Network Working Group
Internet-Draft
Intended status: Informational
Expires: 7 November 2026

B. Morrison
Alter Meridian Pty Ltd

6 May 2026

Discovery of Model Context Protocol Servers via DNS TXT Records
draft-morrison-mcp-dns-discovery-03

Abstract

This document defines a DNS-based mechanism for the discovery of Model Context Protocol (MCP) servers, the identity properties of the organisations that operate them, and (new in this revision) the cryptographic identity envelope bound to an individual Sovereign-tier ~handle published under the same zone. Three TXT resource records are defined. The _mcp.<domain> record (defined in v01) advertises the presence, endpoint URL, transport protocol, cryptographic identity, and capability profile of an MCP server associated with a domain name. The _org-alter.<domain> record (introduced in v02) advertises the canonical organisational identity of the domain operator: legal entity name, registry identifier, founding date, primary regions of operation, and any regulatory frameworks under which the operator is bound to refuse external automated access. The _alter.<domain> record (introduced in this revision) publishes an Ed25519-signed identity envelope binding a ~handle to a public key, an IdentityLog Signed Tree Head root, and a revocation commitment. Taken together, the three records provide service discovery, organisational identity bootstrap, and individual identity recognition from a single canonical source: the domain's own DNS zone. This revision additionally requires DNSSEC [RFC4033] validation of envelope responses and a DANE TLSA [RFC6698] pin binding the MCP endpoint's leaf certificate to the published zone. A companion URI scheme (alter:) is registered provisionally with IANA per [RFC7595] for handle dispatch. The mechanism complements HTTPS-based discovery (.well-known/mcp/server-card.json and .well-known/alter-envelope.json) by providing a lightweight, resolver-cached bootstrap that requires no HTTPS round-trip. The design follows the precedent established by DKIM [RFC6376], SPF [RFC7208], DMARC [RFC7489], MTA-STS [RFC8461], and the existing _mcp. / _org-alter. labels of v01-v02.

Status of This Memo

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

Copyright Notice

Table of Contents

1. Introduction

Model Context Protocol (MCP) [MCP] is an open protocol for structured interaction between AI agents and tool-providing servers. A complete agent-to-organisation-to-individual interaction chain has three distinct discovery requirements:

1. *Service discovery.* Where is the MCP server endpoint? What transport does it speak? What cryptographic key authenticates it? This is the question v01 of this document answers via the _mcp.<domain> record.

2. *Organisational identity bootstrap.* Who is the organisation operating the server? What is its legal entity? Where is it registered? Under what regulatory frameworks does it operate, and which automated access pathways must it refuse to participate in? This is the question v02 answers via the _org-alter.<domain> record.

3. *Individual identity recognition.* Who is the Sovereign-tier person bound to a ~handle hosted under the domain? What public key signs their statements? What append-only log anchors the lifecycle of their identity? How may their envelope be revoked? This is the question v03 introduces via the _alter.<domain> record.

The three questions are distinct. An MCP client may need to discover an endpoint without caring about the operator's identity or any individual handle. An onboarding wizard installing an org-alter instance may need to read the operator's identity without caring (yet) about the MCP endpoint. A recognition verifier (resolving an alter:~blake URI) needs the individual envelope without necessarily invoking an MCP session. Conflating any two of these into a single TXT record would force every consumer to parse fields it does not need and would crowd the 255-octet character-string limit. Splitting them across three underscore-prefixed labels mirrors the pattern established by DKIM (_domainkey._domain) and DMARC (_dmarc._domain): each record serves a single semantic purpose.

This revision is fully backward-compatible with v01 and v02. Implementations that consume only the _mcp.<domain> record continue to work unchanged. Implementations that wish to bootstrap an org-alter identity may additionally query _org-alter.<domain>. Implementations that wish to recognise an individual ~handle may additionally query _alter.<domain>.

The envelope layer formalised in the new _alter.<domain> record is specified in full by a companion document, the ALTER DNS Publication specification [ALTER-DNS-PUB], which pins the envelope JSON schema, the JSON Canonicalisation Scheme (JCS, [RFC8785]) serialisation, and the resolver-side verification algorithm. This document is the IETF-track surface for the underscore-prefixed DNS label registration and its DNSSEC / DANE / IdentityLog cross-references; it does not duplicate the envelope wire format beyond what is necessary to specify a conformant TXT grammar.

The individual-identity layer is grounded, as the organisational layer is, in the identity field framework of [MORRISON-IFT]. A ~handle is not a reserved alphanumeric slot but a durable recognition attractor in the identity field. A DNS record provides a discrete checkpoint into that field: the envelope published at _alter.<zone> is the handle-holder's own canonical declaration, signed by their Ed25519 key, witnessed by the IdentityLog STH anchor surface [ALTER-STH], and consumable by any resolver with access to a DNSSEC-validating recursive resolver.

1.1. 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

(Terminology from v01 and v02 is retained. Additional terms introduced in this revision are defined below.)

Envelope An Ed25519-signed JSON object binding a ~handle to a public key, an IdentityLog root reference, an inception timestamp, a revocation hash commitment, a signature algorithm tag, a detached signature, and a caveats array. The envelope is the unification primitive of the ALTER identity architecture. Its full JSON schema, canonical serialisation rule, and verification procedure are pinned by [ALTER-DNS-PUB]. This document specifies only the TXT grammar that carries the five load-bearing envelope fields across DNS.

~handle A Sovereign-tier identifier, leading tilde mandatory (e.g. ~blake). Bot-tier handles carry a .bot suffix (e.g. ~alice.bot); Instrument-tier handles use the prefix ~cc- (e.g. ~cc-opus-4-7).

IdentityLog The append-only transparency log anchoring envelope lifecycle events (mint, caveat-add, revocation, key-rotation). A Signed Tree Head ("STH") is emitted per-minute and cross-anchored to Cloudflare R2, IPFS, a federation of independent mirrors, and the Base L2 chain via the IdentityLogAnchor contract. Protocol details are in [ALTER-STH].

