Mplify Standard

Mplify 185

AI Agent Legal Identity Standard

July 2026

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Disclaimer

© Mplify Alliance 2026. All Rights Reserved.

The information in this publication is freely available for reproduction and use by any recipient and is believed to be accurate as of its publication date. Such information is subject to change without notice and Mplify Alliance (Mplify) is not responsible for any errors. Mplify does not assume responsibility to update or correct any information in this publication. No representation or warranty, expressed or implied, is made by Mplify concerning the completeness, accuracy, or applicability of any information contained herein and no liability of any kind shall be assumed by Mplify as a result of reliance upon such information.

The information contained herein is intended to be used without modification by the recipient or user of this document. Mplify is not responsible or liable for any modifications to this document made by any other party.

The receipt or any use of this document or its contents does not in any way create, by implication or otherwise:

a) any express or implied license or right to or under any patent, copyright, trademark or trade secret rights held or claimed by any Mplify member which are or may be associated with the ideas, techniques, concepts or expressions contained herein; nor
b) any warranty or representation that any Mplify members will announce any product(s) and/or service(s) related thereto, or if such announcements are made, that such announced product(s) and/or service(s) embody any or all of the ideas, technologies, or concepts contained herein; nor
c) any form of relationship between any Mplify member and the recipient or user of this document.

Implementation or use of specific Mplify standards, specifications, or recommendations will be voluntary, and no Member shall be obliged to implement them by virtue of participation in Mplify Alliance. Mplify is a non-profit international organization to enable the development and worldwide adoption of agile, assured and orchestrated network services. Mplify does not, expressly or otherwise, endorse or promote any specific products or services.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Contents

1. Purpose and Scope 5

1.1 Purpose 5
1.2 Scope 5
1.3 Target Industry Verticals 6
1.4 Relationship to Existing Standards 6
1.5 Operational and Economic Value Summary 6
1.6 Deployment Profiles and Incremental Adoption Guidance 7

2. Normative References 7

2.1 Informative References 7

3. Definitions and Terminology 8

3.1 Core Definitions 8
3.2 Role Definitions 10
3.3 Credential Type Taxonomy 10
3.4 Trust-Domain Concepts 11
3.4.1 Prevention of Context Drift 11

4. Agent Identity Model 13

4.1 Design Principles 13
4.2 Identity Structure 16
4.3 DID Document Requirements 16
4.4 Agent Type Classification 17
4.5 Cross-Federation Trust Model 17
4.6 Governing Authority Qualification 19
4.7 Key Storage Requirements 19
4.8 Open Agent Interconnection (OAI) Model 20
4.9 Sentinel Functional Responsibilities 21

5. Agent Identity Lifecycle 21

5.1 Lifecycle Overview 21
5.2 State Definitions 22
5.3 Bootstrapping Protocol 22
5.4 Key Rotation 23
5.5 Decommissioning 23

6. Credential Architecture 24

6.1 Credential Design Principles 24
6.2 Core Credential Types 24
6.2.1 NetworkAuthorityCredential 24
6.2.2 Domain-Specific Operational Credentials 24
6.3 Credential Verification Protocol 25
6.4 Revocation Infrastructure 26
6.4.1 Revocation Authority and Status-List Governance 27
6.4.2 Revocation Resilience 27
6.5 Credential Composition Patterns 27

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

6.5.1 Delegation Depth Limits 27
6.5.2 Multi-Credential Presentations 28
6.5.3 Conditional Credentials 28
6.6 Human Approval and Override Patterns 28
7. Privacy and Selective Disclosure 29
7.1 Privacy Principles 29
7.2 Selective Disclosure Mechanism 29
7.3 Privacy Limitations 29
8. Regulatory Alignment 30
8.1 Alignment Framework 30
8.2 European Union Artificial Intelligence Act (EU AI Act, Regulation 2024/1689) 30
8.3 NIST AI Risk Management Framework (AI RMF 1.0, NIST AI 100-1) 30
8.4 Digital Networks Act — DNA 30
8.5 SOC 2 Type II Alignment 31
8.6 Legal Enforceability Considerations 32
9. Conformance Requirements 33
9.1 Conformance Levels 33
9.2 Mandatory Requirements Summary 33
10. List of contributing Memebers 34
Appendix A -- Glossary Cross-Reference 36
Appendix B -- Glossary Cross-Reference 36
B.1 Key Storage Requirements 36
B.2 Open Agent Interconnection (OAI) Model 37
B.3 Human Approval and Override Patterns 37

