Independent Submission
Internet-Draft
Intended status: Experimental
Expires: 25 November 2026

E. Noss
Groundmark
M. Jeftovic
easyDNS Technologies Inc.

24 May 2026

Groundmark Attestation Framework and Identity Service Provider Profile
draft-noss-jeftovic-groundmark-attestation-00

Abstract

This document defines the Groundmark attestation framework and the profile of Identity Service Providers (IDSPs) that issue Groundmark attestations. It is the companion document to the Groundmark core protocol, which specifies DNS publication and request-signature mechanics. This document specifies the role and obligations of Identity Service Providers, a four-level taxonomy of attestation strength, an initial vocabulary of claim types, the JSON schema and signature construction for attestation payloads, the method-disclosure discipline that defines an IDSP’s accountability obligation, and revocation semantics for the attestation layer.

The core protocol document defines the discovery mechanisms (the _agentclaim DNS TXT record and the HTTP retrieval of attestation payloads) on which this document depends. Together the two documents specify Groundmark.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 25 November 2026.

Table of Contents

1. Introduction

The Groundmark core protocol [I-D.noss-jeftovic-groundmark-core] specifies how an agent's signing public key and references to externally hosted attestations are published in DNS, and how relying parties verify HTTP request signatures using the published key. The core protocol intentionally treats attestation content as opaque: DNS publishes a reference, the relying party retrieves it over HTTPS, and the question of what an attestation says and how it should be evaluated is deferred to this document.

This document specifies that evaluation framework. It defines:

* The role of an Identity Service Provider (IDSP), an entity that verifies a claim about an operator or agent and publishes a signed attestation under the Groundmark framework.

* A four-level taxonomy of attestation strength, from cryptographic-only Level 0 through regulatory-grade Level 3.

* An initial vocabulary of claim types, with structured value schemas and registration policy for extensions.

* The JSON schema and canonical signature construction for an attestation payload served at a "ref=" URL.

* The method-disclosure obligation: an IDSP's primary duty is to accurately characterize the verification it performed, not to assert a conclusion. This obligation is what makes Groundmark attestations evaluable rather than reducible to brand recognition.

* Revocation semantics specific to the attestation layer, including fail-mode behaviour for unreachable revocation endpoints.

The design is informed by Kim Cameron's Laws of Identity [Cameron2005], reframed for a setting in which the principal whose identity is being attested is an organizational operator (or, in the individual-operator case, a hosted-identity provider) rather than a human individual interacting directly.

1.1. Relationship to the Core Protocol

This document depends on the core protocol [I-D.noss-jeftovic-groundmark-core] for:

* DNS record format and publication for _agentclaim records.

* DNSSEC requirements for all Groundmark DNS lookups.

* The HTTPS retrieval transport for attestation payloads.

* Request-signature mechanics, against which an attestation is ultimately evaluated.

A relying party implementing Groundmark MUST implement the core protocol. Implementing this document additionally is OPTIONAL: a relying party may choose to require only signature verification (Level 0) without evaluating attestations, in which case the mechanisms in this document are not exercised. The core protocol's permissionless default mode remains available to all participants.

1.2. Design Principles

The framework defined here is shaped by three principles, in addition to those stated in the core protocol.

* Authenticate the minimum necessary. The right attestation level for any transaction is the lowest that meets the relying party's legitimate risk profile. The framework is designed so that most transactions can proceed at Level 0; higher levels are available but not encouraged.

* IDSPs attest to methods, not conclusions. An IDSP's primary accountability obligation is to characterize what verification it performed and to stake its reputation on the accuracy of that characterization. The trust decision belongs to the relying party. This is what allows a relying party to make a contextual judgment rather than a binary accept/reject decision based on the IDSP's brand alone.

* No single point of control. The framework permits a diversity of IDSPs operating under different trust frameworks, jurisdictions, and competence areas. No central authority designates which IDSPs are authoritative. Recognition of an IDSP for any specific purpose is a property of the relying party's policy and, where applicable, the relying party's regulatory framework.

1.3. Non-Goals

This document does not:

* Designate, certify, or otherwise centrally validate IDSPs. The framework specifies what an IDSP does and the obligations it accepts; recognition is a matter of relying-party policy or external trust framework.

* Define the substantive content of any specific regulatory framework. Level 3 attestations may exist within frameworks (for example, FINTRAC PCMLTFA for Canadian KYC, or applicable professional licensing regimes); this document specifies how a Level 3 attestation is structured and verified, not the underlying regulatory standard.

* Replace any existing credential or attestation system. An IDSP MAY use Verifiable Credentials, JWTs, or other formats internally; the Groundmark attestation payload is the format in which an attestation is presented to a Groundmark relying party.