Organ A broadcast surface for the envelope. The three organs are:

DNS publication (this document), the local alter-runtime L3 daemon, and the Patent-N HAD silicon quorum. The term is canonical; do not substitute "channel", "vector", or "emitter" at the architectural level.

Recognition The act of a resolver observing and verifying an envelope on cryptographic merit. Recognition is distinct from claim: publishing a TXT record is not a claim of identity, it is an observable assertion that the resolver may verify or reject. No field of this document carries a claim verb; resolvers recognise envelopes, they do not honour publisher assertions about them.

DNSSEC Validation The act of an authenticating DNS resolver verifying the RRSIG chain from the root trust anchor to the TXT RRset, per [RFC4033], [RFC4034], and [RFC4035], and setting the AD bit on the response delivered to the stub client.

DANE TLSA Pin A DNS TLSA resource record [RFC6698] binding a server’s leaf TLS certificate (or the public key therein) to the zone that hosts the envelope. In this document, the pin applies to the MCP endpoint at mcp.<zone>.

alter: URI A dispatch URI scheme provisionally registered with IANA per [RFC7595]. Full registration body and handler guidance are specified in [ALTER-DNS-PUB] Section 7; a normative cross-reference is given in Section 9 of this document.

(Terms from v02, Org-Identity Record, Identity Bootstrap, Canonical Entity Identifier, Regulatory Refusal Marker, are retained.)

3. Record Format: _mcp.<domain> (Service Discovery)

Section 3 of v01 of this document defines the _mcp.<domain> record, its ABNF grammar, field definitions (v, url, proto, pk, epoch, cap, attest, scope, priority, ttl, ext), forward-compatibility rules, and multi-string concatenation behaviour. These definitions are unchanged in this revision and are incorporated here by reference. Implementations MUST treat any existing _mcp.<domain> record as conformant to the v01 specification.

4. Record Format: _org-alter.<domain> (Identity Bootstrap)

Section 4 of v02 of this document defines the _org-alter.<domain> record, its ABNF grammar, field definitions (v, org, entity, entity-type, founded, regions, regulated, bootstrap, mcp-policy, epoch, pk, attest, ext), identity bootstrap procedure, and registry cross-checks. These definitions are unchanged in this revision and are incorporated here by reference. Implementations MUST treat any existing _org-alter.<domain> record as conformant to the v02 specification.

5. Record Format: _alter.<domain> (Envelope Publication)

This section defines the new Envelope Publication record introduced in v03. The record publishes the five load-bearing fields of the ALTER identity envelope (binding a ~handle to its Ed25519 public key, IdentityLog root, inception timestamp, revocation commitment, and detached signature) at an underscore-prefixed label under the handle's hosting zone. The full envelope JSON schema and wire format, including fields not carried over DNS (the implicit signature_alg constant and the optional caveats array), is pinned by [ALTER-DNS-PUB].

5.1. DNS Location

The Envelope Record is a DNS TXT resource record [RFC1035] published at the label _alter prepended to the hosting zone:

_alter.<zone>. IN TXT "<record-value>"

The underscore prefix conforms to the conventions established in [RFC8552] for globally scoped, underscore-prefixed DNS node names.

A zone MAY host more than one ~handle (one envelope per handle); in that case the zone MUST publish multiple TXT RRs at the same owner name. Resolvers MUST disambiguate returned records by the h= field and select the record matching the requested handle.

A domain MAY publish any combination of _mcp.<domain>, _org-alter.<domain>, and _alter.<domain> records independently (service-only, identity-only, envelope-only, or any intersection). The recommended pattern for an operator running an org-alter instance that hosts the operator's own principal handle (e.g. ~blake at truealter.com) is to publish all three.

5.2. ABNF Grammar

The record value is a semicolon-delimited sequence of key-value pairs. The following ABNF (per [RFC5234]) defines the syntax:

```
‘‘‘ alter-record = version ";" SP handle-field ";" SP pubkey-field ";" SP ilr-field ";" SP ts-field ";" SP rev-field ";" SP sig-field *( ";" SP unknown-field )

version = "v=alter1" handle-field = "h=" handle pubkey-field = "pk=" algo ":" base64url ilr-field = "ilr=" base64url ; SHA-256 of IdentityLog root ts-field = "ts=" 1*DIGIT ; inception_ts, Unix seconds rev-field = "rev=" base64url ; SHA-256 of revocation pre-image sig-field = "sig=" base64url ; Ed25519 detached signature unknown-field = token "=" *VCHAR

handle = "~" 1_( ALPHA / DIGIT / "-" / "_" ) [ "." "bot" ] / "~cc-" 1_( ALPHA / DIGIT / "-" / "." ) algo = "ed25519" base64url = 1_( ALPHA / DIGIT / "-" / "_" ) token = 1_( ALPHA / DIGIT / "-" / "_" ) ‘‘‘
```

The seven keys above are REQUIRED and MUST appear in the order shown. Publishers MUST NOT omit or reorder them. Resolvers MUST tolerate additional (unknown) fields appended after the seven required fields and MUST ignore them per the forward-compatibility rule (Section 6.4 below). Resolvers MUST tolerate arbitrary inter-field ordering on parse (publishers-emit-ordered, parsers- accept-unordered); canonicalisation for signature verification is specified in Section 6.3.

5.3. Field Definitions

5.3.1. v (REQUIRED)

Protocol version identifier. MUST be the literal string alter1. MUST appear as the first field in the record. Resolvers MUST reject any record whose v field is absent, is not the first field, or contains a value other than alter1.

The version namespace v=alter1 on _alter.<zone> is independent of the identically-named v=alter1 on _org-alter.<zone>. The two namespaces are disambiguated by the enclosing record label and MUST NOT be conflated. Future versions of either record may advance independently (e.g. _alter.<zone> may progress to v=alter2 while _org-alter.<zone> remains at v=alter1, or the reverse).