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

1. Purpose and Scope

1.1 Purpose

This specification defines a standard framework for establishing, managing, and verifying the identity of software agents that operate within federated legal and commercial workflows. It addresses a fundamental question in the era of artificial intelligence: **how does a machine participant prove who it is, what it is authorized to do, and who is accountable for its actions?**
The standard is motivated by the operational reality that software agents are increasingly performing tasks that have legal and commercial consequences -- executing transactions, accessing protected resources, presenting authorizations, and making decisions on behalf of human principals. These activities require a trust infrastructure that is at least as rigorous as the one applied to human participants but adapted for the distinct characteristics of machine actors: they operate continuously, can be instantiated and retired programmatically, may act across organizational boundaries, and must be governed at scale.
This specification provides that trust infrastructure through three interlocking capabilities:

1. **Verifiable Identity** -- Each agent possesses a globally unique, cryptographically verifiable digital identity that is issued by and bound to a recognized governing authority.
2. **Scoped Authorization** -- Each agent carries machine-readable credentials that define precisely what it is authorized to do, issued by the appropriate authority and verifiable by any relying party.
3. **Accountable Governance** -- Each agent's identity and credentials are subject to a complete lifecycle -- creation, use, rotation, suspension, revocation, and retirement -- managed by its governing authority.
4. **Legal Status Clarification:** An Agent Identity under this specification is a technical identity and authorization construct only. It does not confer legal personhood, legal capacity, rights, duties, or independent liability on an Agent. All legal rights, obligations, and liability remain with the relevant natural person or legal entity, or with the parties contractually responsible for the Agent, as determined by applicable law and contract. This document does, however, allow a legal venue in a jurisdiction such as a court of law establish a point of irrevocable liability for an agent action.

1.2 Scope

This specification applies to software agents operating within federated environments, with particular relevance to agents that participate in legal authorization workflows. An Agent may operate autonomously or under human supervision; both modes are within scope.
In Scope:

- Agent identity creation, issuance, and binding to a governing authority
- Agent credential issuance, presentation, and verification
- Agent identity lifecycle management
- Privacy-preserving agent authentication using selective disclosure
- Real-time credential revocation and its verification
- Mapping of agent identity to regulatory accountability requirements
- Cross-federation trust establishment between governing authorities

Out of Scope:

- Internal implementation details of agent decision-making logic
- Human user identity management

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

- Specific cryptographic algorithm selections
- Network transport security

1.3 Target Industry Verticals

While the core identity model is industry-neutral, this specification has been developed with the following initial verticals in mind:

Vertical	Representative Use Case
Telecommunications	Automated cross-connect ordering via electronic Letters of Authorization (eLOA) between carrier agents
Data Center Operations	Agent-to-agent verification of cabinet access, power provisioning, and interconnection rights
Legal Services	Delegation of signing authority for contracts, NDAs, and regulatory filings through credentialed agents

The framework is extensible to any domain where software agents act on behalf of organizations in federated, multi-party workflows requiring verifiable authorization and accountability.

1.4 Relationship to Existing Standards

This specification builds upon and is intended to be used in conjunction with the following standards families:

- **W3C Decentralized Identifiers (DIDs)** -- for the identifier format and resolution model
- **W3C Verifiable Credentials (VCs)** -- for the credential format and proof model
- **DIDComm v2** -- for authenticated agent-to-agent communication
- **Mplify Alliance LSO Sonata** -- for lifecycle service interfaces

This specification does not replace these standards. Rather, it provides a normative profile that specifies how these standards are applied to the particular requirements of AI agent identity in legal and commercial contexts.

1.5 Operational and Economic Value Summary

This specification should be read not only as a security and compliance profile, but also as an operational automation layer for high-friction legal and commercial workflows. By replacing manual document inspection with cryptographically verifiable credentials, operators can reduce cycle time, error rates, onboarding cost, and audit friction while retaining clear legal accountability.