1.4. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Terminology

The terminology of the core protocol [I-D.noss-jeftovic-groundmark-core] applies. The following additional terms are used in this document.

Attestation A signed statement by an IDSP that a specific claim about an operator or agent has been verified to a stated standard using stated methods. Serialized as a JSON document per Section 6.

Claim A bounded predicate about an operator or agent that an attestation verifies (for example, "the operator has passed FINTRAC tier-basic KYC", or "the operator holds an active real-estate licence in Ontario, Canada").

Claim Type A registered identifier denoting the predicate structure of a claim. The initial vocabulary is defined in Section 5.

Identity Service Provider (IDSP) A trusted third party that performs verification of a claim and publishes a signed attestation. Defined more fully in Section 3.

Level An integer from 0 to 3 indicating the strength of verification associated with a Groundmark attestation, as defined in Section 4.

Method Disclosure The structured description, present in every Groundmark attestation, of the verification methods the IDSP engaged. Defined in Section 3.1.

Relying Party As in the core protocol. In this document’s context, a relying party evaluating attestations decides which IDSPs it trusts and for what purposes, and which attestation levels and claim types it requires.

3. The Identity Service Provider Role

An IDSP under Groundmark is an entity that:

1. Performs verification of a specific claim about an operator or agent, using methods the IDSP discloses in each attestation it issues.

2. Publishes attestations as signed JSON documents over HTTPS, at the URLs referenced by _agentclaim "ref=" fields.

3. Publishes its own signing public key in DNS using the _agentid convention defined by the core protocol, under a DNSSEC-signed zone.

4. Operates a revocation mechanism per Section 8.

Any person, organization, or institution with appropriate competence may operate as an IDSP for claims within its competence. No central designation is required. Recognition of an IDSP by a relying party, or by a regulatory framework whose requirements a relying party is attempting to satisfy, is independent of this specification.

3.1. Method Disclosure

An IDSP's primary obligation under Groundmark is method disclosure: each attestation MUST contain a structured description of the verification methods the IDSP performed. This obligation is what distinguishes an IDSP from a conventional certificate authority. A certificate authority asserts a conclusion ("this entity controls this domain"); a Groundmark IDSP asserts a conclusion together with a disclosure of how that conclusion was reached, allowing the relying party to make a contextual trust decision.

The structure of the method_disclosure object is specified in Section 6.3. The IDSP is responsible for the accuracy of this disclosure; misrepresentation of methods is the principal failure mode against which the IDSP's reputation is exposed.

3.2. IDSP Trust Anchors

This document does not specify how a relying party comes to trust a specific IDSP for a specific purpose. The expected mechanisms include:

* Direct evaluation by the relying party of an IDSP's public practices statement, audit reports, and method disclosures.

* Recognition by a regulator or industry body whose rules the relying party operates under.

* Inclusion in a curated list maintained by a community or trust framework operator.

These are out of scope for this specification. What this specification provides is the substrate that makes any of these evaluation modes possible: every Groundmark attestation includes the IDSP's identity, the methods it performed, and the cryptographic material needed to verify the IDSP's signature.

4. Attestation Level Taxonomy

Groundmark defines four attestation levels indicating the strength of verification associated with a claim. The level is a floor, not a ceiling: a relying party requiring Level 1 MAY accept a Level 2 or Level 3 attestation for the same claim type. A relying party requiring Level 2 MUST NOT accept a Level 1 attestation in its place.

The levels do not aggregate. Three Level 2 attestations do not constitute a Level 3 attestation. Each claim stands on its own merits.

4.1. Level 0 - Permissionless

No attestation required. The relying party verifies an HTTP request signature against the agent's DNS-published key per the core protocol; domain ownership establishes the accountability chain through DNS and (if DNSSEC-validated) through the registrar-verified registrant.

This is the default mode under Groundmark and is expected to cover the majority of agent transactions. Self-asserted _agentclaim records using the "val=" form (for example, operator-contact) are Level 0 disclosures: they are operator assertions, not third-party verifications.

A relying party operating at Level 0 CAN conclude that the request originated from a party that controls the agent domain. It CANNOT conclude anything about the operator's identity, legal standing, location, or compliance.

4.2. Level 1 - Operator Accountability

A real, identifiable party stands behind the agent. Level 1 establishes that an operator exists, that the operator demonstrably controls the agent domain, and that the operator has accepted accountability under the IDSP's trust framework. The claim is about existence and accountability, not capability or compliance: if something goes wrong, there is a known party to contact.