5.3.2. h (REQUIRED)

The Sovereign-, Bot-, or Instrument-tier ~handle to which the envelope binds. The leading tilde is mandatory. The h= value is the sole field resolvers MAY use to disambiguate multiple envelope TXT RRs sharing an owner name.

5.3.3. pk (REQUIRED)

An Ed25519 public key prefixed by its algorithm namespace and encoded in base64url without padding per [RFC4648] Section 5:

pk=ed25519:<base64url-no-pad-32-bytes>

Resolvers MUST reject records whose algorithm prefix is not ed25519 until a future revision registers additional algorithms. The pk value is the verification key for the detached signature in the sig field.

5.3.4. ilr (REQUIRED)

Base64url-no-pad SHA-256 digest of the IdentityLog root witnessed at envelope creation. Resolvers MUST cross-reference this value against the IdentityLog witness surface [ALTER-STH] to confirm the envelope was minted within a recognised tree state. Failure to cross-reference renders the envelope unverified.

5.3.5. ts (REQUIRED)

Envelope inception timestamp, expressed as decimal Unix seconds (integer). Resolvers MAY use this field to detect clock skew, evaluate caveat maturity, or reject envelopes with implausibly future inception.

5.3.6. rev (REQUIRED)

Base64url-no-pad SHA-256 digest of the revocation pre-image. Revocation is effected by revealing the pre-image to the IdentityLog; upon reveal, the envelope is considered revoked and MUST NOT be honoured by resolvers. The rev field is a forward-secure commitment: the pre-image is never published in DNS and is released only at revocation time to the log.

Publishers MUST NOT treat removal of the TXT record as revocation. Absence of a record is indistinguishable from misconfiguration; only pre-image reveal is load-bearing.

5.3.7. sig (REQUIRED)

Base64url-no-pad Ed25519 detached signature over the JCS-canonicalised envelope JSON with the signature field absent. Canonicalisation is specified by [RFC8785]. The signing input is the envelope JSON reconstructed from the parsed TXT fields plus the implicit constant signature_alg: "Ed25519" and an empty caveats array; caveats, when present, ride the HTTPS .well-known organ and do not appear in the DNS record. The canonical envelope schema, including JCS input construction, is pinned in [ALTER-DNS-PUB] Section 4.

5.3.8. Unknown fields

Fields not enumerated above MUST be ignored by v03 resolvers per the forward-compatibility rule (Section 6.4). Future revisions of this document MAY register additional envelope fields; such extensions MUST be distinguishable from private-use extensions by registration via the mechanism in Section 10.

5.4. Canonical Serialisation

The sig input is constructed as follows:

1. Parse the TXT RR character-strings into key-value pairs.

2. Construct the envelope JSON object:

   json { "handle": "<h>", "pubkey": "<pk>", "identitylog_root": "<ilr>", "inception_ts": <ts>, "revocation_hash": "<rev>", "signature_alg": "Ed25519", "caveats": [] }

   Values are typed per Section 6.2 (inception_ts as JSON integer; all other values as JSON strings; caveats as JSON array, empty unless a companion .well-known fetch supplies content). The signature field MUST be absent from the signing input.

3. Apply [RFC8785] JSON Canonicalisation Scheme to the object.

4. The resulting byte stream is the Ed25519 signing input.

Verification reverses this construction and checks the detached signature in the sig field against the derived byte stream.

Publishers MUST emit TXT fields in the order given in Section 6.2, but the DNS key=value ordering has no role in signature computation: the signed bytes are always the JCS serialisation of the JSON object.

5.5. Forward Compatibility

Resolvers MUST ignore unknown fields in the _alter.<domain> record. This rule, identical to the v01 _mcp and v02 _org-alter specifications, ensures that future extensions do not break existing implementations.

Publishers MUST NOT introduce new fields that repurpose or overload the seven required field names; new fields MUST use new names registered via the procedure in Section 10.

5.6. Multi-String Reassembly

Where the serialised envelope exceeds the 255-octet character-string limit of [RFC1035] Section 3.3.14, publishers MUST split at ; boundaries between complete key-value pairs. Splitting within a key-value pair is prohibited. Resolvers MUST concatenate the character-strings of a TXT RR in the order returned by the DNS library (i.e. the RR wire order) before parsing.

6. DNSSEC Requirement

The zone publishing an _alter.<domain> envelope record MUST be DNSSEC-signed per [RFC4033], [RFC4034], and [RFC4035]. Authoritative servers MUST respond with valid RRSIG coverage for the TXT RRset. Recursive resolvers handling queries for the envelope RRset MUST perform DNSSEC validation and MUST set the AD (Authenticated Data) bit on the response delivered to the stub client.

Stub clients (MCP clients, alter-runtime daemons, onboarding wizards, recognition verifiers) consuming _alter.<domain> envelope records MUST reject any response that lacks a set AD bit or that fails local RRSIG verification when operating in validating-stub mode. An envelope obtained over an unvalidated DNS path is not an envelope; it is unauthenticated TXT content. Treating it otherwise is a downgrade vulnerability (Section 11).

This requirement is specific to _alter.<domain> records. DNSSEC is RECOMMENDED but not REQUIRED for _mcp.<domain> and _org-alter.<domain> records in this revision, for backward compatibility with v01 and v02 deployments. Future revisions of this document MAY promote DNSSEC to REQUIRED for the other two records once deployment data justifies the promotion.

7. DANE TLSA Pin

The MCP endpoint associated with a published envelope MUST carry a DANE TLSA resource record [RFC6698] binding the endpoint's leaf TLS certificate or SubjectPublicKeyInfo to the zone. The TLSA record MUST be published at:

_443._tcp.mcp.<zone>. IN TLSA <usage> <selector> <matching-type> <cert-association-data>

Recommended parameters:

* *Usage field.* 3 (DANE-EE), pin the end-entity certificate directly, with no CA chain reliance. A publisher that explicitly requires CA-chain validation MAY use 1 (PKIX-EE) instead. Publishers MUST NOT use 0 (PKIX-TA) or 2 (DANE-TA) for the envelope organ; the trust basis of the envelope is the end-entity leaf.

