Transparency Services

Orie Steele <orie@transmute.industries>
Henk Birkholz <henk.birkholz@ietf.contact>

Transparency Logs and Verifiable Data Structures

Agenda

- Review of Transparency Services
- Review of Common Primitives
- Interoperability Opportunities
- Q & A

Certificate Transparency

What certificates have been issued for this domain?
Is there any evidence of compromise based on the audit log?
When was the certificate created (different than the validity period of the certificate)?

- https://letsencrypt.org/docs/ct-logs
- https://crt.sh/?q=sigstore.dev
- https://datatracker.ietf.org/doc/html/rfc6962

Key Transparency

Which public keys are used for this phone number with this messaging provider?
Is the public key I have for a phone number, the same public key you have?
When have public keys changed?

- iMessage Key Transparency
  - https://security.apple.com/blog/imessage-contact-key-verification/
  - https://github.com/google/keytransparency/
  - https://datatracker.ietf.org/doc/draft-ietf-keytrans-architecture/

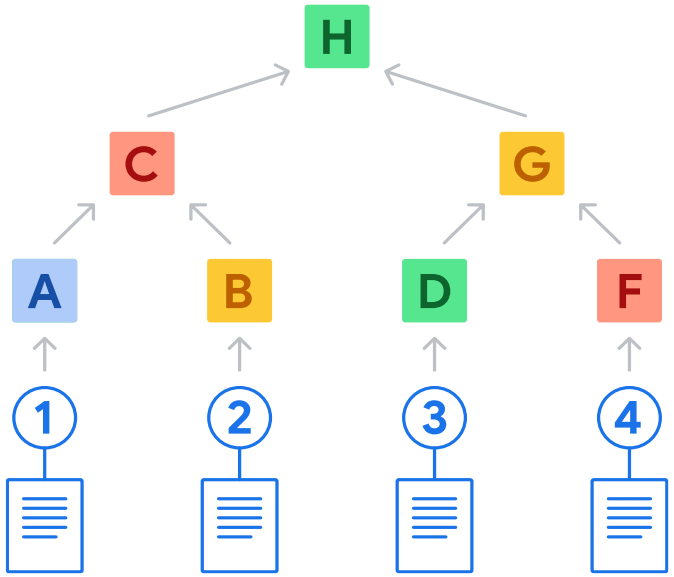
Software Transparency

What signatures have been issued for from this email address?
Is there any evidence of compromise based on the audit log?
When was the signature created?
Has the software changed since it was signed or included in a transparency log?

- https://www.sigsum.org/
- https://sigstore.dev/
- https://datatracker.ietf.org/doc/draft-ietf-scitt-architecture/

Interoperability Opportunities

- Transparency Log
  - SHA-256 Binary Merkle Tree (H)
    - Tiles - tile(0,0) aka (A,B,D,F)
    - Inclusion Proofs - (A,C,H)
    - Consistency Proofs - (A,C,H) to next tree head
- Transparency Log Entries
  - (1) Certificates / Public Keys
  - (2) COSE / JOSE / PGP Signatures
  - (3) COSE Receipts … "signed inclusion proofs"
  - (4) SCITT Receipts … "signed binary transparency proofs"
  - https://github.com/sigstore/rekor/blob/5d0bb6ed7826fe895aa0775a5e148b480d7ac3d3/pkg/types/cose/v0.0.1/entry.go#L62



Shared Primitives

Commonalities between sigstore and scitt

- Digital Signatures
  - P-256 ECDSA as described in RFC6605
  - Ed25519 EdDSA as described in RFC8032
- Certificates
  - Encryption with TSL 1.3 as described in RFC8446
  - Document Signing as described in RFC9336
- Authentication / Authorization
  - JSON Web Tokens as described in RFC7519
  - OAuth 2.1 - draft-ietf-oauth-v2-1
  - OpenID Connect - openid.net/foundation
- Media-Types
  - JSON (STD90)
  - CBOR (STD94)

SIGSTORE For SCITT Issuers

Putting Things Together

- Using Sigstore to produce:
  - SCITT Signed Statements (Signatures over arbitrary content)
    - Issuer Certificate is Stored in Sigstore (No Change)
    - Signature is COSE Sign1 (Supported by Rekor?)
  - SCITT Receipts (Signatured over merkle tree proofs)
    - Issuer Certificate is Stored in Sigstore (No Change)
    - Inclusion/Consistency Proof is retrieved from Trillian
    - Receipt is COSE Sign1 (Supported by Rekor?)

Summary of Difference

Consensus Opportunities

- Sigstore
  - Transparency Log is a SHA-256 Binary Merkle Tree
    - Inclusion and Consistency Proofs Supported
  - Identity Assurance Provided by Open ID Connect
  - Identity Credential short lived x509 Certificate
  - Signature format agnostic (COSE supported, but not the only choice)
- SCITT
  - SCITT is a collection of proposed standards, not a standalone product
  - Transparency Log agnostic … SHA-256 Binary Merkle Tree Supported
  - Identity Credential Agnostic … x509 Certificates Supported
  - Signature format MUST be COSE … Rekor Supported