Required verification, performed by the IDSP:

* Operator existence: the operator is a real person or organization.

* Operational control: the operator demonstrably controls the agent domain.

* Acceptance of terms: the operator has agreed to the IDSP's accountability framework.

Acceptable methods are at the IDSP's discretion subject to the method-disclosure obligation. Typical methods include business registration lookup, confirmation messaging to an organizational domain, or credential binding to a verified account.

A relying party operating at Level 1 CAN conclude that an identifiable operator is accountable for the agent's actions, and that the operator accepted terms establishing accountability. It CANNOT conclude that the operator is compliant with any specific regulation or legal standard.

4.3. Level 2 - Claim-Specific Verification

A bounded, specific fact about the operator or agent has been attested. Each Level 2 attestation is a single predicate: "the operator is over 18", "the operator holds a valid Ontario real estate licence", "the agent has passed basic financial-services onboarding". The minimal-disclosure principle is operationally critical at this level: the IDSP attests to the predicate, not to the underlying data.

A relying party operating at Level 2 CAN conclude that the stated predicate was verified by a third party to the standard disclosed, and that the IDSP staked its reputation on the accuracy of the attestation. It CANNOT conclude that the claim satisfies a regulatory requirement without also verifying that the IDSP and its methods are recognised by the relevant regulator.

4.4. Level 3 - Regulatory-Grade

The attestation meets a specific legal or regulatory standard. Level 3 covers claims that must meet defined legal or regulatory requirements such as Know Your Customer / Anti-Money Laundering verification under a named framework, professional licensure verified against government records, or biometric identity proofing under a named assurance level.

A Level 3 attestation MUST carry the "framework" field (Section 6.2) identifying the regulatory or trust framework under which the attestation is issued. It MUST carry the "jurisdiction" field where the framework is jurisdictionally scoped.

A relying party operating at Level 3 CAN rely on the attestation to satisfy its own regulatory obligations where the relevant trust framework explicitly permits such reliance. The relying party MUST verify that the named IDSP is recognized by the relevant regulator or

framework before relying on a Level 3 attestation for compliance purposes; this specification does not designate any IDSP as authoritative for any framework.

4.5. Level Stacking and Composability

An agent identity MAY carry multiple _agentclaim records simultaneously, each independent. A single agent might present a Level 1 operator-verified attestation alongside several Level 2 claim-specific attestations from different IDSPs. The relying party selects only the claims its policy requires.

Multiple attestations from different IDSPs do not implicitly refer to the same underlying operator identity, since IDSPs do not coordinate with one another. Section 9.3 addresses this limitation.

5. Claim Type Vocabulary

This section defines the initial vocabulary of claim types. Each claim type specifies the JSON structure of the claim_value field in an attestation payload (Section 6).

Custom claim types MUST use the form "x-<registrant-domain>-<claim-name>", where <registrant-domain> is a domain controlled by the entity defining the claim type. Custom claim types do not require IANA registration; relying parties MUST ignore claim types they do not recognize, per the core protocol.

Claim types intended for broad interoperability SHOULD be registered in the IANA Groundmark Claim Types registry established by this document (Section 12).

5.1. operator-verified

Level: 1.

The IDSP has verified that a real, identifiable operator controls the agent and has accepted the IDSP’s accountability terms.

claim_value:

```
{
  "verified": true,
  "operator_display_name": "Acme Corp"
}
```

"verified" is REQUIRED and MUST be true (an attestation with verified=false MUST NOT be issued; such a record should be absent, or, if previously present, revoked).

"operator_display_name" is OPTIONAL and SHOULD only be included if the operator has explicitly chosen to disclose it publicly. Relying parties MUST NOT use operator_display_name as the sole basis of any trust or access-control decision. See Section 9.2.

5.2. age-over-18, age-over-21

Level: 2.

The operator meets the specified age threshold. The claim value is a boolean predicate; the operator's actual age or date of birth is not disclosed.

claim_value:

```
{ "result": true }
```

5.3. jurisdiction-licensed

Level: 2 or 3, depending on the IDSP's method and the framework under which it is operating.

The operator holds a valid licence in a named jurisdiction and practice category. The licence number MUST NOT be included in the claim value.

claim_value:

```
{
  "licensed": true,
  "jurisdiction": "CA-ON",
  "licence_category": "real-estate-agent",
  "licence_status": "active"
}
```

"jurisdiction" uses the conventional country-subdivision form (ISO 3166-1 alpha-2, optionally followed by "-" and an ISO 3166-2 subdivision code).

When this claim is published as a Level 3 attestation, the top-level "framework" field of the attestation MUST identify the regulatory licensing framework.

