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

E. Noss
Groundmark
M. Jeftovic
easyDNS Technologies Inc.

24 May 2026

DNS-Anchored Identity Discovery for Autonomous Agents
draft-noss-jeftovic-groundmark-core-00

Abstract

This document defines the core of Groundmark, a protocol for DNS-anchored identity discovery and request authentication for autonomous software agents. Operators publish an agent's signing public key in a DNS TXT record under a reserved label, and optionally publish references to externally hosted attestations under a second reserved label. Agents authenticate HTTP requests using HTTP Message Signatures (RFC 9421) with a key identifier that resolves through DNS. DNSSEC is required for all Groundmark DNS lookups. The protocol provides operator accountability discovery without a central registry, and is designed to compose with existing identity, authorization, and payment systems rather than to replace them.

The attestation framework, including the role of Identity Service Providers, the attestation level taxonomy, and the claim vocabulary, is defined in a companion document.

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

Autonomous software agents are increasingly conducting transactions on the Internet on behalf of human and organizational principals. Existing work addresses several adjacent concerns: HTTP-native payment flows [X402], per-session delegated authorization protocols, and tool invocation frameworks. None of these answers a prior question: when an agent presents itself to a counterparty, is there an accountable party behind it, and how is that established without a central registry?

This document defines Groundmark, a protocol that uses the Domain Name System (DNS) as the discovery anchor for agent identity. DNS is well suited to this role: it is open and well-governed; domain registrants are verified by registrars operating under established policy; the infrastructure is universally deployed; and any domain holder can publish agent identity records without any new global registry or delegation. The approach is modelled on DomainKeys Identified Mail (DKIM) [RFC6376], which established the pattern of publishing a public key in a DNS TXT record under a reserved label for the purpose of verifying cryptographic assertions in an application-layer protocol.

Groundmark separates three concerns:

* Identity discovery: locating an agent's signing public key in DNS.

* Request authentication: binding an HTTP request to an agent's key using HTTP Message Signatures [RFC9421].

* Attestation discovery: pointing relying parties to externally hosted attestations describing what has been verified about the operator.

The first two are specified in this document. The third is introduced mechanically here, by defining a DNS record that references attestation endpoints; the trust semantics of those attestations are defined in the companion document [I-D.noss-jeftovic-groundmark-attestation].

The semantics of the agent's domain string itself are not material to this protocol. An agent at "agent47.example.com" carries the same protocol weight as one at "x7q.example.com". What matters is the verifiable chain of control from the DNS root through a registrar-verified registrant to the holder of the agent's subdomain.

1.1. Design Principles

Groundmark is designed around the following principles:

* Authenticate the minimum necessary. The intent is to enable permissionless interaction by default and to make stronger attestations available only where a relying party's risk profile demands them.

* DNS is a discovery layer, not a claims database. TXT records publish cryptographic material and point to where attestations are held. Attestations themselves are served over HTTPS.

* No single point of control. DNS is the common substrate, but no single entity controls which operators may publish agent identities, and the attestation framework defined in the companion document permits a diversity of Identity Service Providers.

1.2. Non-Goals

This specification explicitly does not:

* Define legal identity, human identification, or any form of personal attribution requirement.

* Define a global trust framework, governance regime, or compliance requirement.

* Mandate participation. An operator may decline to publish Groundmark records; a relying party may decline to require them.

* Replace decentralized identifiers (DIDs), Verifiable Credentials, OAuth, OpenID Connect, MCP, or any other identity, authorization, or capability protocol. Groundmark is designed to compose with these systems and MAY act as a discovery and bootstrap layer for any of them.

* Define a new cryptographic primitive. Request authentication uses HTTP Message Signatures [RFC9421] with conventional algorithms.

* Define attestation semantics, IDSP obligations, attestation levels, or claim vocabularies. Those are the subject of the companion document [I-D.noss-jeftovic-groundmark-attestation].

A consequence of the first three of these non-goals is that anonymous and pseudonymous agent operation remains a first-class deployment mode under Groundmark. The presence of an agent's public key in DNS does not, by itself, identify or accountably link the operator to a legal person; that linkage is the subject of optional attestations described in the companion document, and is opt-in by both the operator (who chooses whether to obtain and publish an attestation) and the relying party (which chooses whether to require one).

1.3. 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

Operator An person or organization that deploys and controls an autonomous agent. The operator is the principal whose accountability is at issue in the attestation framework defined by the companion document.

