RTTP and IQA - reviewer brief

Two open URI-scheme specifications, with reference implementations you can check yourself. Reviewer version of 20 September 2026. One A4 sheet.

> **If you read one thing:** the four self-test commands in section 4, about five minutes, verify the published conformance vectors offline on your own machine - no account, no repository, no network trust.

1. The problem

Machine-to-machine communication inherits the web's trust model in exactly two places: **who you are** depends on a name registry you must query, and **what you signed** depends on an issuer you must trust. In constrained, air-gapped or adversarial environments those two dependencies are precisely the parts that are unavailable, seized or untrustworthy - so verification either fails open or fails entirely, where it matters most.

2. What the two specifications do

RTTP - intent addressing. A URI of the form `rttp://<authority>/<action>` names a claim of intent directed at an identified subject. The subject address is derived by computation from the authority, so no registry, resolver or lookup service is consulted. Agent-to-agent messages use a fixed 128-byte `PulseHeader128` frame; extension space sits in bytes that were already zero, so older readers keep working.

IQA - subject-attestation addressing. A URI of the form `iqa://<subject>.<organ>.<root>[/<action>]` names the attestation standing of a subject, as reported by one of three named authority organs. The organ set and the four actions are closed sets; the URI carries no proof and no credential.

Common design rules. Malformed input is rejected, never normalised. There is no `userinfo`, no port, no query and no fragment, so there is no field in which to place a credential. Neither specification defines a reporting endpoint, a resolver or an account.

3. Status as of 20 September 2026

Item	State
`rttp` URI scheme, IANA registration	Submitted **2026-09-17** under RFC 7595, ticket **#1459939**; under Expert Review; **not registered**
`iqa` URI scheme, IANA registration	Submitted **2026-09-18** under RFC 7595, ticket **#1459963**; under Expert Review; **not registered**
Internet-Drafts	`draft-li-rttp-intent-addressing-01` and `draft-li-iqa-subject-attestation-00`; Independent Submission, Informational, published
Implementations	Three per protocol (Python, JavaScript, Rust), no shared code; zero dependencies by default
Distribution	PyPI `rttp` 1.2.6 / `iqa-org` 1.2.7; npm `@aicent/rttp` 1.2.6 / `@aicent/iqa` 1.2.7; crates.io `rttp` / `iqa-org` (pre-release)
Licence	Code Apache-2.0; specifications published openly

Check the "not registered" claim yourself: there is no `rttp` or `iqa` row in the IANA URI scheme registry (link in section 8).

4. Verify it yourself

Five minutes, Python, one protocol:

```
$ pip install rttp
$ python -m rttp.selftest
[PASS] all 29 checks passed (3 skipped)

$ pip install iqa-org
$ python -m iqa.selftest
[PASS] all 35 checks passed (1 skipped)   # core install
```

JavaScript / Node: `npx @aicent/rttp -> [PASS] all 25 checks passed`; `npx @aicent/iqa -> [PASS] all 32 checks passed`. **The Ed25519 checks are skipped in the default install;** `pip install rttp[ed25519]` (or `iqa-org[ed25519]`) enables them, giving 53 and 59 checks respectively.

Why this is worth more than a table of numbers: the conformance vectors ship *inside* the installed package, so the self-test replays them from the package, offline, with no repository checkout and no network access. An implementation that disagrees with the vectors fails on your machine, not ours. **Then, about ten more minutes:** read the two Internet-Draft plain-text versions (12 and 14 pages) and the two specification pages - links in section 8.

5. What we are not claiming

- **Not registered, not standardised, no RFC yet** - both schemes are submitted and under review.
- Specifications with reference implementations, **not a product**; no commercial customers.
- No transport layer and no client library.
- Post-quantum and staking/economics mechanisms are **out of scope and not implemented**, and the specifications say so in writing.
- Package downloads are **not** presented as adoption: today they are dominated by automated infrastructure traffic.
- An `iqa` URI names the subject being attested, so the scheme **cannot be used for anonymous reference** - stated in the draft's privacy section.

6. What would help

Any one of these is useful, and all of them are small:

1. **Replay the vectors** (about 20 minutes) and tell us the result - including failure.
2. **Use it as a course or student project.** We will answer questions and publish the resulting report unedited, including conclusions we do not like.
3. **Critique one specific claim.** A good candidate: whether "no issuer" survives the replay, correlation and linkability threats listed in the drafts.

Context, stated plainly: we are seeking grant funding for the public layer of this work (specifications, reviewer-facing documentation, an independent implementation, an external security review). If, after looking, you consider the work sound, an independent evaluation or a short technical assessment would be directly useful to that process. We would rather hear that it is unsound than hear nothing.

7. Research questions this raises

Offered as open problems, not claims. Each has a working baseline artifact, which is what usually makes a study feasible.

1. **Unknown URI schemes in browsers and operating systems.** How are they handled, and what does fail-closed look like in practice? A conformance vector set exists as a baseline.
2. **Addressing without a registry.** Usability and interference-resistance when the address is computed rather than looked up.
3. **Attestation without an issuer.** If the certificate authority is removed, what actually breaks, and what threat remains?
4. **Consent models for protocol handlers.** Is "no navigation, scheme prefix check, explicit consent, parsing is not attestation" sufficient in practice?
5. **Empirical study of the standards process.** Two live RFC 7595 registration requests, including a documented naming negotiation, as primary material.

8. Links

Specifications (stable)
IQA: https://iqa.org/RFC-009/
RTTP: https://rttp.com/RFC-002/
Plain text: https://iqa.org/RFC-009/source-v1.2.6.txt · https://rttp.com/RFC-002/source.txt

Internet-Drafts
https://datatracker.ietf.org/doc/draft-li-rttp-intent-addressing/
https://datatracker.ietf.org/doc/draft-li-iqa-subject-attestation/
https://www.ietf.org/archive/id/draft-li-rttp-intent-addressing-01.txt
https://www.ietf.org/archive/id/draft-li-iqa-subject-attestation-00.txt

Packages and sources
PyPI: https://pypi.org/project/rttp/ · https://pypi.org/project/iqa-org/
Rust docs: https://docs.rs/rttp · https://docs.rs/iqa-org
Sources: https://github.com/Aicent-Stack

Registry
IANA URI schemes: https://www.iana.org/assignments/uri-schemes/uri-schemes.xhtml

Contact. ShaoBao Li - independent maintainer, project hosted at the GitHub Organization `Aicent-Stack`. Email: lee@iqa.org (IQA) or lee@rttp.com (RTTP).