* *Selector field.* 1 (SPKI), pin the SubjectPublicKeyInfo so that certificate rotations preserving the keypair do not invalidate the record. Selector 0 (full certificate) MAY be used but requires more frequent TLSA republication.

* *Matching-type field.* 1 (SHA-256). Matching type 2 (SHA-512) is reserved for future revisions.

Clients establishing an MCP session at https://mcp.<zone>/ in conjunction with a resolved envelope MUST fetch and validate the TLSA record, MUST abort the TLS handshake on mismatch, and MUST NOT fall back to PKIX-only validation on TLSA failure.

The TLSA requirement is scoped to envelopes whose MCP session establishment is triggered by the resolved envelope (i.e. when the envelope resolution and the subsequent MCP session are part of a single recognition transaction). MCP clients that do not resolve an envelope (e.g. v01-only clients consuming only _mcp.<domain>) are out of scope for this requirement and continue to operate under v01 rules.

8. IdentityLog Cross-Reference

The ilr= field of the _alter.<domain> record carries a base64url-no-pad SHA-256 digest of an IdentityLog Signed Tree Head (STH) root witnessed at envelope creation. The IdentityLog protocol (leaf hashing, Merkle-tree construction, STH cadence, witness federation, Cloudflare R2 / IPFS / Base L2 anchor path) is specified in full by [ALTER-STH]; this document does not duplicate that specification.

Resolvers verifying an envelope from DNS MUST cross-reference the ilr= value against at least one IdentityLog witness surface (federation mirror, R2 canonical read, IPFS content address, or Base L2 IdentityLogAnchor contract). The specific surface is a matter of deployment preference; [ALTER-STH] Section 6 gives conforming client profiles. Failure to cross-reference renders the envelope unverified and it MUST NOT be admitted to any recognition-gated decision.

The revocation check also crosses the IdentityLog: a resolver MUST consult the IdentityLog revocation-witness surface for any reveal whose SHA-256 equals the rev= field of the envelope. If a matching pre-image has been revealed, the envelope is revoked and MUST NOT be honoured regardless of the freshness of the TXT RRset.

9. alter: URI Scheme Cross-Reference

An IANA-registered URI scheme alter: provides a dispatchable surface for ~handle references: operating-system URI handlers (xdg-mime, LSHandlers, Windows registry, Android intent-filter) invoke a resolver that retrieves and verifies the envelope through the organ chain defined in [ALTER-DNS-PUB] (DNS first, with fallback to the HTTPS .well-known surface and, where available, the local alter-runtime L3 daemon).

Registration is provisional per [RFC7595] Section 3. The full registration body, scheme syntax, semantics, encoding considerations, interoperability and security considerations, author and change controller, is published in [ALTER-DNS-PUB] Section 7 and submitted to IANA separately from this document. The IANA request is in progress as of the publication date of this revision; the final registration reference will be substituted when available.

Handlers invoked via alter: URIs MUST perform full envelope verification, DNSSEC validation (Section 7), DANE TLSA binding (Section 8) when establishing any HTTPS session, IdentityLog cross-reference (Section 9), and the eleven-step verification algorithm of [ALTER-DNS-PUB] Section 8, before acting on any content or directive derived from the envelope.

10. Discovery and Bootstrap Procedures

10.1. Discovery Procedure: _mcp.<domain>

The discovery procedure defined in Section 4 of v01 is unchanged in this revision. Clients querying _mcp.<domain> follow the v01 algorithm exactly.

10.2. Identity Bootstrap Procedure: _org-alter.<domain>

The identity bootstrap procedure defined in Section 6 of v02 is unchanged in this revision. Onboarding wizards reading _org-alter.<domain> follow the v02 algorithm exactly.

10.3. Envelope Recognition Procedure: _alter.<domain>

Given an alter:~<handle> reference, a zone hint, or a raw _alter.<zone>. query, a resolver MUST execute the following steps in order. Any failure terminates recognition and the envelope MUST be treated as unverified.

1. *Query.* Issue a DNS TXT query for _alter.<zone>.. Use DoH or DoT in preference to UDP/53 where operationally feasible.

2. *DNSSEC validation.* Validate the RRSIG chain from the root trust anchor to the TXT RRset (Section 7). Confirm the AD bit on the response when relying on an upstream validating resolver, or locally RRSIG-validate in validating-stub mode. On failure, abort.

3. *Chunk reassembly.* Concatenate character-strings in RR order; parse ;-separated key-value pairs.

4. *Handle disambiguation.* Select the record whose h= field matches the requested ~handle. If no record matches, abort.

5. *Field extraction.* Confirm presence of the seven required fields (v, h, pk, ilr, ts, rev, sig). Reject any record missing any required field, or whose v is not alter1.

6. *Envelope reconstruction.* Build the envelope JSON per Section 6.3, inserting the implicit signature_alg: "Ed25519" constant and an empty caveats array.

7. *JCS canonicalisation.* Apply [RFC8785] JCS to the envelope with the signature field absent.

8. *Ed25519 verification.* Verify the detached sig over the JCS byte stream using the public key in pk. On failure, abort.

9. *IdentityLog cross-reference.* Confirm ilr= corresponds to a STH recognised in the IdentityLog witness set at or after ts (Section 9; [ALTER-STH]). On failure, abort.

10. *DANE TLSA validation.* When establishing an MCP session at mcp.<zone> as part of the same recognition transaction, fetch the TLSA record at _443._tcp.mcp.<zone>. and gate the TLS handshake on the binding (Section 8). On mismatch, abort.

11. *Caveats evaluation.* Fetch the HTTPS .well-known companion surface and evaluate any caveats on the envelope per [ALTER-DNS-PUB]. Caveats that cannot be satisfied bound the subsequent use of the envelope but are not grounds to abort recognition.

12. *Revocation check.* Consult the IdentityLog revocation- witness surface. If a pre-image whose SHA-256 equals rev= has been revealed, the envelope is revoked; abort.