Agent An autonomous software instance acting on behalf of an operator. An agent holds a cryptographic key pair; its public key is published in DNS under its agent domain.

Agent Domain A fully qualified domain name (FQDN) that serves as the agent's identity anchor in DNS. Typically a subdomain under the operator's registered domain (e.g., "purchasing.acmecorp.example") but MAY be a second-level domain.

Identity Service Provider (IDSP) A trusted third party that verifies a claim about an operator or agent and issues a signed attestation. Defined fully in [I-D.noss-jeftovic-groundmark-attestation]; referenced here only insofar as the _agentclaim record points to attestations issued by IDSPs.

Relying Party An agent, API endpoint, or service that receives an agent's request, verifies the request signature against the agent's DNS-published key, and optionally evaluates attestations referenced from DNS.

3. System Overview

A Groundmark-enabled HTTP request proceeds as follows.

1. The operator publishes an _agentid TXT record at the agent domain, containing the agent's signing public key, and OPTIONALLY one or more _agentclaim TXT records referencing externally hosted attestations. All such records MUST be published in a DNSSEC-signed zone.

2. The agent signs each HTTP request using HTTP Message Signatures [RFC9421] as profiled in Section 5. The signature uses the agent's private key and identifies the corresponding public key through a keyid parameter that resolves to the agent domain.

3. The relying party retrieves the _agentid TXT record for the agent domain using a DNSSEC-validating resolver and verifies the request signature against the published public key. A successful verification establishes that the request originated from a party that controls the agent domain.

4. If the relying party's policy requires attestations beyond what is established by signature verification alone, it retrieves _agentclaim records for the agent domain, fetches the referenced attestations, and evaluates them according to the framework specified in the companion document.

DNS in this design is a discovery layer. It points to where keys and attestations are found; it does not itself host claim content. This mirrors the role DNS already plays for email authentication mechanisms such as DKIM [RFC6376] and DMARC [RFC7489].

3.1. Worked Example

The following non-normative example illustrates a complete Groundmark exchange. An operator deploys an agent under the domain "purchasing.acmecorp.example", publishing one identity record and one attestation reference:

```
_agentid.purchasing.acmecorp.example. 300 IN TXT
  "v=gm1; k=ed25519; p=oW8nN1xJxGv...HzG"

_agentclaim.purchasing.acmecorp.example. 300 IN TXT
  "v=gm1; claim=operator-verified;
   idsp=verify.example.net;
   ref=https://verify.example.net/a/4f9f3e1c7ab93d4f8d22;
   exp=2026-12-01"
```

When the agent issues an HTTP POST to a relying party, the request carries Signature-Input, Signature, and Content-Digest header fields per [RFC9421] and [RFC9530]:

```
POST /orders HTTP/1.1
Host: api.supplier.example
Content-Type: application/json
Content-Digest: \
  sha-256=:X48E9qOokqqrvdts8nOJRJN3OWDUoyWxBf7kbu9DBPE=:
Signature-Input: gm=("@method" "@authority" "@target-uri" \
    "content-digest");created=1747654321;\
    expires=1747654621;\
    keyid="purchasing.acmecorp.example";alg="ed25519"
Signature: gm=:9R...base64url...==:

{"sku":"X-42","quantity":3}
```

The relying party:

1. Parses Signature-Input and confirms required components and parameters.

2. Resolves _agentid.purchasing.acmecorp.example via a DNSSEC-validating resolver.

3. Verifies the Content-Digest against the request body.

4. Reconstructs the [RFC9421] signature base from the covered components.

5. Verifies the Ed25519 signature against the public key from the _agentid record.

If the relying party's policy requires more than Level 0, it additionally retrieves _agentclaim.purchasing.acmecorp.example, fetches the attestation at the ref= URL, verifies it per [I-D.noss-jeftovic-groundmark-attestation], and applies its policy.

4. DNS Record Formats

Groundmark defines two TXT record formats published under reserved underscore labels at the agent domain. Both share the version tag "v=gm1" to distinguish them from other TXT record uses at the same domain.

The use of two distinct labels reflects a deliberate separation of concerns: _agentid carries the cryptographic material that is the root of request authentication, and is expected to be small, stable, and aggressively cached only within a short TTL. _agentclaim carries references to attestations whose lifecycle is independent of the signing key and which may be added, removed, or replaced without affecting the agent's underlying identity.