Key operational and economic benefits include:

- **Reduced order latency:** authorizations that previously required manual review of PDFs or email attachments can be checked automatically through structured credentials and Sentinel policy validation.
- **Lower dispute and audit cost:** signed credential lifecycle events, DIDComm receipts, and Sentinel decisions provide a machine-readable evidence trail.
- **Reusable federation onboarding:** once trust anchors and credential schemas are established, new agent workflows can reuse the same identity and revocation infrastructure.
- **Reduced fraud and misuse:** scoped credentials, short expirations, revocation, and key-storage-tier enforcement limit the blast radius of compromised or misconfigured agents.
- **Progressive adoption:** implementers may start with single-operator authentication and evolve toward full cross-operator DID/VC federation without replacing the entire architecture.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

1.6 Deployment Profiles and Incremental Adoption Guidance

The normative words in this specification apply to implementations claiming the relevant conformance level or deployment profile. They do not imply that every pilot, internal tool, or single-operator deployment must immediately implement the complete DID/VC stack. Implementers SHOULD select the smallest profile that matches the trust boundary and risk of the workflow, then migrate upward as federation requirements increase

Deployment Trust Profile examples	Description	Typical Use
Trust Profile Type 1	Minimal agent authentication suitable for single-operator deployments with limited legal or commercial consequence.	Internal pilots, MCP-enabled tools, controlled lab environments.
Trust Profile Type 2	Cross-operator credential verification with partial DID/VC adoption and explicit Sentinel validation.	Bilateral partner pilots, limited production workflows between known operators.
Trust Profile Type 3	Complete DID/VC lifecycle and policy-driven federation for multi-operator production workflows.	Machine-scale legal or commercial operations across multiple operators.

2. Normative References

The following documents are referenced in this specification. For dated references, only the edition cited applies. For undated references, the latest edition applies.

Identifier	Title	Publisher	Status
W3C DID v1.1	Decentralized Identifiers (DIDs) v1.1	World Wide Web Consortium	Candidate Recommendation Snapshot, March 2026
W3C VC-DATA-MODEL-2.0	Verifiable Credentials Data Model v2.0	World Wide Web Consortium	W3C Recommendation, May 2025
DIDComm v2	DIDComm Messaging v2.0	Decentralized Identity Foundation	Specification, 2023
BitstringStatusList	Bitstring Status List v1.0	W3C Credentials Community Group	W3C Working Group Draft, 2024
RFC 8032	Edwards-Curve Digital Signature Algorithm (EdDSA)	IETF	Informational, January 2017
NIST AI 100-1	Artificial Intelligence Risk Management Framework (AI RMF 1.0)	NIST	Publication, January 2023
EU Regulation 2024/1689	Artificial Intelligence Act	European Parliament and Council	Regulation, June 2024

2.1 Informative References

Identifier	Title	Publisher	Status
W3C AI Agent Protocol CG	AI Agent Protocol Community Group	World Wide Web Consortium	Active Community Group, established June 2025
NIST NCCoE	Accelerating the Adoption of Software and AI Agent Identity and Authorization	NIST NCCoE	Concept Paper, early 2026
Agent Network Protocol	Agent Network Protocol White Paper	ANP Working Group	White Paper, January 2026
EC GPAI CoP	General-Purpose AI Code of Practice	European Commission	Published, July 2025
EC DNA Proposal	Proposal for a Regulation for the Digital Networks Act (DNA)	European Commission	Legislative proposal, adopted by the European Commission on 21 January 2026

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Note: *The W3C AI Agent Protocol Community Group is developing identity and authentication specifications for AI agents. As of March 2026, these specifications are in early development. This specification is designed to align with the Community Group's outputs as they mature.*

Note: *The NIST NCCoE concept paper specifically addresses AI Agent identity and authorization, exploring how existing IAM frameworks can be adapted for AI agents. This specification incorporates the core principles from that paper and is designed to be compatible with forthcoming NIST guidance.*

Note: *The European Commission's General-Purpose AI (GPAI) Code of Practice, published July 2025 under Article 56 of the EU AI Act, provides implementation guidance for transparency obligations and systemic risk management relevant to agents built on GPAI models. Section 8.2 maps this specification's mechanisms to GPAI-specific obligations.*

3. Definitions and Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

3.1 Core Definitions

Agent: A software system that acts on behalf of a human or organizational principal to perform tasks within a federated network. An Agent may operate autonomously or under human supervision. An Agent is a distinct identity holder, separate from the human or organization it represents.

Agent Identity: The complete set of attributes that uniquely identify an Agent within a federation, comprising a Decentralized Identifier (DID), one or more Verifiable Credentials (VCs), and associated cryptographic key material managed in secure environments conforming to the key storage requirements defined in Section 4.7. The DID component of an Agent Identity is globally unique and verifiable. The credentials component is valid within the scope of the issuing federation. Cross-federation recognition requires mutual trust establishment between Governing Authorities (see Section 4.5).