Only after all twelve steps succeed is the envelope considered verified. A verified envelope is the sole admissible input to a recognition-over-qualification gate; unverified envelopes MUST be refused upstream of any authorisation or trust decision.

The twelve-step procedure above is the IETF-surface summary of the eleven-step verification algorithm in [ALTER-DNS-PUB] Section 8; they differ only in that this document splits caveats and revocation into distinct steps for clarity.

11. Caching

Caching of _mcp.<domain> records follows v01. Caching of _org-alter.<domain> records follows v02.

_alter.<domain> records SHOULD be cached for the duration of the DNS TTL. Resolvers MUST NOT serve stale envelope TXT past the RRset TTL unless they are themselves validating caches and can re-confirm RRSIG coverage on each serve. Recognition verifiers MAY cache successful verification results locally for a short interval (bounded above by the RRset TTL or 3600 seconds, whichever is smaller) to amortise the cost of repeated JCS and Ed25519 operations, but MUST re-run the revocation check (Section 11.12) on each recognition event, not on each cache refresh.

12. Security Considerations

(Security considerations from v01 and v02 are retained. Additional considerations introduced by the envelope layer are below.)

12.1. DNSSEC Downgrade

The mandatory DNSSEC requirement in Section 7 is the primary defence against on-path manipulation of envelope TXT content. An attacker who can inject unsigned responses, e.g. via a compromised resolver or a DNS middlebox that strips RRSIG, would otherwise be able to substitute an attacker-controlled envelope at the resolver boundary. Stub clients MUST reject any response lacking AD or failing local RRSIG verification. Operators MUST NOT downgrade the _alter. RRset to unsigned during KSK/ZSK rollover (see [RFC6781] for best-current practice on rollover).

12.2. TLSA Pin Rotation

The DANE TLSA requirement in Section 8 binds the MCP endpoint’s TLS leaf to a specific hash. Operators rotating certificates MUST publish the new TLSA record before the new certificate is activated on the live listener, with a grace window of at least twice the TLSA RRset TTL. Selector 1 (SPKI) survives rotations that preserve the keypair; selector 0 requires republication on every rotation. Loss of the TLS private key forces revocation via the revocation_hash reveal path (Section 9) rather than silent cert replacement.

12.3. Envelope Substitution

An attacker in control of a domain’s DNS can publish an arbitrary envelope for any ~handle claimed to be hosted under that zone. The three structural defences are:

1. *IdentityLog witness.* The ilr= cross-reference constrains the envelope to STHs witnessed by the IdentityLog federation; substitution of a locally-minted envelope that has not been witnessed will fail Section 11.9. An attacker who wishes to substitute must also corrupt at least one IdentityLog mirror, which is a detectable equivocation per [RFC9162] design.

2. *Ed25519 signature.* The detached signature binds the envelope to a specific Ed25519 key. An attacker who does not hold the private key cannot forge a valid sig. An attacker who does hold the private key has already compromised the handle; the revocation path (Section 9) is the residual mitigation.

3. *DNSSEC.* Section 7 prevents tampering with the TXT RRset in transit. This does not prevent a malicious zone operator from publishing a malicious envelope, that attack is caught at (1) and (2), but it prevents third-party substitution.

12.4. Revocation Opacity

Revocation is effected by revealing the pre-image to the IdentityLog, not by removing the TXT record. Absence of a record is indistinguishable from misconfiguration; resolvers MUST NOT treat absence as revocation. This design is deliberate: a zone briefly unreachable (DNS outage, registrar incident, tooling error) must not accidentally become a revocation event.

The cost is that a compromised zone may continue to serve a valid (but intended-to-be-revoked) envelope until the rightful handle-holder reveals the pre-image. Pre-image reveal is a low-friction operation, a single authenticated POST to any IdentityLog mirror, but it requires the rightful holder to act. Handle-holders SHOULD establish a pre-committed revocation reveal procedure at mint time.

12.5. Clock Skew and ts=

The ts= inception timestamp is advisory: resolvers MAY use it to detect implausibly future envelopes (e.g. minted more than a few hundred seconds after current wall time) but MUST NOT rely on local clock for security-critical decisions. The authoritative ordering anchor is the IdentityLog STH tree position, not the inception timestamp.

12.6. Cross-Record Key Consistency

When all three records (_mcp, _org-alter, _alter) are published under the same zone and each carries a pk field, the values MUST be evaluated for consistency. The _mcp.pk and _org-alter.pk fields are v01/v02 service and organisational keys respectively; the _alter.pk field is the Sovereign-tier envelope key. These are structurally distinct purposes, and the keys MAY differ. However, where a zone operator deliberately binds all three to the same Ed25519 key (a common pattern for a single-operator deployment), a mismatch across records indicates either rotation-in-progress or compromise; resolvers SHOULD surface the discrepancy.

12.7. Passive-Stream Coupling

The _alter.<domain> record carries only the five load-bearing envelope fields and the protocol version. No inferred trait, no passive-stream derivative, and no provenance-tagged attribute rides this record. This is a structural property, not a recommendation: the ABNF of Section 6.2 enumerates every field the resolver accepts, and the forward-compatibility rule only permits future named extensions, not arbitrary attribute carriage. The privacy implications of passive inference are addressed at the envelope

semantic layer [ALTER-DNS-PUB] and its caveats surface, not in DNS.

13. Privacy Considerations

(Privacy considerations from v01 and v02 are retained. Additional considerations introduced by the envelope layer are below.)

13.1. Public Handle Disclosure

Publishing _alter.<domain> exposes the bound ~handle, its Ed25519 public key, its IdentityLog root, its inception timestamp, and its revocation-hash commitment to any DNS observer. For a Sovereign-tier handle this is by design: the envelope is intended to be publicly verifiable. Handle-holders who require concealment MUST NOT publish an _alter.<domain> record; alternative organs (the local alter-runtime daemon for local-only recognition, or the Patent-N HAD silicon quorum for device-local presence proof) support recognition without DNS publication.