4.1. The _agentid TXT Record

The _agentid TXT record is published at:

```
_agentid.<agent-domain>. IN TXT
  "v=gm1; k=ed25519; p=<base64url-public-key>"
```

The record declares the agent's signing public key and algorithm. It is the root of cryptographic verification for the agent identity.

4.1.1. ABNF Grammar

```
agentid-record  = version 1*( ";" SP agentid-param )
version         = "v=gm1"
agentid-param   = key-alg-param / pubkey-param / keyid-param /
                  ext-param
key-alg-param   = "k=" key-alg
key-alg         = "ed25519" / x-token
pubkey-param    = "p=" base64url-value
keyid-param     = "kid=" 1*( ALPHA / DIGIT / "-" / "_" / "." )
ext-param       = param-name "=" param-value
param-name      = 1*( ALPHA / DIGIT / "-" )
param-value     = *( %x21-3A / %x3C-7E )
                  ; printable ASCII excluding ";"
base64url-value = 1*( ALPHA / DIGIT / "-" / "_" / "=" )
x-token         = "x-" 1*( ALPHA / DIGIT / "-" )
```

The OPTIONAL "kid=" parameter allows an operator to publish an explicit key identifier matching the keyid parameter used in the request signature, which supports key rotation by deploying multiple _agentid records during transition (see Section 6.1).

4.1.2. Processing Requirements

The "k=" (key algorithm) and "p=" (public key) parameters MUST be present. The version tag "v=gm1" MUST appear first.

This document defines "ed25519" as the only mandatory-to-implement algorithm. The "p=" field MUST contain the Base64url-encoded [RFC4648] raw public key bytes for the specified algorithm (32 bytes for Ed25519). Support for additional algorithms MAY be added by future documents updating this specification.

An operator publishing a single signing key MUST publish exactly one _agentid record. An operator may publish multiple _agentid records only as part of a key rotation procedure (see Section 6.1); in that case, every published record MUST carry a distinct "kid=" value, and request signatures MUST identify the intended key by keyid.

A relying party MUST treat the identity as invalid, and MUST NOT proceed with signature verification against any record at the label, if any of the following conditions hold at the time of lookup:

* More than one _agentid record is present at the label and any of them lacks a "kid=" parameter.

* More than one _agentid record is present and two or more of them share the same "kid=" value.

Operators SHOULD set the TTL of the _agentid record to 300 seconds or less to enable timely revocation. Relying parties MUST respect DNS TTL values and MUST NOT cache _agentid records beyond their TTL.

Relying parties MUST use a DNSSEC-validating resolver when retrieving _agentid records. A response that fails DNSSEC validation MUST be treated as an invalid identity. See Section 7.1.

4.2. The _agentclaim TXT Record

The _agentclaim TXT record points to an externally hosted attestation about the operator or agent. The semantics of attestations themselves are defined in [I-D.noss-jeftovic-groundmark-attestation]; this document defines only the DNS record format and processing requirements.

The most common form references an IDSP-hosted attestation:

```
_agentclaim.<agent-domain>. IN TXT
  "v=gm1; claim=<type>; idsp=<domain>;
   ref=<https-url>; exp=<ISO8601-date>"
```

A self-asserted variant uses "val=" instead of "ref=" and "idsp=", and carries no IDSP signature:

```
_agentclaim.<agent-domain>. IN TXT
  "v=gm1; claim=operator-contact;
   val=ops@acmecorp.example"
```

Relying parties MUST treat self-asserted claims as unverified operator assertions. The companion document specifies how such records are incorporated into the attestation level taxonomy.

Multiple _agentclaim records MAY be present at the same label. Each represents an independent claim from a potentially different IDSP and is evaluated independently.

4.2.1. ABNF Grammar

```
agentclaim-record = version 1*( ";" SP agentclaim-param )
version           = "v=gm1"
agentclaim-param  = claim-param / idsp-param / ref-param /
                    exp-param / val-param / ext-param
claim-param       = "claim=" claim-type
claim-type        = token / x-token
idsp-param        = "idsp=" hostname
ref-param         = "ref=" https-url
exp-param         = "exp=" date-value
val-param         = "val=" param-value
ext-param         = param-name "=" param-value
token             = 1*( ALPHA / DIGIT / "-" )
hostname          = 1*( ALPHA / DIGIT / "." / "-" )
https-url         = "https://" *( %x21-7E )
date-value        = 4DIGIT "-" 2DIGIT "-" 2DIGIT
                    [ "T" 2DIGIT ":" 2DIGIT ":" 2DIGIT
                      [ "Z" /
                        ( ( "+" / "-" ) 2DIGIT ":" 2DIGIT ) ] ]
param-name        = 1*( ALPHA / DIGIT / "-" )
param-value       = *( %x21-3A / %x3C-7E )
x-token           = "x-" 1*( ALPHA / DIGIT / "-" )
```

4.2.2. Processing Requirements

The "claim=" field MUST be present in all _agentclaim records.

