TECHNICAL CONTRIBUTION / OWNER REVIEW

HTTP6: Problem Statement and Architecture for Hereditary Secure Application Channels

Individual Internet-Draft source candidate — for author review and routing

Frank Dylan Rosario

November 9, 2026

Local draft candidate — not submitted

Abstract

This document proposes an architecture for preserving human-originated authority through distinct workload instances and bidirectional application-message sequences. A human admission establishes a policy-bound root. Each permitted descendant has its own identity, authenticated ancestry, bounded permissions, and current runtime state. Application records bind requests and responses to the live instance, intended peer, canonical operation, policy, and consumed sequence position. HTTP6 is the working project name, not an assigned version of HTTP.

The work distinguishes a standards-compatible profile carried over HTTPS from a native-channel research target with independently enrolled service trust and a separately specified authenticated key establishment. The purpose of this first draft candidate is to obtain feedback on problem scope, overlap with existing standards, required component specifications, and the proper IETF or research venue. The Rosario–Wang private-reference construction and deployment assurance artifacts are separately identified dependencies, rather than implicit assumptions about the complete wire protocol.

1. Status and intended contribution

This Markdown is an owner-review source candidate. It does not include a representation that it has been submitted under IETF contribution terms, adopted by a working group, or accepted as a standard. On release, the author must select the proper submission format and boilerplate, confirm the draft name, provide contact details, and make the required contribution and IPR representations. No registry assignment is requested by this version. [R15; R28–R29]

The initial request is DISPATCH routing, because the work crosses application semantics, secure-channel authentication, workload identity, and attestation. The current DISPATCH charter covers the relevant ART, SEC, and non-transport WIT discussion. Any transport-specific extension is coordinated with the appropriate existing group or separately agreed venue. HTTPBIS currently requires rechartering for new HTTP-version work. [R16–R17]

2. Problem statement

An authenticated connection and a successful human login do not by themselves describe the authority of every software instance acting later. A service may spawn workers, delegate to another organization, invoke tools, and continue after its original operator is offline. The resource needs to distinguish the original human approval, the current requesting instance, the delegated operation, and the resulting business effect.

Existing standards address many of these components. TLS supplies authenticated channel protection; sender-constrained OAuth binds proof to a key and request context; workload identity and attestation supply independently meaningful execution evidence. HTTP6 proposes a cohesive application profile in which inherited authorization, response authenticity, instance generation, one-use action state, and policy-controlled child creation remain bound throughout the sequence. A standards proposal must compare this integration with composed existing mechanisms rather than contrast it only with poorly configured bearer credentials. [R19–R20; R24; R31]

The interoperability question is whether independent implementations can exchange the same admitted authority context and produce identical acceptance decisions for requests, responses, revocation, recovery, and delegation transitions. A private single-vendor application can be built without Internet standardization; public protocol work requires interoperating implementations and participating stakeholders. [R15]

3. Terminology and identity boundaries

A **human admission** is a context-bound authorization event using the selected human authentication profile. A **root instance** is a measured, distinct execution principal admitted by that event. A **child instance** is separately admitted under an authenticated parent delegation. A **twin** is the peer responsible for matching authenticated channel state; the word describes cryptographic state machines, not physical entanglement.

A **policy commitment** identifies canonical authorization rules and their interpreter versions. An **operation identifier** names a business effect independently of transport retries. A **channel epoch** identifies the active endpoint pairing and key generation. A **record position** identifies a direction, stream, epoch, and sequence. A **service descriptor** binds an intended origin or service identifier to an independently trusted service key and profile.

The human approval and the service owner's authority are separate. The client carries a human-rooted delegation; the server authenticates under its own admitted service identity. Their connection binds both. Standard child grants carry the approved origin of authority, not the human's secret commitment. Every child proves possession of its own instance key.

4. Goals and profiles

The design goals are action-specific authorization, distinct workload provenance, monotonically constrained delegation, authenticated bidirectional application records, controlled recovery, and explicit lifecycle cessation. A native enrolled-peer profile additionally investigates independence from public-CA server certificates and the TLS protocol on that connection.

