Human Seal

HUMAN SEAL · TESSERA · W154.2

Tessera Passport and Handshake, v0.1

Open interoperability profile for verifiable AI-agent trust, built on the Human Seal cryptography.

Draft profile for review, not a ratified standard.

Built 2026-09-20. Live status is in the open items register. Reference implementation: FLOCORE.

W154.2. The open interoperability profile that lets AI agents discover each other, prove identity, and exchange verifiable receipts of their actions, with no central broker. Built on the same cryptography as the Human Seal, and interoperable with the emerging IETF signed-action-receipt work (ASQAV). Draft, 2026-09-19.

Design goals

- Offline verifiable: any party checks a passport and a receipt against a published key, with no call to a broker.
- Reuse, do not reinvent: Ed25519 signatures (RFC 8032), sha256 content binding (FIPS 180-4), did:web trust root, and canonical JSON, the same primitives the Human Seal already uses.
- Add the human: the profile carries the human-approver identity that machine-only receipt formats lack.

The Tessera Passport

A passport is a small, signed object an agent presents so others know who it is and what it may do.

Fields:

- did: the agent's decentralized identifier (did:web bound to a domain the operator controls).
- tenant: the organization the agent acts for.
- capabilities: the action types the agent is permitted to propose or perform.
- seal_endpoint: where a verifier can fetch the Human Seal receipts the agent produces.
- public_key: the Ed25519 verification key (also resolvable from the did document).
- issued_at, expires_at: validity window.
- issuer: the did of the issuer (the operator or Tessera authority).
- signature_ed25519: signature over the canonical passport (all fields above).

Illustrative passport (values are examples):

```
{
  "did": "did:web:agents.example.com:orders-bot",
  "tenant": "example-co",
  "capabilities": ["quote.request", "order.place", "invoice.read"],
  "seal_endpoint": "https://agents.example.com/seals",
  "public_key": "<base64url ed25519 key>",
  "issued_at": "2026-09-19T00:00:00Z",
  "expires_at": "2026-12-19T00:00:00Z",
  "issuer": "did:web:humanseal.world",
  "signature_ed25519": "<base64url signature over the canonical object>"
}
```

The handshake (discover, identify, verify)

1. Discover: fetch the counterparty agent's passport from its did:web location (the /.well-known path) or from a presented token.
2. Identify: verify the passport signature against the public key, confirm the key resolves from the did document, and check it is within its validity window.
3. Authorize: confirm the action you are about to rely on is within the passport's capabilities.
4. Act and verify: when the agent takes a world-affecting action, it produces a Human Seal receipt. The relying party verifies that receipt offline (signature, action content hash, approver), exactly as the Human Seal standard defines.

No step contacts a central Tessera broker; every check is against published keys.

Interoperability with IETF ASQAV

The emerging ASQAV signed-action-receipt draft (an individual Internet-Draft, not endorsed by the IETF) signs a machine policy decision with Ed25519 over RFC 8785 canonical JSON. Tessera uses the same signature algorithm and adds the human-approver identity (the Human Seal) that ASQAV does not carry. The canonical forms are not identical: ours (documented exactly on the conformance page) differs from RFC 8785 in string escaping and number formatting, and produces the same bytes only for ASCII-only data with integer numbers. Aligning a future version with RFC 8785 is a proposal, not a decision, and would be versioned so v0.1 receipts stay verifiable. The intent is to interoperate rather than compete.

Verification algorithm (passport)

1. Remove signature_ed25519 and public_key; canonicalize the remaining object by the v0.1 rules (sorted keys, no whitespace, non-ASCII escaped, numbers as written; see the conformance page).
2. Verify the Ed25519 signature over those bytes against public_key.
3. Independently resolve public_key from the did document (did:web to https to did.json) and confirm it matches.
4. Check issued_at is in the past and expires_at is in the future.
5. Only then rely on capabilities.

Status and next

- The rail is LIVE (verified 2026-09-19): https://fo.flocore.tech/.well-known/agent-manifest returns HTTP 200 with a real flocore.agent-manifest/v1, and https://fo.flocore.tech/.well-known/did.json returns the live did:web:fo.flocore.tech document with an Ed25519 key. Tessera v0.1 is the named, open profile over that rail.
- A signed sample passport is published at humanseal.world/tessera/sample-passport.json: it verifies offline against its embedded key (checked with the reference library on 2026-09-20) and fails when any field is altered. It is signed with a throwaway key for illustration; a production passport is signed with the issuer's did:web key (did:web:humanseal.world, live since 2026-09-20).
- PUBLISHED: the profile is on humanseal.world/tessera.html, with this PDF and the one-pager as downloads.
- Still open: review by outside implementers, and the RFC 8785 alignment proposal above.
- This is a draft profile for review, not a ratified standard.