When "ref=" is present, "idsp=" MUST also be present. When "val=" is present, "ref=" and "idsp=" MUST NOT be present. A record that satisfies neither shape MUST be ignored.

Relying parties MUST ignore records with unrecognized "claim=" values rather than treating them as errors, to preserve forward compatibility.

"ref=" URLs MUST use the HTTPS scheme. The path component of a "ref=" URL MUST contain at least 128 bits of cryptographically random entropy (for example, a UUIDv4 or 22-character Base64url-encoded random string). This document does not treat "ref=" URLs as confidential, but the entropy requirement defends against enumeration attacks against IDSP attestation endpoints (see Section 7.4).

Operators SHOULD set the TTL of _agentclaim records to 300 seconds or less. The "exp=" value provides claim-level expiry that is independent of DNS TTL; relying parties MUST NOT use a claim whose "exp=" time has passed, regardless of cache state.

Relying parties MUST use a DNSSEC-validating resolver when retrieving _agentclaim records. A response that fails DNSSEC validation MUST be treated as an absent record.

5. Request Authentication

Groundmark profiles HTTP Message Signatures [RFC9421] for the purpose of authenticating agent requests. This document does not define a bespoke signing mechanism: the signature base, the Signature-Input header field, and the Signature header field are all as defined by [RFC9421]. This section specifies which of the choices left open by [RFC9421] are constrained for Groundmark implementations.

The use of [RFC9421] provides body integrity, authority binding, nonce semantics, and a well-reviewed canonicalization, and avoids introducing a novel cryptographic construction.

5.1. Signing Algorithm

Groundmark implementations MUST support Ed25519 [RFC8032], identified in Signature-Input by the algorithm name "ed25519" as registered in the "HTTP Signature Algorithms" registry established by [RFC9421]. Other algorithms MAY be used where both signer and verifier support them, but Ed25519 is the only mandatory-to-implement algorithm in this document.

5.2. Key Identification

The keyid signature parameter in Signature-Input MUST be one of the following two forms:

* The agent’s fully qualified domain name (FQDN), in lowercase ASCII, with no "#" suffix. In this form, the relying party retrieves the _agentid record at the named domain; verification proceeds if and only if exactly one record is present at the label. (Equivalently, the bare-FQDN form is only valid outside of a key-rotation window.)

* The agent’s FQDN followed by "#" and an opaque key identifier; for example, "purchasing.acmecorp.example#k2". In this form, the relying party retrieves the _agentid records at the named domain and selects the record whose "kid=" parameter matches the identifier after the "#". This form MUST be used whenever multiple _agentid records are concurrently published during key rotation.

A relying party MUST reject a signature whose keyid does not resolve to exactly one usable _agentid record under the rules above.

5.3. Required Covered Components

A Groundmark-compliant signature MUST cover, at minimum, the following [RFC9421] derived components:

* @method

* @authority

* @target-uri

A Groundmark-compliant signature on any HTTP request that carries a request body MUST additionally cover the Content-Digest HTTP field as defined in [RFC9530]. The signer MUST include a Content-Digest field in the signed request and MUST select an algorithm from the "Hash Algorithms for HTTP Digest Fields" registry; the algorithm "sha-256" is mandatory to implement for both signers and verifiers.

A signature on a request with no body (for example, most GET requests) MAY omit Content-Digest.

5.4. Required Signature Parameters

The Signature-Input header field MUST include the following parameters:

* "created": the time of signing, as a UNIX timestamp in seconds.

* "keyid": as specified in Section 5.2.

* "alg": present and matching the algorithm associated with the public key in the _agentid record. (As a profile of [RFC9421], this document REQUIRES "alg" to be present, although [RFC9421] itself permits the algorithm to be derived from the key.)

The Signature-Input header field SHOULD include "expires", set to a time no more than 300 seconds after "created". Relying parties MUST reject signatures whose "created" time differs from the verifier's current clock by more than 300 seconds in either direction, and MUST reject signatures whose "expires" time, if present, is in the past.