Principal: The human individual or legal entity on whose behalf an Agent operates and who bears ultimate accountability for the Agent's actions. Every Agent MUST be bound to exactly one Principal. In cases where multiple organizations share accountability, the Principal SHOULD be the legal entity that represents the joint arrangement.

Unincorporated Joint Arrangements: Where a joint venture, consortium, outsourcing arrangement, or other joint arrangement has no separate legal personality, the participating parties MUST contractually designate one existing legal entity as the accountable Principal for the Agent. If no accountable Principal is designated, the Agent MUST NOT be issued an Identity Credential. Internal allocation of liability, indemnity, or approval responsibility among the participating parties remains outside the scope of this specification.

Governing Authority: The federation member organization that issues, manages, and may revoke an Agent's identity and credentials. The Governing Authority serves as the trust anchor for its fleet of agents. Qualification criteria for Governing Authorities are defined in Section 4.6.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Controller: The entity designated in a DID Document's `controller` field that holds ultimate authority over the DID and its associated key material. The Controller can update the DID Document, rotate keys, and deactivate the DID -- even without the DID subject's cooperation. In this specification, the Controller is always the Governing Authority. This follows the W3C DID v1.1 definition (Section 5.1.2) but constrains it to organizational governance.

Decentralized Identifier (DID): A globally unique identifier conforming to the W3C DID specification. Unlike the self-sovereign model described in the W3C standard, this specification designates the Governing Authority as the DID controller, enabling centralized governance over a decentralized identifier format. The DID itself is globally resolvable; the credentials bound to it are scoped to the issuing federation.

Identity Credential: A Verifiable Credential issued by the Governing Authority that establishes an Agent's membership in the federation and defines its type, capabilities, and Principal binding. The `NetworkAuthorityCredential` is the canonical Identity Credential type defined by this specification. The generic term "Identity Credential" refers to this category; the specific type name `NetworkAuthorityCredential` is used in credential schemas and implementation code. Identity Credentials MUST NOT be issued by Agents to other Agents.

Operational Credential: A Verifiable Credential issued by one authorized Grantor Agent to another Agent within the scope of a domain-specific workflow. An Operational Credential's trust is derived from the issuing Agent's verified Identity Credential and the Governing Authority's role as trust anchor. A representative example is a digital Letter of Authorization (eLOA) issued by a Grantor Agent to a Requestor Agent (see Section 6.2.2 for a detailed business example).

Verifiable Credential (VC): A tamper-evident, machine-verifiable digital credential conforming to the W3C Verifiable Credentials Data Model. VCs encode an Agent's authorizations, capabilities, and operational scope. This specification distinguishes between Identity Credentials and Operational Credentials.

Verifiable Presentation (VP): A tamper-evident, machine-verifiable package of one or more VCs presented by an Agent to a Verifying Party, optionally using selective disclosure to reveal only the information required.

Sentinel: A privacy-preserving verification endpoint that validates Agent credentials before granting access to a protected resource. The Sentinel verifies Verifiable Presentations and enforces credential status checks. Each organization operating protected resources runs its own Sentinel Agent(s).

Maximum Revocation Interval (MRI): The maximum permissible time between a credential revocation event and the guaranteed rejection of that credential by all Sentinels in the federation. This specification sets the MRI at five (5) minutes.

Irrevocable Liability: A binding legal or financial obligation that cannot be canceled, withdrawn, or modified without the consent of all parties involved. It signifies a final, permanent commitment, meaning the obligor cannot back out without facing breach of contract or financial penalties

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

3.2 Role Definitions

Grantor Agent: The Agent that operates on behalf of the asset owner or authorizing party. The Grantor Agent creates and transfers Operational Credentials that delegate specific rights or authorizations to other Agents. Only Grantor Agents holding a verified `GrantorRoleCredential` may issue Operational Credentials.

Requestor Agent: The Agent that operates on behalf of the service buyer or receiving party. The Requestor Agent receives Operational Credentials from Grantor Agents and presents them to Sentinel Agents for verification when requesting access to resources or services.

