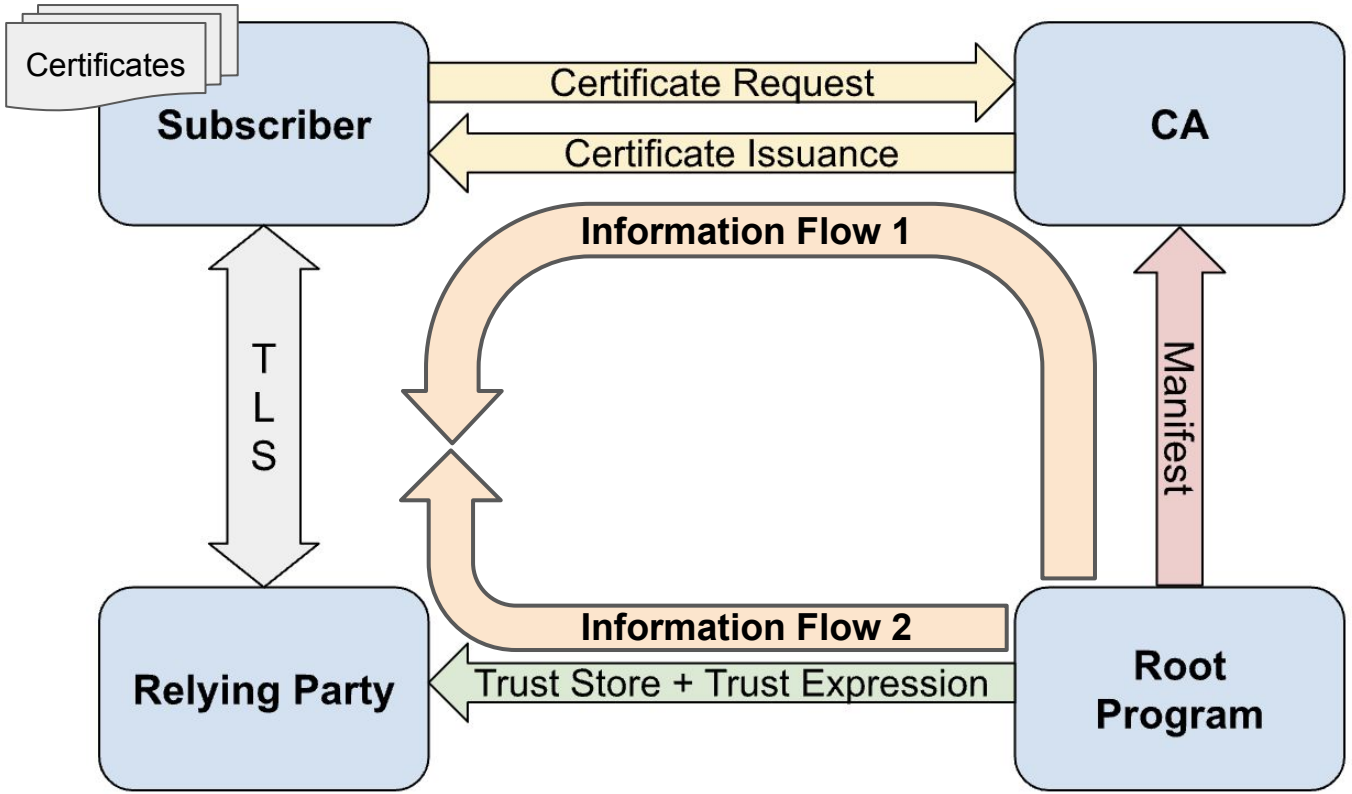
04

Overview of Trust Expressions


Subscriber
CA
Relying Party
Root Program

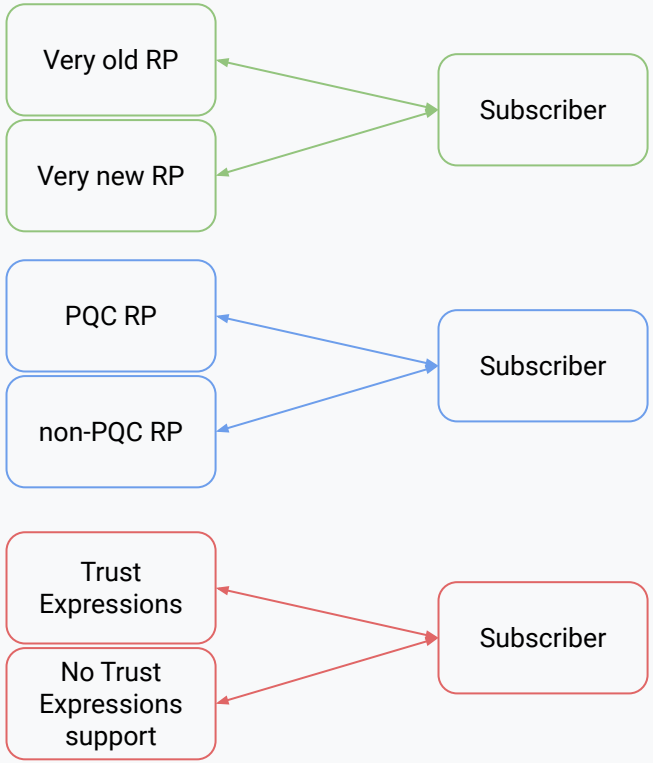

Certificates
Subscriber
Certificate Request
Certificate Issuance
CA
Certificate
+ metadata
TLS
Manifest
{
"name": "example",
"max_age": 864000,
"trust_anchors": {
"A1": {"type": "x509", "data": "..."},
"A2": {"type": "x509", "data": "..."},
...
},
"versions": [
{
"timestamp": 1672531200,
"entries": [
{"id": "A1", "labels": [0, 100], "max_lifetime": 7776000},
{"id": "A2", "labels": [1, 100], "max_lifetime": 7776000},
{"id": "B1", "labels": [2, 101], "max_lifetime": 7776000},
{"id": "B2", "labels": [3, 101], "max_lifetime": 7776000}
]
},
{
"timestamp": 1675209600,
"entries": [
{"id": "A1", "labels": [0, 100], "max_lifetime": 7776000},
{"id": "A2", "labels": [1, 100], "max_lifetime": 7776000},
{"id": "C1", "labels": [4, 102], "max_lifetime": 7776000},
{"id": "C2", "labels": [5, 102], "max_lifetime": 7776000}
]
}
]
}
Relying Party
Trust Store + Trust Expression
Root Program


Certificates
Subscriber
Certificate Request
Certificate Issuance
CA
TLS
Manifest
anchor 1,
anchor 2,
…
(example, 100)
Relying Party
Trust Store + Trust Expression
Root Program


Certificates
Subscriber
Certificate Request
Certificate Issuance
CA
Information Flow 1
TLS
Manifest
Information Flow 2
Relying Party
Trust Store + Trust Expression
Root Program

How do trust expressions help?

- Enable sites to serve certificates to mutually-incompatible client trust stores
- Ease the ability for the ecosystem to transition smoothly across major paradigm shifts without relying on flag days