For higher-risk transactions, relying parties SHOULD require a "nonce" parameter, MUST track recently observed nonces for the duration of the verification window, and MUST reject a signature whose nonce has been observed within that window. Maintenance of a nonce cache is a relying-party concern; the size of the cache and its eviction policy are out of scope for this specification.

5.5. Verification Steps

A relying party verifying a Groundmark request signature MUST perform the following steps. Failure of any step MUST result in rejection of the signature.

1. Parse the Signature-Input and Signature header fields per [RFC9421]. Confirm that the required covered components and signature parameters specified in Section 5.3 and Section 5.4 are present.

2. Verify that "created" is within 300 seconds of the verifier's current clock, and that "expires", if present, is in the future.

3. Resolve the agent domain from the keyid parameter and retrieve the appropriate _agentid TXT record using a DNSSEC-validating resolver. If DNSSEC validation fails or no usable record is present, reject the request.

4. If a request body is present, verify that the value in the Content-Digest header field matches a digest of the received body, computed using an algorithm the verifier supports.

5. Reconstruct the signature base per [RFC9421] from the covered components in the order indicated by Signature-Input.

6. Verify the signature in the Signature header field against the reconstructed signature base using the public key from the _agentid record.

7. If the verifier's policy requires attestations beyond identity verification, proceed to retrieve and evaluate _agentclaim records per [I-D.noss-jeftovic-groundmark-attestation].

6. Revocation

Two independent revocation mechanisms are available to operators.

DNS-layer revocation The operator deletes or rotates the _agentid record. The agent identity becomes invalid within the record's TTL. Operators performing emergency revocation SHOULD also publish an empty replacement record or otherwise serve a known-unusable value during the TTL window, and SHOULD set TTLs to 300 seconds or less in normal operation to minimize the worst-case revocation delay.

Claim-layer revocation The IDSP marks a specific attestation as revoked at the attestation endpoint or its associated revocation endpoint. The agent identity remains usable; only the specific claim is invalidated. The mechanics of claim-layer revocation are specified in [I-D.noss-jeftovic-groundmark-attestation].

These mechanisms are independent: deletion of an _agentid record revokes the entire agent identity (and implicitly invalidates all attestations referencing it); revocation of a single _agentclaim leaves the underlying identity intact.

6.1. Key Rotation

Operators rotating an agent signing key SHOULD use the following procedure to avoid request rejection during transition.

If the operator’s current _agentid record does not carry a "kid=" parameter, the operator MUST first publish a replacement of the current record that includes a "kid=" parameter, and wait at least one TTL period before proceeding. This is because, during rotation, the validity rule of Section 4.1.2 requires every published record to carry a "kid=".

The rotation itself proceeds as follows:

1. Generate the new key pair.

2. Publish a second _agentid record at the agent domain, with a "kid=" parameter distinct from the existing record.

3. Begin issuing requests with the new key. Requests MUST identify the new key via the keyid signature parameter using the "<FQDN>#<kid>" form specified in Section 5.2.

4. After a transition period not less than twice the published _agentid TTL, remove the previous _agentid record.

Relying parties supporting key rotation MUST select the _agentid record matching the keyid value carried in the signature. Relying parties MAY cache multiple _agentid records concurrently up to their respective TTLs.

7. Security Considerations

7.1. DNSSEC Requirement

All DNS lookups in the Groundmark verification flow MUST be performed using a DNSSEC-validating resolver [RFC4033]. Without DNSSEC validation, DNS cache poisoning attacks could substitute an attacker’s public key for the legitimate agent’s key, rendering the entire verification flow meaningless. Implementations MUST treat any of the following as a fatal verification failure: a SERVFAIL response from a validating resolver, an explicit AD=0 response from a recursive resolver that the implementation relies on for validation, or any other condition in which the implementation cannot confirm a validated answer.

This DNSSEC requirement applies to operators publishing _agentid and _agentclaim records, and to IDSPs publishing their own signing keys. It does not require DNSSEC adoption across the general Internet. Entities making the deliberate decision to participate in Groundmark are the appropriate population to bear this operational requirement, and the requirement is proportionate to the trust being claimed.

Implementations MAY use DNS-over-HTTPS [RFC8484] or other authenticated transport to a trusted recursive resolver as part of their DNSSEC strategy, but the validation chain itself MUST be verified end-to-end.