13.2. DNS Query Metadata

A resolver querying _alter.example.com reveals to its recursive resolver that it intends to verify the envelope hosted under that zone. Query metadata privacy is addressed at the transport layer: clients SHOULD prefer DoH ([RFC8484]) or DoT ([RFC7858]) over UDP/53 where operationally feasible. This consideration is identical to v01 / v02 and is repeated here for emphasis given the greater individual-identity sensitivity of the envelope surface.

13.3. Revocation Unlinkability

The rev= field is the SHA-256 of a secret pre-image; publishing it does not disclose the pre-image. An observer cannot predict the pre-image or link it back to any identifier. Reveal at revocation time links the pre-image to the envelope, but only at the moment of revocation, not during the envelope’s active lifetime.

14. IANA Considerations

14.1. Underscored DNS Node Name Registration

This document requests IANA to update the entries in the "Underscored and Globally Scoped DNS Node Names" registry established by [RFC8552] to reflect v03:

RR Type	_NODE NAME	Reference
TXT		

_mcp	[this document], v01 Sec.3		TXT	_org-alter	[this document], v02 Sec.4		TXT	_alter	[this document], v03 Sec.6

The _alter label is used to publish envelope records as defined in Section 6 of this document. Formal registration of _alter in the RFC 8552 registry is proposed on Standards Action maturation of this draft; during the Internet-Draft phase, the label operates under the provisional-use convention established by _dmarc, _mta-sts, _mcp (this draft), and _org-alter (this draft).

14.2. alter: URI Scheme Registration

This document cross-references the provisional URI scheme registration of alter: per [RFC7595] Section 3. The full registration body is published in [ALTER-DNS-PUB] Section 7 and is submitted to IANA separately. This document does not duplicate the registration body; it refers to the sibling specification and notes that recognition verifiers invoked via alter: URIs MUST follow Section 11.12 of this document for envelope verification.

14.3. Envelope Version Registry

This document defines the version tag v=alter1 for the _alter.<domain> record, independent of the identically-named tag on the _org-alter.<domain> record. Future versions (v=alter2 and beyond) SHOULD be coordinated with the ALTER implementation community and documented in successor revisions of this draft. Until a formal IETF working group is chartered for identity- envelope DNS publication, the authors maintain the version namespace.

14.4. Org-Alter Version Registry (unchanged from v02)

The version tag v=alter1 for the _org-alter.<domain> record is preserved from v02. No changes are requested in this revision.

14.5. Registry Namespace Registry (unchanged from v02)

The initial set of entity field registry namespaces (abn, acn, ein, ch, cro, lei) defined in v02 is preserved unchanged.

14.6. Framework Token Registry (unchanged from v02)

The initial set of regulated framework tokens (disp, itar, ear, hipaa, gdpr, soc2, iso27001, iso42001, essential8, aprs) defined in v02 is preserved unchanged.

14.7. Signature Algorithm Registry

This document defines the initial pk= and sig= algorithm namespace ed25519 for the _alter.<domain> record. Future algorithms (e.g. ed448, ml-dsa-65) MAY be registered by successor documents. Resolvers MUST reject records whose algorithm prefix is not registered at the resolver's protocol version.

15. Examples

This section provides non-normative examples of Envelope Records for common deployment scenarios.

15.1. Minimal Envelope for a Single Handle

A zone hosting a single Sovereign-tier handle publishes its envelope at _alter.<zone>.:

_alter.truealter.com. 3600 IN TXT ("v=alter1; h=~blake; " "pk=ed25519:<EXAMPLE-pubkey-32B-base64url>; " "ilr=<EXAMPLE-sth-root-32B-base64url>; " "ts=1729123456; " "rev=<EXAMPLE-revocation-hash-32B-base64url>; " "sig=<EXAMPLE-ed25519-signature-64B-base64url>")

All base64url values in this example are illustrative. Production values are the Ed25519 public key, SHA-256 digests, and 64-byte detached signature encoded per [RFC4648] Section 5 without padding.

15.2. Zone Hosting Multiple Handles

A zone hosting more than one handle publishes multiple envelope TXT RRs at the same owner name. Resolvers disambiguate by the h= field:

_alter.example.org. 3600 IN TXT "v=alter1; h=~alice; pk=ed25519:..."
_alter.example.org. 3600 IN TXT "v=alter1; h=~bob; pk=ed25519:..."
_alter.example.org. 3600 IN TXT "v=alter1; h=~carol.bot; pk=ed25519:..."

A resolver asked to verify ~bob at example.org selects the second RR.

15.3. Full ALTER Zone (All Three Records)

A zone operator running an org-alter instance for their own principal handle publishes all three records:

_mcp.truealter.com. IN TXT "v=mcp1; url=https://mcp.truealter.com/ ..." _org-alter.truealter.com. IN TXT "v=alter1; org=Alter Meridian Pty Ltd; ..." _alter.truealter.com. IN TXT "v=alter1; h=~blake; pk=ed25519:...; ilr=...; ts=...; rev=...; sig=..." _443._tcp.mcp.truealter.com. IN TLSA 3 1 1 <sha256-of-spki>

Together these expose: the MCP service endpoint and its capabilities (_mcp); the legal entity, regulatory posture, and jurisdictional regions (_org-alter); the Sovereign-tier envelope for ~blake (_alter); and the DANE TLSA pin on the MCP endpoint. A resolver may consume any subset according to its recognition requirement.

15.4. Instrument-Tier Handle

An AI instrument handle uses the ~cc- prefix:

_alter.truealter.com. 3600 IN TXT ("v=alter1; h=~cc-opus-4-7; " "pk=ed25519:...; ilr=...; ts=...; rev=...; sig=...")

Instrument-tier envelopes are bound to a specific model version. Rotation of the model version produces a new ~cc- handle with a new envelope; the prior envelope remains verifiable over its active lifetime and is revoked by the IdentityLog reveal path when the model is retired.

16. Interoperability with v01 and v02

A domain that publishes only a v01 _mcp.<domain> record continues to work with all v01, v02, and v03 clients.