W1 compatibility carries the inner HTTP6 application records over existing HTTPS and retains its outer TLS trust. **W2 managed client** places the protected state in a controlled workspace or native companion with an explicit carrier. **N1 native** uses an HTTP6-specific authenticated handshake and record binding on an explicitly selected transport. The N1 suite and byte encoding remain work to freeze and review. Ordinary QUIC retains its TLS security interface; a different handshake cannot be represented as standard QUIC without specifying and reviewing that different binding. [P1; R24; R27]

Public-Web-PKI independence is a trust-model choice. TLS already supports PSK authentication and raw-public-key mechanisms exist; certificate-free wording alone is not the claimed innovation. N1 supplies an enrollment procedure, service-key authority, revocation and recovery process, and independently authenticated first-contact binding. Unknown peers are not trusted merely because they claim to run the same circuit. [R24–R25]

5. Proposed component specifications

A reviewable work plan separates: (a) problem statement and requirements; (b) an authority and instance data model; (c) the cryptographic admission relation; (d) native authenticated key establishment; (e) canonical record encoding and key schedule; (f) HTTP/application-semantic binding; (g) enrollment, revocation and recovery; and (h) browser integration. Components can use existing standards when their semantics fit and define a narrow new profile where the interoperability need is demonstrated.

The receiving community should determine which components fit WIMSE, RATS, HTTPBIS, TLS, or another venue. Web Bot Auth is relevant only to the human-facing website use case within its charter. CFRG can provide a research venue for the cryptographic construction. WICG can incubate browser-facing features separately. The architecture does not assume that a single working group owns all these responsibilities. [R17–R23]

6. Proposed lifecycle and invariants

A client verifies its locally installed code and trust roots, parses the intended origin, authenticates the service descriptor, and verifies a fresh server contribution. It generates a distinct instance key and proposes its permitted scope. The human authorizes the context through the Human Admission Client and the private-reference admission service (Proof Authority). The completed handshake binds both identities, the human receipt, policies, algorithm choices, ephemeral contributions, and protocol profile. Application records begin only after explicit peer key confirmation.

Children enter through an independently enforced mint. The mint checks live ancestry, an authorized spawning operation, approved template, appraised runtime, requested policy, lifetimes, and shared resource budget. It registers a new instance and binds the child key to its grant. A leaf without spawning authority remains unable to mint an authorized child.

Effective permissions are evaluated over complete normalized requests:

$$\mathcal{A}_v = \mathcal{A}_u \cap \mathcal{A}(P_v), \qquad q \in \mathcal{A}_v \Rightarrow q \in \mathcal{A}_0. \tag{P1}$$

Here u is the parent, v the child, and $\mathcal{A}_0$ the original root ceiling. The implication follows by induction over authenticated parent links. Resource owners also apply their independent service-side policies. Dynamic posture and revocation may narrow exercise of the committed ceiling; broadening it requires an authorized new admission.

7. Record contract

This is a semantic contract, not a frozen wire format. The eventual specification must select exact data types, canonical bytes, cryptographic suites, size bounds, critical-extension behavior, and test vectors.

```
record_contract: proposed_not_wire_frozen
public_authenticated_header:
  version: selected-profile-version
  channel_epoch: admitted-epoch
  direction: client-to-server-or-server-to-client
  stream_id: admitted-stream
  sequence: unused-directional-position
  record_kind: request-response-chunk-ack-rekey-close
  ciphertext_length: bounded-by-profile
encrypted_content:
  sender_instance: admitted-instance-reference
  peer_identity: authenticated-recipient
  root_receipt_ref: approved-origin-of-authority
  lineage_ref: authenticated-delegation-chain
  policy_digest: admitted-policy-commitment
  operation_id: business-operation-identifier
  request_reference: present-for-response
  method_or_operation: canonical-operation
  target_resource: canonical-resource
  arguments_and_payload: canonical-protected-bytes
  purpose_and_classification: trusted-policy-inputs
  predecessor_reference: profile-defined-stream-state
protection:
  authenticated_encryption: selected-reviewed-AEAD-profile
  authorization_proof: bound-to-header-ciphertext-and-channel-state
```

Low-entropy payload hashes and sensitive lineage metadata are not exposed as unauthenticated or public confirmation handles by default. The profile inventories what remains observable, including sizes, endpoints, and timing. The enrollment commitment is not serialized into the record contract.