7.2. Replay Attacks

The 300-second "created" window in [RFC9421] signatures limits replay exposure. Relying parties servicing high-value transactions SHOULD require a "nonce" parameter and maintain a nonce cache for the duration of the verification window; this prevents replay within the window at the cost of relying-party state.

The "@authority" covered component, required by Section 5.3, binds the signature to a specific target host, preventing cross-service replay even if the signature is otherwise valid.

7.3. Body Integrity

For requests with a body, the Content-Digest covered-component requirement of Section 5.3 provides body integrity: an intermediary that modifies the request body will invalidate the signature. Implementations MUST verify the Content-Digest value against the received body before accepting the signature as valid.

7.4. ref= URL Entropy

"ref=" URLs function as opaque bearer references to attestation content. The minimum 128-bit entropy requirement in Section 4.2.2 defends against enumeration of IDSP attestation endpoints. IDSPs MUST use cryptographically random URL path components and MUST NOT use sequential, predictable, or low-entropy identifiers. Implementations SHOULD avoid unnecessary exposure of "ref=" URLs in logs, telemetry, or third-party-accessible interfaces.

The high-entropy "ref=" design does not by itself render attestation content confidential at the network layer; rather, it ensures that relying parties retrieve attestations by reference rather than by guessable enumeration. Confidentiality properties of attestation content are specified by the companion document.

7.5. Subdomain Takeover and Endpoint Decommissioning

"ref=" URLs are embedded in DNS TXT records and may be cached by relying parties. If an IDSP decommissions a subdomain referenced by outstanding attestations without first invalidating those attestations, a third party who later acquires the subdomain could serve fraudulent attestations at existing URLs. To mitigate this:

* IDSPs MUST NOT release a hostname used in attestation "ref=" URLs until all attestations referencing that hostname have either expired (per "exp=" in their _agentclaim records) or been explicitly revoked.

* IDSPs SHOULD serve HTTP 410 Gone responses at decommissioned attestation paths for a minimum of 90 days before releasing the hostname.

* Relying parties MUST treat a TLS certificate change or unreachable endpoint at a previously valid "ref=" URL as cause for re-checking the corresponding _agentclaim record and the IDSP’s identity-discovery records.

Additional revocation and decommissioning guidance specific to the IDSP role is given in [I-D.noss-jeftovic-groundmark-attestation].

7.6. Key Compromise

Compromise of an agent signing key permits an attacker to impersonate the agent until the corresponding _agentid record is removed and caches expire. To minimize exposure:

* Operators SHOULD set _agentid TTLs to 300 seconds or less.

* Operators SHOULD provision the rotation procedure of Section 6.1 in advance so that emergency key replacement can be performed without delay.

* Relying parties MUST NOT cache _agentid records beyond their DNS-declared TTL.

This document defines no mechanism for an emergency "kill" signal distinct from DNS-level deletion. Operators requiring more aggressive revocation guarantees should consider running their authoritative nameservers with the capacity to make rapid zone updates and short TTLs.

8. Privacy Considerations

This section addresses the privacy properties of the protocol defined in this document, in the spirit of [RFC6973]. Privacy considerations arising from specific attestation types are the subject of the companion document.

8.1. Organizational Disclosure Model

Groundmark's primary deployment model is organizational. The operator is, in the typical case, a company or other organization deploying agents in a commercial or operational context, under domains that already publish substantial public metadata: MX records, SPF, DKIM selectors, DMARC policy, CAA records, and (subject to applicable law) registrant data. The addition of _agentid and _agentclaim records extends this existing organizational-disclosure model, rather than introducing a new category of disclosure.

What is publicly visible when an operator publishes Groundmark records is:

* The fact that the named agent domain participates in Groundmark.

* The agent's signing public key.

* The type of any attestations available for the agent (the "claim=" field) and the identity of the IDSP issuing them (the "idsp=" field).

Attestation content itself is not in DNS. It is reachable only by following the high-entropy "ref=" URL, which is not enumerable from the DNS record alone.

8.2. Individual Operator Considerations

The organizational framing above does not address the case of an individual operator (sole proprietor, freelancer, or hobbyist) deploying agents under a personal domain. A _agentclaim record indicating, for example, that a kyc-basic attestation exists under the operator's personal domain reveals compliance posture in a way that is qualitatively different from the same record under a corporate domain.