5.4. kyc-basic

Level: 2 or 3.

The operator has passed a basic Know Your Customer screening.

claim_value:

```
{
  "passed": true,
  "framework": "FINTRAC-PCMLTFA",
  "tier": "basic"
}
```

The "framework" field inside claim_value is REQUIRED for this claim type and identifies the screening framework applied. When the attestation is Level 3, the top-level "framework" field MUST also be present and SHOULD match the claim_value "framework".

5.5. kyc-enhanced

Level: 3.

The operator has passed an enhanced KYC process, including source-of-funds verification or equivalent under the named framework.

claim_value:

```
{
  "passed": true,
  "framework": "FINTRAC-PCMLTFA",
  "tier": "enhanced"
}
```

5.6. operator-contact

Level: 0 (self-asserted).

A self-declared contact point. This claim type MAY be published directly in a _agentclaim record using "val=" without a "ref=" URL or "idsp=", per the core protocol. Relying parties MUST treat it as unverified self-assertion.

claim_value (when retrieved as an IDSP-issued attestation, which is uncommon for this claim type):

```
{
  "contact_type": "email",
  "value": "ops@acmecorp.example"
}
```

6. Attestation Payload Schema

This section defines the JSON document served by an IDSP at the URL referenced by an _agentclaim "ref=" field.

6.1. Transport and Caching

The transport for attestation retrieval is HTTPS, per the core protocol. Additional requirements specific to the attestation layer:

* IDSPs MUST serve attestation payloads with Content-Type: application/json (or application/jose+json for Verifiable Credential or JWT-bearing variants under future extensions).

* IDSPs MUST require no authentication from the relying party. The "ref=" URL functions as an opaque bearer reference.

* IDSPs MUST validate TLS at the transport layer per the underlying HTTPS implementation; relying parties MUST validate the attestation endpoint’s TLS certificate per [RFC9110].

* IDSPs MUST set HTTP Cache-Control directives [RFC9111] consistent with the attestation’s expires_at time. IDSPs MUST NOT set max-age values so short as to force effectively per-request revalidation; the framework is designed for cached retrieval, and IDSPs that operate the revocation endpoint as a per-transaction authentication gate create a denial-of-service vector against the relying-party population (see Section 9.5).

* Relying parties MAY cache attestation payloads up to the lesser of the IDSP-supplied max-age and the attestation’s expires_at time.

6.2. Top-Level Fields

The attestation is a flat JSON [RFC8259] object using snake_case field names. The following fields are defined.

schema_version (REQUIRED, string) Schema version. The current value is "1.0". Relying parties SHOULD reject schema versions they do not understand.

subject (REQUIRED, string) The agent domain this attestation covers, as a fully qualified domain name in lowercase ASCII.

claim_type (REQUIRED, string) A value from the claim type vocabulary in Section 5, a vendor extension of the form "x-<registrant-domain>-<claim-name>", or an IANA-registered extension.

claim_value (REQUIRED, object) The minimal predicate value. Structure is determined by claim_type.

level (REQUIRED, integer) The attestation level (0 through 3). Relying parties MUST reject attestations whose level is below the level required by the relying party's policy.

idsp (REQUIRED, string) The IDSP's domain name. The IDSP's public signing key MUST be retrievable via the core-protocol _agentid convention at _agentid.<idsp>.

issued_at (REQUIRED, string) The time at which the attestation was issued, formatted per [RFC3339].

expires_at (REQUIRED, string) The time at which the attestation expires, formatted per [RFC3339]. Relying parties MUST reject attestations whose expires_at is in the past.

method_disclosure (REQUIRED, object) Structured description of the verification the IDSP performed. See Section 6.3.

endpoint_url (REQUIRED, string) The canonical HTTPS URL at which this attestation is served. MUST match the "ref=" URL in the corresponding _agentclaim record. Bound into the signed payload to prevent portability attacks (see Section 9.1).

revocation_endpoint (REQUIRED, string) HTTPS URL for revocation status. Returns { "revoked": true | false }. See Section 8.

signing_key_id (REQUIRED, string) Identifier of the IDSP signing key used. Matches the "kid=" value of the corresponding _agentid record at the IDSP's domain.

signature (REQUIRED, string) Base64url-encoded [RFC4648] Ed25519 signature over the canonical payload. See Section 7.

jurisdiction (OPTIONAL, string) Applicable jurisdiction. REQUIRED at Level 3 when the framework is jurisdictionally scoped.

framework (OPTIONAL, string) Named regulatory or trust framework. REQUIRED at Level 3.

short_lived_token (OPTIONAL, object) A short-lived signed status token analogous to certificate stapling. See Section 8.3.