Conceptually, the record authenticator binds the public framing, ciphertext, and current authorization context:

$$\pi_j = \mathsf{MAC}_{k^{\mathrm{auth}}_{d,e,s}}\left(\mathsf{Enc}(\texttt{record}, H_j, h(C_j), j)\right). \tag{P2}$$

H_j is the canonical header; C_j is the authenticated ciphertext; the key is separated by direction d, epoch e, and stream s. This expression specifies binding requirements, not a complete security proof or a final selection of MAC and AEAD composition. Key generation, nonce uniqueness, traffic limits, erasure, and key-confirmation details belong to the reviewed cryptographic profile. Implementations reject mismatched direction, sender generation, policy, or retired state before exposing content.

8. Acceptance and execution

```
RECEIVE(record, channel):
    parse bounded canonical framing; reject unknown critical features
    verify admitted profile, peer, direction, epoch, and stream
    authenticate and decrypt under the selected record suite
    verify instance possession, complete lineage, and current admission
    normalize the exact application operation once
    evaluate every ancestor ceiling and the resource owner's policy
    atomically reserve applicable budgets and consume the action position
    execute or reconcile the same business operation
    record the durable outcome; emit a separately protected response
```

The response binds the exact request and resulting content before rendering or consumption. A lost response is recovered through a new protected record referring to the same operation. An authenticated ACK establishes the record-level event defined by the profile, not necessarily completed business execution. Independent streams avoid requiring a globally serialized history across unrelated clients or assets.

Unauthorized record rejection and useful authorized behavior must both be demonstrated. Tests inspect the actual effect at the resource, including alternate routes that might bypass the requesting frontend. A broker cannot replace the child's narrow grant with its own broader infrastructure credentials.

9. Native secure-channel requirements

The final N1 specification must define transcript-bound peer authentication, independently authenticated service naming, approved key-establishment choices, fresh secret contributions, explicit key confirmation, direction separation, unique AEAD nonces, downgrade resistance, replay state, bounded parsing, key update, failure behavior, and recovery. It must state forward-secrecy and post-compromise-recovery properties separately and justify each through its key schedule and threat model.

A timestamp indexes state and contributes no presumed secret entropy. An admission success bit gates key use and is not an encryption key. A ratchet with no new secret input does not automatically recover from compromise. Version negotiation and fallback choices are authenticated. The initial profile should reject state-changing early-data effects until a reviewed anti-replay and intent-binding design explicitly permits them.

Trust-anchor enrollment, service key rotation, expiration, withdrawal, and disaster replacement are part of the protocol's operational security. A signed descriptor from an untrusted issuer cannot bootstrap its own trust. Cross-organization trust requires a defined governance and verification path rather than treating a local enterprise registry as a universal Internet authority.

10. HTTP, browser, and infrastructure integration

The application binding must specify how methods, target URIs, status codes, content metadata, redirects, errors, caching, streaming, retries, and cancellation retain their meaning. Claiming to be a next HTTP version requires more than assigning a higher number. HTTP's common semantics and version framework provide the compatibility reference; new version work requires the appropriate agreed scope. [R17; R26]

A native client also needs an origin model, trusted address presentation, cross-origin rules, content-execution isolation, downloads, service-worker treatment, and secure-context policy. An installed extension is not assumed to replace a general browser's transport implementation. The first demonstration can use a controlled renderer and explicitly brokered resource fetching. General browser integration is separately evaluated with browser implementers and relevant web-platform venues. [P1, §10; R23]

Reverse proxies, load balancers, and caches have explicit roles. A TLS-terminating gateway in W1 does not automatically become the ultimate application authorizer. An edge intermediary receives a distinct constrained grant when it needs to act on protected content. Cached objects have content/provenance semantics distinct from live operation authority; private decrypted responses are not shared across principals by accident. Federation and external services preserve the original delegated scope.

11. Security and privacy analysis agenda

The attacker model includes passive recording, active relay, substituted origins, forged descriptors, malicious children, copied binaries, runtime compromise at specified boundaries, stale snapshots, malicious input, and state-store failures. Authentication, permission, confidentiality, metadata exposure, denial of service, and audit privacy are separate evaluation objectives.