- Gracefully fallback to existing behavior if trust signals are absent or unrecognized



04

Alternatives

Alternatives

Preloaded Intermediates

- In lieu of dynamic path building, pre-transmit CAs to relying parties
- Requires clients accept updates over time to continue functioning

Abridged CA Certs

- Create a compression dictionary for intermediate CA certificates
- Imposes some restrictions on agility in exchange for compression

certificate_authorities

- Relying party transmits a list of X.509 names of trust anchors
- Transmitting trust anchors takes over 13 Kb in modern browsers

Cross-signatures

- Old trust anchors sign new ones for backwards compatibility
- Challenges with incentives, degraded performance for older clients

05

Wrapping things up

How to get involved

- Read the Trust Expressions explainer
- Come find me between sessions to chat
- Read the IETF draft
- Prototype and experiment with Trust Expressions
- Provide feedback on tlswg mailing list or filing GitHub issues
- Reach out to us by way of chrome-root-program@google.com

References

Trust Expressions Draft: https://datatracker.ietf.org/doc/draft-davidben-tls-trust-expr/

Trust Expressions GitHub: https://github.com/davidben/tls-trust-expressions

Certificate_authorities: https://datatracker.ietf.org/doc/html/rfc8446#section-4.2.4

Abridged Certs Compression: https://datatracker.ietf.org/doc/draft-ietf-tls-cert-abridge/

TLSWG Mailing List: https://mailarchive.ietf.org/arch/browse/tls/

Cloudflare PQC Blog Post: https://blog.cloudflare.com/pq-2024

06

Q&A