6.3. The method_disclosure Object

methods (REQUIRED, array of strings) An array of method identifiers, each drawn from the Groundmark Method Identifier registry established by this document (Section 12). Implementations MUST tolerate identifiers they do not recognize.

method_detail (OPTIONAL, string) Free-text description providing additional context. Intended for human review, not machine parsing.

subcontractors (OPTIONAL, array of strings) Domain names of any third-party services engaged in the verification. Present when the IDSP subcontracted part of the verification.

evidence_class (OPTIONAL, array of strings) An array of evidence-class identifiers from the Groundmark Method Identifier registry, indicating the categories of evidence the IDSP examined (for example, "government-issued-id", "corporate-registry", "biometric-liveness"). The initial set is defined in the registry; community-defined extensions follow the registry’s policy.

6.4. Worked Example

The following non-normative example illustrates the JSON payload served by an IDSP at the URL referenced by an _agentclaim "ref=" field. It is the Level 1 operator-verified attestation referenced by the worked example in the core protocol.

```
{
  "schema_version": "1.0",
  "subject": "purchasing.acmecorp.example",
  "claim_type": "operator-verified",
  "claim_value": {
    "verified": true,
    "operator_display_name": "Acme Corp"
  },
  "level": 1,
  "idsp": "verify.example.net",
  "issued_at": "2026-05-19T12:00:00Z",
  "expires_at": "2026-12-01T00:00:00Z",
  "method_disclosure": {
    "methods": [
      "corporate-registry-lookup",
      "organizational-domain-email-confirmation"
    ],
    "method_detail":
      "Operator legal entity confirmed against the Ontario
       Business Registry; domain control confirmed by email
       challenge to postmaster@acmecorp.example.",
    "evidence_class": [
      "corporate-registry",
      "domain-control"
    ]
  },
  "endpoint_url":
    "https://verify.example.net/a/4f9f3e1c7ab93d4f8d22",
  "revocation_endpoint":
    "https://verify.example.net/r/4f9f3e1c7ab93d4f8d22",
  "signing_key_id": "k1",
  "signature": "MEUCIQ...base64url...=="
}
```


The IDSP’s signing public key is published at _agentid.verify.example.net. The relying party retrieves it via a DNSSEC-validating resolver, selects the record with kid=k1, reconstructs the canonical payload using [RFC8785] on the object with the signature field removed, and verifies the Ed25519 signature against the IDSP’s public key.

7. Signature Construction and Verification

Attestation signatures use Ed25519 [RFC8032]. The signature is computed over a canonical serialization of the attestation payload with the signature field removed.

The canonical serialization is JSON Canonicalization Scheme [RFC8785] applied to the attestation payload object, omitting the signature key. JCS provides a deterministic byte sequence regardless of input ordering or whitespace, removing a class of canonicalization bugs.

The IDSP's public key is published in DNS using the _agentid.<idsp-domain> convention defined by the core protocol. IDSPs MUST sign their zones with DNSSEC. A public key retrieved from a non-DNSSEC-signed zone MUST NOT be used for attestation verification. The signing_key_id field in the attestation identifies which of the IDSP's keys was used; this corresponds to the "kid=" field in the IDSP's _agentid record.

7.1. Verification Steps

A relying party verifying an attestation payload MUST:

1. Confirm that schema_version is understood.

2. Confirm that expires_at is in the future.

3. Confirm that endpoint_url matches the "ref=" URL from which the attestation was retrieved. Reject otherwise (see Section 9.1).

4. Confirm that level meets or exceeds the relying party's policy floor.

5. Retrieve the IDSP's _agentid record(s) from DNS using a DNSSEC-validating resolver. Select the record matching signing_key_id. Reject if no such record exists, if DNSSEC validation fails, or if the IDSP zone is unsigned.

6. Compute the canonical serialization per [RFC8785] on the payload with signature removed.

7. Verify the Ed25519 signature using the IDSP's public key.

8. Check the revocation_endpoint according to the rules in Section 8.

8. Revocation

Two independent revocation mechanisms operate at the attestation layer, in addition to DNS-level revocation provided by the core protocol (deletion of the _agentclaim record).

DNS-layer revocation (under the core protocol) Removal of the _agentclaim record invalidates the claim within the record's TTL.

Endpoint-layer revocation The IDSP marks a specific attestation as revoked at its revocation_endpoint. The agent identity remains usable; only the specific claim is revoked. This is appropriate when an operator's standing changes with respect to a single claim without affecting the underlying identity.