Sentinel Agent: The Agent that operates on behalf of the resource operator or service provider. The Sentinel Agent verifies Operational Credentials presented by Requestor Agents and enforces access policies, including real-time revocation checks and domain-specific data validation. Each Governing Authority operates its own Sentinel Agent(s); a Sentinel cannot verify on behalf of a different Governing Authority unless cross-federation trust has been established (see Section 4.5).

Orchestrator Agent: The Agent that manages the lifecycle of other Agents on behalf of its Governing Authority. The Orchestrator Agent handles bootstrapping, monitoring, key rotation, and decommissioning operations. Each Governing Authority operates its own Orchestrator Agent; an Orchestrator cannot manage agents belonging to a different Governing Authority.

Supervisor Agent: The Agent that operates on behalf of the Principal to provide asynchronous oversight of operational workflows. The Supervisor manages Human-in-the-Loop (HITL) escalation queues, monitors behavioral anomalies, and instructs the Orchestrator when emergency suspension of a rogue agent is required.

3.3 Credential Type Taxonomy

This specification defines a two-tier credential model:

Tier 1 -- Identity Credentials (issued by Governing Authority only):

Credential Type	Purpose	Recipient
`NetworkAuthorityCredential`	Proves the agent is a recognized member of the federation	Every Agent (mandatory)
`GrantorRoleCredential`	Authorizes the agent to issue Operational Credentials	Grantor Agents
`RequestorRoleCredential`	Authorizes the agent to receive and present credentials	Requestor Agents
`SentinelRoleCredential`	Authorizes the agent to verify credentials	Sentinel Agents
`OrchestratorRoleCredential`	Authorizes the agent to manage other agents' lifecycles	Orchestrator Agents

Tier 2 -- Operational Credentials Example (issued by authorized Grantor Agents):

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Credential Type examples	Purpose	Issued By
`ServiceOrderCredential`	Authorizes a service provisioning action	Grantor Agent
`AccessAuthorizationCredential`	Grants physical or logical access to a facility	Grantor Agent
`eLOA`(electronicLetter of Authorization)	Authorizes a cross-connect installation	Grantor Agent

The Governing Authority determines which Tier 1 credentials an Agent receives during the bootstrapping process (Section 5.3), based on the Agent's intended role. Tier 2 credentials are created during domain-specific workflows by Agents that hold appropriate Tier 1 role credentials. If the issuing Agent's `GrantorRoleCredential` is revoked, it can no longer issue valid Operational Credentials.

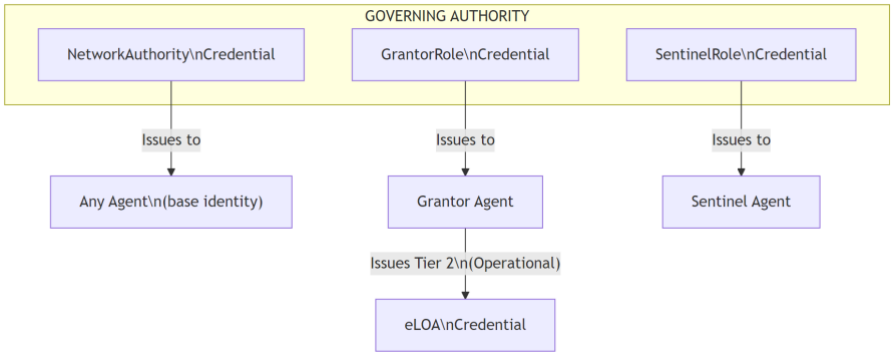


Figure 1: Credential Type Taxonomy — Two-tier credential issuance model

3.4 Trust-Domain Concepts

Real-world deployments rarely operate as one uniform federation. Implementations MUST make trust boundaries explicit so that Sentinels, Governing Authorities, and relying parties can determine where trust is native, where it must be re-established, and where additional legal or human approval is required.

Trust Domain Typology:

- Formal Trust Domain: A trust domain backed by explicit legal agreements, cryptographic verifiability, and strict deterministic business rules. Agents operating in this domain process high-liability transactions (e.g., executing contracts, placing cross-connect orders).
- Informal Trust Domain: A trust domain characterized by conversational, un-verified interactions without legal enforceability (e.g., an agent negotiating loosely over chat or gathering general information).