For this case, the recommended pattern is provider-hosted agent identities, analogous to the WHOIS privacy proxy model long used in domain registration. An individual operator participating in Groundmark SHOULD publish agent identity records under a provider's domain rather than under a personal domain, where the provider's

underlying relationship with the operator preserves the accountability chain that an IDSP-issued attestation depends on. This keeps the operator's personal domain outside the public record while preserving the attestation framework's accountability semantics.

This is a recommendation, not a normative requirement; nothing in this specification prevents an individual operator from publishing Groundmark records under a personal domain if they choose.

8.3. Selective Disclosure

The protocol publishes only public keys in _agentid and only attestation references (not values) in _agentclaim. The attestation framework specified in the companion document is built around minimal disclosure: attestation values are bounded predicates such as "the operator is over 18", not the underlying personal data.

8.4. Resolver-Side Observation

DNS resolution of _agentid and _agentclaim records is, like any DNS lookup, observable by network operators on the path between the relying party and the authoritative servers. Relying parties concerned about this leakage SHOULD use authenticated, encrypted transports to their recursive resolver (such as DNS-over-HTTPS [RFC8484]).

9. Operational Considerations

9.1. Caching and Resolver Load

Groundmark verification is designed to be amortized through caching, not performed in full on every request. A relying party that has recently verified a request from a given agent domain may have its _agentid record cached at the local resolver; an attestation fetched once may be valid for hours or days, subject to its "exp=" time and the IDSP's Cache-Control directives.

To support this:

* IDSPs MUST set Cache-Control max-age values on attestation responses consistent with the attestation's expected freshness, and MUST NOT set values so short as to force effectively per-request revalidation. Specific normative guidance for IDSPs is given in the companion document.

* Relying parties SHOULD cache _agentid records and attestation payloads up to the lesser of the resource's stated lifetime and any local cache policy.

* Implementations operating at high transaction volumes (for example, serving requests from AI agent fleets) SHOULD prefer a resolver architecture that aggregates DNS lookups for an agent domain rather than re-resolving per request.

The two-level structure (cheap, near-constant _agentid lookups; less frequent, cacheable attestation fetches) is intended to make Groundmark workable at scales characteristic of automated agent deployments.

9.2. Coexistence with Other Protocols

_agentid and _agentclaim are scoped underscored node names under [RFC8552]. They do not conflict with other underscore-labeled records published at the same domain, including _dmarc, _dkim, DANE TLSA records, or vendor-defined labels such as those used by [IoDNS]. An operator MAY simultaneously publish Groundmark identity records and IoDNS metadata records under the same agent domain without interaction at the DNS layer.

10. Related Work

This section positions Groundmark relative to other work in adjacent problem spaces. Groundmark is designed to compose with these specifications rather than to replace them.

10.1. HTTP Message Signatures (RFC 9421)

Groundmark uses [RFC9421] for request authentication. It does not modify or extend [RFC9421]; it specifies a profile (covered components, required signature parameters, key identification convention) suitable for DNS-anchored agent identity.

10.2. DKIM and DMARC

Groundmark's DNS publication pattern is modelled on DKIM [RFC6376], which established the practical pattern of publishing a public key in a DNS TXT record under a reserved label for the purpose of application-layer signature verification. [RFC7489] (DMARC) demonstrates the extension of this pattern to policy publication. Groundmark applies the same pattern to agent identity rather than email authentication.

10.3. W3C Decentralized Identifiers and Verifiable Credentials

Groundmark and the W3C identity ecosystem operate at different layers. Groundmark addresses operator-accountability discovery: a stable, DNS-anchored pointer to who stands behind an agent. Decentralized Identifiers (DIDs) and Verifiable Credentials (VCs) address identifier resolution and credential format and presentation. The two are composable: an IDSP operating within the Groundmark framework MAY issue a Verifiable Credential as its attestation payload and reference it via a Groundmark "ref=" URL, and an operator MAY publish a DID in addition to or instead of a domain-anchored identity by extending _agentid semantics under future versions of this specification. Nothing in Groundmark precludes any of these compositions.

10.4. Agent Name Service (ANSv2) and DNS-AID

Several other DNS-anchored agent-discovery proposals exist as Internet-Drafts at the time of writing. ANSv2 addresses agent naming and versioning; DNS-AID addresses DNS-based agent discovery using SVCB records. Groundmark addresses operator identity and attestation under different underscored DNS labels and may coexist with these protocols at the same domain. The architectural distinction is that ANSv2 and DNS-AID address what an agent is and how it is reached, while Groundmark addresses who is accountable for the agent and what has been verified about that accountable party.