The revocation endpoint response is structurally minimal:

```
{ "revoked": true }
```

or

```
{ "revoked": false }
```

IDSPs MAY include additional fields (revoked_at, reason) but MUST include the boolean revoked field as defined. Relying parties MUST NOT require additional fields.

The revocation endpoint MUST NOT require authentication from the relying party. IDSPs MUST NOT use the revocation endpoint as a per-transaction authentication gate (see Section 9.5).

8.1. Fail-Mode Behaviour

When the revocation endpoint is unreachable, relying parties MUST apply the following fail behaviour, scaled by attestation level:

Level 3 Relying parties MUST fail closed. An unreachable revocation endpoint MUST be treated as a verification failure and the transaction declined. Continuing under Level 3 without a current revocation check is a non-conformant deployment.

Level 2 Relying parties MUST fail closed UNLESS all of the following hold: the attestation was issued less than 24 hours before the current time, the relying party's policy permits soft-fail at Level 2 under this transaction's risk profile, and the relying party logs the soft-fail per Section 8.2.

Level 1 Relying parties MAY fail open subject to the logging requirements of Section 8.2. Fail-open is permitted at Level 1 because the underlying claim is operator-accountability rather than compliance: the worst-case consequence of an erroneously accepted Level 1 attestation is materially different from the worst-case consequence at higher levels.

8.2. Audit Logging of Fail-Open Decisions

Fail-open behaviour at any level MUST be logged. The logging requirements below are the minimum; relying parties may impose additional requirements under their own policies.

The fail-open log entry MUST include:

* The agent domain and the attestation's endpoint_url.
* The idsp, claim_type, and level from the attestation.
* The reason for fail-open (typically: revocation endpoint unreachable, with an indication of the failure mode).
* The time of the decision.
* The identity of the transaction or operation that proceeded under fail-open.

The log MUST be retained by the relying party for a period not less than one year, or for the period required by any applicable regulatory framework, whichever is longer. The log SHOULD be available to the IDSP on reasonable request, since an unexpected pattern of fail-open events on a particular IDSP is operationally significant to both parties.

8.3. Short-Lived Status Tokens

Real-time revocation checks against an IDSP endpoint introduce the same class of operational fragility as OCSP. To mitigate this without reintroducing OCSP's worst failure modes, an IDSP MAY include a short_lived_token field in the attestation payload, analogous to certificate stapling.

A short-lived status token is a signed assertion of the form:

```
{
  "status": "active",
  "as_of": "2026-05-19T12:00:00Z",
  "valid_until": "2026-05-19T18:00:00Z",
  "signature": "<base64url-ed25519>"
}
```

The token is signed by the same IDSP signing key as the parent attestation, over the canonical serialization of the token object with signature removed.

When a current short-lived status token is present and valid, the relying party MAY treat it as a substitute for a fresh revocation_endpoint check during the token's validity window. The relying party MUST NOT extend the token's effective validity past valid_until. Token validity windows SHOULD be no longer than 24 hours; IDSPs that issue longer-lived tokens defeat the purpose of the mechanism.

IDSPs MAY rotate tokens by re-serving the attestation payload with an updated short_lived_token; the rest of the payload (and its signature) is unchanged. Relying parties caching the attestation SHOULD periodically re-fetch to obtain refreshed tokens; this is substantially cheaper than full revocation checks.

9. Security Considerations

9.1. Endpoint Portability Attacks

The endpoint_url field in the canonical payload binds an attestation to the specific URL at which it is served. Without this binding, an attacker who obtained a valid attestation payload (perhaps by previously serving as the IDSP under a since-decommissioned identity) could serve that payload at a different location and have it accepted by relying parties. Including endpoint_url in the signed canonical payload prevents this attack. Relying parties MUST verify the endpoint_url match per Section 7.1; this is not optional.

9.2. Display Name Impersonation

The operator_display_name field in operator-verified attestations is operator-chosen and self-asserted. The IDSP attesting that the operator exists and is accountable does not, by default, verify that the chosen display name is accurate or non-deceptive.

Relying parties MUST NOT use operator_display_name as the basis of trust or access-control decisions. The field is a UI affordance only.

Future versions of this specification may require IDSPs to validate display names against protected brand-name lists as a condition of Level 1 attestation. Such a requirement is out of scope for this document, but the field's presence in the schema provides a hook for such future requirements.

9.3. Cross-Claim Consistency

A single agent may carry attestations from multiple IDSPs that implicitly refer to different underlying operator identities, since IDSPs do not coordinate with one another. A relying party evaluating two claims from two different IDSPs cannot, in general, determine whether both claims describe the same operator.