3.4.1 Prevention of Context Drift

To ensure legal enforceability and prevent unintended liability, Agents MUST be designed such that they rigorously adhere to the Formal Trust Domain and cannot 'drift' into the Informal Trust Domain during execution. To prevent context drift, Agents executing operations within a Formal Trust Domain MUST NOT rely on natural human language reasoning for authorization decisions. Instead, they MUST operate using minimal instruction sets based entirely on strict deterministic business rules. All operational workflows MUST map strictly to predefined data schemas, and the Sentinel/Supervisor MUST reject any operation containing unmapped or unstructured semantic instructions.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Domain Name / Concept	Definition	Examples
Trusted Identity	A digital or physical identity that a relying party is willing to accept because it is issued, attested, or validated by a recognized Trust Anchor and is bound to a subject, scope, validity period, and accountability record. A Trusted Identity does not automatically create authorization; it only establishes that the subject is recognized under the applicable trust framework.	• A U.S. passport identifying a U.S. citizen; • An Agent DID with a valid NetworkAuthorityCredential issued by a recognized Governing Authority; • A Provider A's Requestor Agent identity verified by Provider B's Sentinel before accepting a quotation request.
Trust Domain	A bounded set of operators, systems, agents, policies, credential schemas, verification rules, and legal commitments that share a common Trust Anchor or mutually recognized set of Trust Anchors. Trust is native inside the domain and does not require re-establishment for every transaction, provided the credential remains valid and within scope.	• The U.S. passport issuance domain, where the U.S. Department of State acts as the issuing authority; • A national driving-license domain operated by state motor vehicle authorities; • An Mplify operator federation where participating Governing Authorities are recognized by a common trust registry.
Credential Acceptance Domain	The set of relying parties, jurisdictions, platforms, operators, or systems that agree to accept a given credential for a defined purpose. This may be larger, smaller, or different from the Trust Domain that issued the credential.	• A U.S. passport may be accepted internationally as proof of identity and nationality, but each destination country still decides entry conditions. • A U.S. driving license may be accepted for domestic identity checks but not as a general international travel document. • A NetworkAuthorityCredential may be accepted by multiple operators only where their Sentinels recognize the issuing Governing Authority.
Trust Anchor	The root authority, registry, issuer, or legally recognized entity whose assertion allows relying parties to validate identities, credentials, keys, or status information. A Trust Anchor may be internal to one domain or shared across multiple domains.	• U.S. Department of State for U.S. passports; • A state DMV for a driving license; • A federation Governing Authority; • A regulated certification authority used to validate organizational credentials.
Trust Boundary	The perimeter where a credential that is valid in one Trust Domain must be re-evaluated before it is accepted in another domain, system, jurisdiction, or business process. Crossing a Trust Boundary may require policy checks, credential-status checks, additional authorization, or human approval.	• Airline check-in or boarding desk validating ESTA status before travel to the United States; • U.S. border control or pre-clearance in a foreign airport;
Extended Trust Boundary	A Trust Boundary that is operated collectively across multiple jurisdictions or participants under shared rules, so that the effective boundary is moved outward or consolidated.	• The Schengen external border, where common rules apply to external border checks while internal border controls are generally removed among participating states; • U.S. pre-clearance facilities located in foreign airports, where U.S. entry checks are performed before physical arrival in the United States.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Domain Name / Concept	Definition	Examples
Delegated Trust	A controlled transfer of authority from one Trust Domain, authority, or credentialed actor to another. Delegated Trust MUST be scoped, time-bound, revocable, and auditable. Delegated Trust does not eliminate the original accountability chain unless the legal framework explicitly transfers liability.	• a company delegating limited signing authority to an AI Agent for NDAs below a defined value; • a travel authorization such as ESTA allowing a carrier to board a traveler under defined conditions, without replacing final border admissibility checks.
Cross-Domain Credential Recognition	The act by which a relying party in one Trust Domain accepts a credential issued in another Trust Domain after verifying issuer trust, credential integrity, credential status, scope, policy compatibility, and legal enforceability.	• A passport issued by one country being recognized by another country as identity evidence, while visas or entry authorizations determine permission to enter; • Provider B recognizing Provider A's NetworkAuthorityCredential but requiring an additional Operational Credential for the specific RFQ or service-order action.