The binary-conditioned privacy proposition in the supporting paper concerns additional information about the human commitment, given the complete acquired artifact and prior knowledge. It is conditional on the exact observation distributions. A public protocol claim requires the actual production construction and its composition with headers, outcomes, timing classes, active interactions, and persistent state. The reference independent-mask model does not itself certify an arbitrary compiled binary. [P2, Appendix B]

The initial acceptance campaign covers replay races, wrong-direction proofs, forged responses, request substitution, unsafe redirects, modified policies, removed ancestors, unauthorized minting, budget races, copied instances, parent revocation, rollback, and first-contact key substitution. Independent implementations need byte-identical positive and negative vectors plus a measured interoperable exchange before a production conformance claim.

12. IANA, deployment, and research status

This candidate requests no assignment. HTTP6 is a project label. The private source profile identifiers are not represented as registered ALPN values, URI schemes, ports, or HTTP versions. Allocation requests would identify each actual registry and its applicable process in a later accepted specification.

The supporting packet contains architecture, schemas, local reference code, and bounded test records. A complete native handshake, production private-reference engine assessment, distributed state qualification, protected-platform evidence, and independently interoperable endpoints remain explicit deliverables. The initial IETF request is discussion and routing of this work, while a parallel NIST demonstration can examine the application-authority model over existing standards.

13. Questions for the receiving community

The author seeks feedback on the unmet interoperability problem, which portions can be expressed as existing-standard profiles, how workload delegation should be represented, where service enrollment belongs, which authenticated key-establishment approach is suitable for the native research target, and which venue should own each proposed component. Particular attention is requested for the boundary between public service identity, human admission, and runtime-specific delegated authority.

Supporting documents

[P1] *HTTP6: Proof-Native Web Protocol, Process-Bound Oscillation and Human-Authorized Fleet Recovery*, November 8, 2026. Existing architecture and engineering specification.

[P2] Frank Dylan Rosario, *HTTP6: Human-Originated Authority for NIST-Aligned Cybersecurity*, Expanded Public Review Edition 2.0, November 9, 2026. Private-reference analysis, hereditary-authority model, schemas, pseudocode, reference code, and evidence specification.

References

[R15] IETF: Bringing new work to the IETF. Official source. Individual Internet-Draft, community support, venue selection, and public contribution responsibilities.

[R16] IETF DISPATCH current charter. Official source. Active standing venue for ART, SEC and non-transport WIT new-work routing; dispatch@ietf.org.

[R17] IETF HTTP Working Group current charter. Official source. New HTTP versions are outside current charter without rechartering.

[R18] IETF TLS Working Group. Official source. Secure-channel review and TLS-related work, subject to scope.

[R19] IETF WIMSE Working Group. Official source. Runtime workload identity propagation, representation, and processing; personal identities and deployment chains excluded from its stated scope.

[R20] IETF RATS Working Group. Official source. Remote-attestation component review.

[R21] IETF Web Bot Auth charter. Official source. Bot access to human-facing websites; user authentication and agent-to-agent/HTTP API use cases excluded.

[R22] IRTF Crypto Forum Research Group. Official source. Cryptographic research discussion; cfrg@ietf.org.

[R23] W3C Web Platform Incubator Community Group. Official source. Web-feature incubation, not transport standard adoption or W3C endorsement.

[R24] RFC 9846: The Transport Layer Security Protocol Version 1.3. Official source. July 2026 TLS 1.3 revision; supersedes RFC 8446. Includes authenticated records, PSK and public-key authentication.

[R25] RFC 7250: Using Raw Public Keys in TLS and DTLS. Official source. Standardized raw-public-key alternative; certificate format and trust bootstrap are different decisions.

[R26] RFC 9110: HTTP Semantics. Official source. HTTP semantics and http/https scheme meaning.

[R27] RFC 9001: Using TLS to Secure QUIC. Official source. Standard QUIC security uses TLS; a new security handshake needs a distinct specified binding.

[R28] IETF: Intellectual property rights and the IETF. Official source. Contribution rights and patent-disclosure responsibilities; owner review before public contribution.

[R29] IETF Internet-Drafts. Official source. An individual draft is a work in progress; submission is not working-group adoption.

[R31] RFC 9449: OAuth 2.0 Demonstrating Proof of Possession (DPoP). Official source. Relevant sender-constrained request-authentication comparator.