For high-assurance transactions requiring multiple claims, relying parties SHOULD prefer attestations from a single IDSP where possible. Future versions of this specification may define a subject_handle field that allows IDSPs to identify a stable operator handle within a single trust framework, enabling cross-claim consistency checks. This is out of scope for this document.

9.4. IDSP Compromise

Compromise of an IDSP's signing key permits an attacker to issue fraudulent attestations until the corresponding _agentid record at the IDSP's domain is removed and caches expire. IDSPs MUST treat their signing keys with the same operational discipline as certificate-authority signing keys, including hardware key storage and audit-grade access controls. An IDSP SHOULD prepare for emergency key rotation per the procedure defined in the core protocol, and SHOULD pre-announce key rotation policy in its public practices statement.

A compromised IDSP signing key invalidates all attestations signed under that key. There is no fine-grained recovery: relying parties detecting an IDSP key compromise SHOULD treat all attestations under that key as unverifiable until the key is rotated and the affected attestations are re-issued.

9.5. Denial of Service via Revocation Endpoint

The revocation endpoint is a relying-party-facing service consulted on a substantial fraction of attestations. IDSPs that operate it as an authenticated, throttled, or per-relying-party-rate-limited service shift operational fragility onto the entire relying-party population. This document REQUIRES the revocation endpoint to be unauthenticated, structurally minimal, and operated at a service quality appropriate to its function in the trust framework. IDSPs that do not meet this requirement create a denial-of-service vector against deployments depending on their attestations.

9.6. Method Disclosure Misrepresentation

The method_disclosure object is the central accountability locus of the IDSP role. An IDSP that misrepresents its methods - claiming verification it did not perform, omitting subcontractors, or misstating evidence classes - undermines the trust framework that makes its attestations evaluable. There is no cryptographic mechanism that detects this; the discipline depends on the relying party's, and the broader community's, ability to evaluate IDSP practices and on the IDSP's reputational exposure to inaccurate disclosure. Trust frameworks that recognize specific IDSPs SHOULD require audit of method-disclosure accuracy as a condition of recognition.

10. Privacy Considerations

This document inherits the privacy considerations of the core protocol [I-D.noss-jeftovic-groundmark-core], and is shaped by the guidance of [RFC6973]. The following considerations are specific to the attestation layer.

10.1. Minimal Disclosure by Construction

The claim-value structures in Section 5 are deliberately minimal. age-over-18 discloses a boolean, not a date of birth. jurisdiction-licensed discloses the existence and category of a licence in a named jurisdiction, not the licence number. kyc-basic and kyc-enhanced disclose only that the relevant screening was performed under a named framework.

Custom claim types defined under the "x-<registrant-domain>-<claim-name>" extension mechanism SHOULD follow the same discipline. Claim types that would require disclosure of richer personal data SHOULD be considered carefully against the minimal-disclosure principle and, where possible, decomposed into bounded predicates.

10.2. Public Visibility of Attestation Existence

What is publicly visible in DNS for a given agent is the existence of an attestation of a given claim_type from a named idsp, plus the high-entropy "ref=" URL. The claim value itself is reachable only by following the "ref=" URL.

This is an intentional design tradeoff. Publishing the claim type and IDSP name supports interoperability and meaningful relying-party policy; publishing the value would either require value confidentiality (complicating retrieval) or render claim-value privacy moot. The recommended deployment pattern for individual

operators concerned about this visibility is provider-hosted identities, as discussed in the core protocol's Privacy Considerations.

10.3. Linkability Across Relying Parties

A relying party that observes a Groundmark request signature obtains the agent's domain and, on attestation retrieval, the attestation's content. Multiple relying parties observing requests from the same agent can correlate by agent domain. This is a property of DNS-anchored identity and is not unique to Groundmark; an agent domain is necessarily globally unique and linkable.

Operators concerned about cross-relying-party linkability SHOULD provision per-relationship agent domains (for example, distinct subdomains per counterparty) or use ephemeral agent domains. These mitigations come at the cost of reduced amortization of attestation issuance and verification, and are accordingly out of scope for the default deployment pattern.

11. Governance Considerations

This specification permits a diversity of IDSP operators, trust frameworks, and jurisdictional regimes. It does not designate any of them.

The framework's accountability properties depend on the existence of mechanisms - external to this specification - for evaluating IDSPs. Such mechanisms may include:

* Industry-specific trust frameworks (analogous to those operated for payment networks, professional licensing bodies, or certificate-authority root programs).

* Regulator-recognized IDSP rosters for specific compliance purposes (for example, recognized KYC providers under a given AML regime).

* Community-curated lists of IDSPs evaluated for specific purposes.