10.5. x402 and IoDNS

[X402] defines an HTTP-native payment protocol in which an agent receiving an HTTP 402 response pays in stablecoins and retries the request. [IoDNS] is a DNS metadata framework that publishes structured machine-readable documents at a domain, including payment endpoint discovery as one document type. Groundmark’s _agentid and _agentclaim records coexist with IoDNS metadata records and with any x402 discovery records at the same agent domain, under different underscored labels. An agent presenting both Groundmark request signatures and IoDNS-advertised payment capability provides relying parties with operator accountability and payment routing from a single DNS hierarchy.

11. IANA Considerations

11.1. Underscored and Globally Scoped DNS Node Names Registry

IANA is requested to add the following entries to the "Underscored and Globally Scoped DNS Node Names" registry established by [RFC8552]:

RR Type	_NODE NAME	Reference
TXT	_agentid	This document
TXT	_agentclaim	This document

Table 1: Groundmark underscore label registrations

The registry operates under Expert Review per [RFC8552].

11.2. HTTP Signature Algorithms

This document does not define new entries in the "HTTP Signature Algorithms" registry established by [RFC9421]. Groundmark implementations use registered algorithms only; the mandatory-to-implement algorithm "ed25519" is already registered by [RFC9421].

12. References

12.1. Normative References

[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>.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, March 2005, <https://www.rfc-editor.org/rfc/rfc4033>.

[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>.

[RFC8552] Crocker, D., "Scoped Interpretation of DNS Resource Records through "Underscored" Naming of Attribute Leaves", BCP 222, RFC 8552, March 2019, <https://www.rfc-editor.org/rfc/rfc8552>.

[RFC9421] Backman, A., Ed., Richer, J., Ed., and M. Sporny, "HTTP Message Signatures", RFC 9421, February 2024, <https://www.rfc-editor.org/rfc/rfc9421>.

[RFC9530] Polli, R. and L. Pardue, "Digest Fields", RFC 9530, February 2024, <https://www.rfc-editor.org/rfc/rfc9530>.

12.2. Informative References

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

[I-D.noss-jeftovic-groundmark-attestation] Noss, E. and M. Jeftovic, "Groundmark Attestation Framework and Identity Service Provider Profile", Work in Progress, Internet-Draft, draft-noss-jeftovic-groundmark-attestation-00, 2026, <https://datatracker.ietf.org/doc/html/draft-noss-jeftovic-groundmark-attestation-00>.

[IoDNS] Jeftovic, M., "Intelligence-over-DNS (IoDNS)", 2025, <https://github.com/easydns/iodns>.

[RFC6376] Crocker, D., Ed., Hansen, T., Ed., and M. Kucherawy, Ed., "DomainKeys Identified Mail (DKIM) Signatures", STD 76, RFC 6376, September 2011, <https://www.rfc-editor.org/rfc/rfc6376>.

[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>.

[RFC7489] Kucherawy, M., Ed. and E. Zwicky, Ed., "Domain-based Message Authentication, Reporting, and Conformance (DMARC)", RFC 7489, March 2015, <https://www.rfc-editor.org/rfc/rfc7489>.

[RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", RFC 8484, October 2018, <https://www.rfc-editor.org/rfc/rfc8484>.

[X402] Coinbase, "x402: An HTTP-Native Payment Protocol", 2025, <https://x402.org>.

Acknowledgements

The authors thank Doc Searls for substantive review of earlier drafts, and the broader registrar and DNS community whose infrastructure makes this approach viable. The authors thank Kim Cameron, whose Laws of Identity [Cameron2005] informed the design of the attestation framework defined in the companion document.

Change Log

-00

* Initial submission.

* Split from prior draft draft-noss-groundmark-agent-identity-dns into a narrow core protocol document plus a companion attestation framework document.

* Replaced the bespoke Agent-Identity HTTP header with a profile of HTTP Message Signatures [RFC9421] including required coverage of @method, @authority, @target-uri, and Content-Digest [RFC9530] for requests with bodies.

* Added explicit Non-Goals section.

* Added Privacy Considerations and Operational Considerations sections.

* Added Related Work section.

* Specified key rotation procedure using a "kid=" parameter and the keyid signature parameter.

Authors' Addresses

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

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