A domain that publishes _mcp.<domain> and _org-alter.<domain> (v02) continues to work with v02 and v03 clients unchanged. v03 clients gain the ability to additionally query _alter.<domain> and gracefully handle its absence.

A domain that publishes all three records benefits from:

* Service discovery via _mcp.<domain> (v01).
* Organisational identity bootstrap via _org-alter.<domain> (v02).
* Individual identity recognition via _alter.<domain> (v03).
* DNSSEC-authenticated envelope delivery (Section 7).
* DANE TLSA binding on the MCP endpoint (Section 8).
* IdentityLog-anchored envelope lifecycle (Section 9).

A domain that publishes only _alter.<domain> (envelope-only, no MCP server, no organisational record) is permitted. This is the appropriate configuration for a Sovereign-tier individual who wishes to be recognisable under their own zone without operating an MCP server endpoint or declaring an organisational identity.

The three records are orthogonal along their semantic axes but share the zone's DNSSEC trust root. A v03-compliant resolver that successfully resolves any subset of the three records treats each resolution as independent and does not fail the resolution of one record because another is absent or malformed.

17. Implementation Status

This section records the status of known implementations at the time of publication, per [RFC7942].

ALTER (https://truealter.com) maintains a reference implementation of all three records:

* _mcp.truealter.com exercising the v01 field set including pk, epoch, attest, and ext.

* _org-alter.truealter.com exercising the v02 field set including entity, regulated, mcp-policy, and bootstrap.

* _alter.truealter.com exercising the v03 envelope field set (h, pk, ilr, ts, rev, sig) for the ~blake handle.

* _443._tcp.mcp.truealter.com publishing a DANE-EE / SPKI / SHA-256 TLSA pin on the MCP endpoint leaf.

* An IdentityLog STH anchor federation with four independent witness surfaces (Cloudflare R2 canonical, IPFS content- addressed, two federation mirrors, Base L2 IdentityLogAnchor contract).

* A standalone L0 discovery and recognition library (alter_discover) that resolves all three records, validates DNSSEC locally, fetches and verifies the DANE TLSA pin, cross- references the IdentityLog witness surface, and produces a structured recognition report.

* An alter: URI handler registered via xdg-mime on Linux/BSD and the corresponding platform registration paths on macOS, Windows, iOS, and Android.

* An onboarding wizard tool (mcp-org-alter onboard) that bootstraps a new org-alter instance from the published records with no manual data entry beyond the domain name.

The reference implementation targets MCP specification version 2025-11, uses streamable-http as the default transport, and treats DNSSEC + DANE + IdentityLog as a mandatory verification chain for any _alter.<domain>-derived recognition.

18. References

18.1. Normative References

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <https://www.rfc-editor.org/info/rfc1035>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <https://www.rfc-editor.org/info/rfc4033>.

[RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <https://www.rfc-editor.org/info/rfc4034>.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005, <https://www.rfc-editor.org/info/rfc4035>.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <https://www.rfc-editor.org/info/rfc4648>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <https://www.rfc-editor.org/info/rfc5234>.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <https://www.rfc-editor.org/info/rfc6698>.

[RFC7208] Kitterman, S., "Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1", RFC 7208, DOI 10.17487/RFC7208, April 2014, <https://www.rfc-editor.org/info/rfc7208>.

[RFC7595] Thaler, D., Ed., Hansen, T., and T. Hardie, "Guidelines and Registration Procedures for URI Schemes", BCP 35, RFC 7595, DOI 10.17487/RFC7595, June 2015, <https://www.rfc-editor.org/info/rfc7595>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8461] Margolis, D., Risher, M., Ramakrishnan, B., Brotman, A., and J. Jones, "SMTP MTA Strict Transport Security (MTA-STS)", RFC 8461, DOI 10.17487/RFC8461, September 2018, <https://www.rfc-editor.org/info/rfc8461>.

[RFC8552] Crocker, D., "Scoped Interpretation of DNS Resource Records through "Underscored" Naming of Attribute Leaves", BCP 222, RFC 8552, DOI 10.17487/RFC8552, March 2019, <https://www.rfc-editor.org/info/rfc8552>.

[RFC8785] Rundgren, A., Jordan, B., and S. Erdtman, "JSON Canonicalization Scheme (JCS)", RFC 8785, DOI 10.17487/RFC8785, June 2020, <https://www.rfc-editor.org/info/rfc8785>.

[MCP] Agentic AI Foundation, "Model Context Protocol Specification", 2026, <https://modelcontextprotocol.io>.

18.2. Informative References

[RFC6376] Crocker, D., Ed., Hansen, T., Ed., and M. Kucherawy, Ed., "DomainKeys Identified Mail (DKIM) Signatures", STD 76, RFC 6376, DOI 10.17487/RFC6376, September 2011, <https://www.rfc-editor.org/info/rfc6376>.

[RFC6781] Kolkman, O., Mekking, W., and R. Gieben, "DNSSEC Operational Practices, Version 2", RFC 6781, DOI 10.17487/RFC6781, December 2012, <https://www.rfc-editor.org/info/rfc6781>.

[RFC7489] Kucherawy, M., Ed. and E. Zwicky, Ed., "Domain-based Message Authentication, Reporting, and Conformance (DMARC)", RFC 7489, DOI 10.17487/RFC7489, March 2015, <https://www.rfc-editor.org/info/rfc7489>.

[RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <https://www.rfc-editor.org/info/rfc7858>.

[RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", BCP 205, RFC 7942, DOI 10.17487/RFC7942, July 2016, <https://www.rfc-editor.org/info/rfc7942>.

[RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", RFC 8484, DOI 10.17487/RFC8484, October 2018, <https://www.rfc-editor.org/info/rfc8484>.

[RFC9162] Laurie, B., Messeri, E., and R. Stradling, "Certificate Transparency Version 2.0", RFC 9162, DOI 10.17487/RFC9162, December 2021, <https://www.rfc-editor.org/info/rfc9162>.