* Direct evaluation by relying parties of an IDSP's public practices statement, audit reports, and method-disclosure accuracy.

The Groundmark framework supports all of these because every attestation carries the IDSP's identity, the methods performed, and the cryptographic material needed to verify the attestation. What this specification provides is the technical substrate that makes governance possible; the governance itself is the work of trust frameworks and the broader community.

12. IANA Considerations

12.1. Groundmark Claim Types Registry

IANA is requested to establish a new registry titled "Groundmark Claim Types" under a new "Groundmark" registry group.

Registration policy: Specification Required, per [RFC8126]. The designated expert SHOULD evaluate proposed registrations for:

* Adherence to the minimal-disclosure principle.

* Clarity of the claim_value schema.

* Compatibility with the level taxonomy.

The initial registry contents are the claim types defined in Section 5:

Claim Type	Level(s)	Reference
operator-verified	1	This document
age-over-18	2	This document
age-over-21	2	This document
jurisdiction-licensed	2, 3	This document
kyc-basic	2, 3	This document
kyc-enhanced	3	This document
operator-contact	0	This document

Table 1: Initial Groundmark Claim Types

12.2. Groundmark Method Identifier Registry

IANA is requested to establish a new registry titled "Groundmark Method Identifiers" under the "Groundmark" registry group.

Registration policy: Specification Required, per [RFC8126].

The initial registry contents are reserved for community development and will be populated by an early companion specification defining the verification method vocabulary. Examples of expected initial entries include "corporate-registry-lookup", "government-issued-id", "biometric-liveness", "domain-control-acme", and "organizational-domain-email-confirmation".

13. References

13.1. Normative References

[I-D.noss-jeftovic-groundmark-core]
Noss, E. and M. Jeftovic, "DNS-Anchored Identity Discovery for Autonomous Agents", Work in Progress, Internet-Draft, draft-noss-jeftovic-groundmark-core-00, 2026, <https://datatracker.ietf.org/doc/html/draft-noss-jeftovic-groundmark-core-00>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC3339] Klyne, G. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, July 2002, <https://www.rfc-editor.org/rfc/rfc3339>.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, October 2006, <https://www.rfc-editor.org/rfc/rfc4648>.

[RFC8032] Josefsson, S. and I. Liusvaara, "Edwards-Curve Digital Signature Algorithm (EdDSA)", RFC 8032, January 2017, <https://www.rfc-editor.org/rfc/rfc8032>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC8259] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, December 2017, <https://www.rfc-editor.org/rfc/rfc8259>.

[RFC8785] Rundgren, A., Jordan, B., and S. Erdtman, "JSON Canonicalization Scheme (JCS)", RFC 8785, June 2020, <https://www.rfc-editor.org/rfc/rfc8785>.

[RFC9110] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Semantics", STD 97, RFC 9110, June 2022, <https://www.rfc-editor.org/rfc/rfc9110>.

[RFC9111] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Caching", STD 98, RFC 9111, June 2022, <https://www.rfc-editor.org/rfc/rfc9111>.

13.2. Informative References

[Cameron2005] Cameron, K., "The Laws of Identity", May 2005, <https://www.identityblog.com/?p=352>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, July 2013, <https://www.rfc-editor.org/rfc/rfc6973>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, June 2017, <https://www.rfc-editor.org/rfc/rfc8126>.

Acknowledgements

The authors thank Doc Searls for substantive review of earlier drafts, the late Kim Cameron whose Laws of Identity [Cameron2005] shaped the philosophical commitments of this work, and the broader registrar, DNS, and identity communities whose infrastructure and prior art make this approach viable.

Change Log

-00

* Initial submission as the companion to [I-D.noss-jeftovic-groundmark-core].
* Split from the prior single-draft draft-noss-groundmark-agent-identity-dns.
* Replaced the open-ended methods array description with explicit reference to the Groundmark Method Identifier registry.
* Specified canonical-serialization using JCS [RFC8785], replacing ad-hoc "lexicographically sorted keys" language.

* Tightened revocation fail-mode semantics: prescriptive logging at every fail-open decision; mandated unauthenticated revocation endpoint to prevent DoS leverage.

* Added short-lived status tokens as an OPTIONAL mechanism.

* Resolved IDSP key rotation through the core protocol's "kid=" mechanism rather than as an open question.

* Added Privacy Considerations and Governance Considerations sections.

Authors' Addresses

Elliot Noss
Groundmark
Email: elliot@noss.org
URI: https://groundmark.org

Mark Jeftovic
easyDNS Technologies Inc.
Email: markjr@easydns.com