[MORRISON-IFT] Morrison, B., "Identity Field Theory: Toward a Physics of Being Known", 2026, <https://doi.org/10.6084/m9.figshare.31951383>.

[ALTER-DNS-PUB] Morrison, B., "ALTER DNS Publication, v1", 2026, <https://truealter.com/docs/protocol/alter-dns-publication-v1>.

[ALTER-STH] Morrison, B., "IdentityLog STH Anchor, v1", 2026, <https://truealter.com/docs/protocol/identitylog-sth-anchor-v1>.

Appendix A. Recognition Pseudocode

The following pseudocode illustrates the combined recognition procedure defined in Section 11.3. It is non-normative; the normative procedure is the twelve-step algorithm in the body of the document and the eleven-step algorithm in [ALTER-DNS-PUB] Section 8 to which it cross-references.

```
''' function recognise_envelope(handle, zone): # Step 1-2: Query +
DNSSEC response = dns_query("_alter." + zone, type=TXT, prefer=DoH)
if not response.ad_bit and not local_rrsig_validate(response): raise
UnauthenticatedResponse
```

```
   # Step 3-5: Chunk reassembly + handle disambiguation + fields
   records = [parse_alter_record(rr) for rr in response.rrset]
   record = find(records, lambda r: r.h == handle)
   if record is None or record.v != "alter1":
       raise RecordNotFound
   for f in ["h", "pk", "ilr", "ts", "rev", "sig"]:
       if not hasattr(record, f):
           raise MalformedRecord

   # Step 6-7: Envelope reconstruction + JCS
   envelope = {
       "handle": record.h,
       "pubkey": record.pk,
       "identitylog_root": record.ilr,
       "inception_ts": int(record.ts),
       "revocation_hash": record.rev,
       "signature_alg": "Ed25519",
       "caveats": [],
   }
   signing_input = jcs_canonicalise(envelope)

   # Step 8: Ed25519 verify
   if not ed25519_verify(record.pk, record.sig, signing_input):
       raise SignatureInvalid

   # Step 9: IdentityLog cross-ref
   if not identitylog_witness_contains(record.ilr, record.ts):
       raise WitnessMissing

   # Step 10: DANE TLSA (if establishing MCP session)
   if establishing_mcp_session(zone):
       tlsa = dns_query("_443._tcp.mcp." + zone, type=TLSA)
       if not tlsa_matches_endpoint(tlsa, "mcp." + zone):
           raise TLSAFailure

   # Step 11: Caveats (advisory only; bounds subsequent use)
   caveats = fetch_well_known_caveats(zone, handle)

   # Step 12: Revocation
   if identitylog_revocation_revealed(record.rev):
       raise EnvelopeRevoked

   return VerifiedEnvelope(record, caveats) '''
```

Appendix B. Document History

draft-morrison-mcp-dns-discovery-03 (April 2026):

* Adds the _alter.<domain> Envelope Record (Section 6).

* Defines v, h, pk, ilr, ts, rev, sig fields for the new record, mirroring the canonical envelope fragment pinned by [ALTER-DNS-PUB].

* Introduces a mandatory DNSSEC validation requirement for _alter.<domain> responses (Section 7).

* Introduces a mandatory DANE TLSA [RFC6698] pin on the MCP endpoint (Section 8) for envelope-triggered MCP sessions.

* Adds the IdentityLog STH cross-reference requirement (Section 9), pointing at [ALTER-STH] for the log protocol specification.

* Adds a provisional alter: URI scheme cross-reference per [RFC7595] (Section 10).

* Adds the envelope recognition procedure (Section 11.3), a twelve-step algorithm that cross-refers to the eleven-step algorithm of [ALTER-DNS-PUB] Section 8.

* Adds IANA registration for _alter underscore-prefixed label (Section 13.1) and the independent v=alter1 envelope version namespace.

* Adds a Signature Algorithm Registry (Section 13.8) with initial value ed25519.

* Adds Security Considerations for DNSSEC downgrade, TLSA pin rotation, envelope substitution, revocation opacity, clock skew, cross-record key consistency, and passive-stream coupling.

* Adds Privacy Considerations for public handle disclosure, DNS query metadata, and revocation unlinkability.

* Adds Examples for minimal envelope, multi-handle zone, full ALTER zone with all three records, and Instrument-tier handle.

* Adds Implementation Status entry for the envelope reference implementation.

* v01 _mcp.<domain> and v02 _org-alter.<domain> record specifications are incorporated by reference and remain unchanged.

draft-morrison-mcp-dns-discovery-02 (April 2026):

* Adds the _org-alter.<domain> Org-Identity Record.

* Defines org, entity, entity-type, founded, regions, regulated, bootstrap, mcp-policy, epoch, pk, attest, ext fields for the organisational record.

* Adds the Identity Bootstrap procedure.

* Adds IANA registration for _org-alter underscore-prefixed label.

* Adds version tag v=alter1 (org-alter namespace) and registry namespace and framework token registries.

* Adds Examples for minimal, full, regulated (DISP), and multi-regulator deployments.

* Adds Implementation Status entry for the orgalter_discover reference library.

* v01 _mcp.<domain> record specification is incorporated by reference and remains unchanged.

draft-morrison-mcp-dns-discovery-01 (April 2026):

* Adds Identity Field Theory grounding for epoch and scope.

* Refines security considerations for identity assurance decay.

* Refines privacy considerations for scope as a privacy boundary.

* Adds Coexistence section with SEP-1959, AID, A2A.

* Adds Implementation Status section.

draft-morrison-mcp-dns-discovery-00 (April 2026):

* Initial submission.

* Defines _mcp.<domain> TXT record format with ABNF grammar.

* Defines discovery procedure with HTTPS fallback.

* Defines pk, epoch, attest, scope, cap, priority, ttl, and ext fields.

* Registers _mcp in the underscored DNS node name registry.

Author’s Address

Blake Morrison
Alter Meridian Pty Ltd
Australia
Email: blake@truealter.com