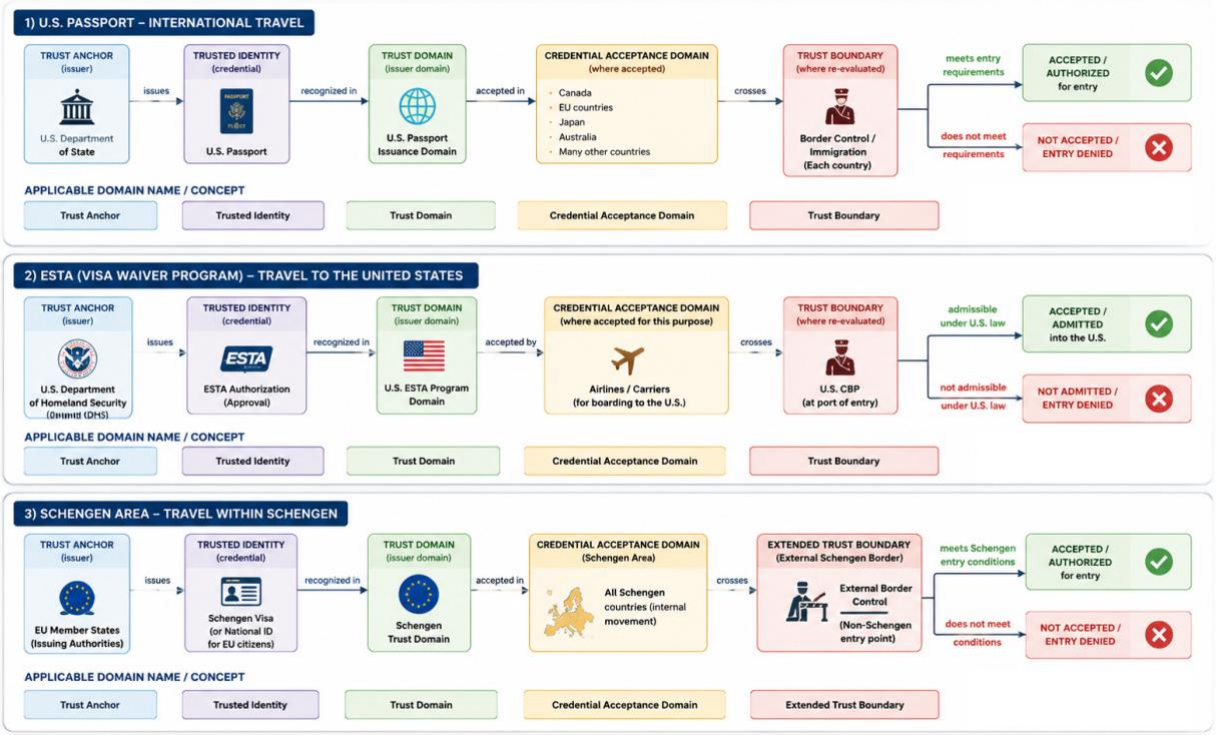


Figure 2: Examples

4. Agent Identity Model

4.1 Design Principles

The Agent Identity Model is founded on three principles that balance the need for decentralized operation with the requirement for centralized governance and legal accountability:

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Principle 1 -- Centralized Trust, Decentralized Identity
All Identity Credentials are issued exclusively by a recognized Governing Authority. Identity infrastructure -- the DID and its associated key material -- is generated and maintained locally by the Agent in secure environments conforming to the key storage requirements defined in Section 4.7. The Governing Authority is designated as the controller of every Agent DID, granting it the ability to intervene in key recovery and identity decommissioning scenarios without requiring the Agent's cooperation.

This design prevents "rogue admin" scenarios where a sub-authority issues identity credentials outside the governance framework. Chaining of Identity Credentials is expressly prohibited. However, Agents MAY issue Operational Credentials to other Agents within the context of domain-specific workflows, subject to the delegation depth limits defined in Section 6.5. The trust of such Operational Credentials is derived from the issuing Agent's verified Identity Credential, not from independent authority.

Principle 2 -- Identity-Authorization Separation
An Agent's identity (its DID) is distinct from its authorization (its VCs). An Agent may possess a valid DID but hold no authorization credentials, in which case it can authenticate but cannot access any protected resources. Conversely, a revoked credential does not invalidate the Agent's DID -- the DID remains in place for audit trail integrity while the credential is revoked.
This separation enables granular lifecycle management. An Agent can have specific authorizations revoked without losing its identity and can have its identity forcibly retired without requiring every credential to be individually processed.

Principle 3 -- Accountability Chain
Every Agent identity MUST be traceable through an unbroken chain of accountability from the Agent's Legal Identity Credential and Operational Credential to its Governing Authority and, through the Governing Authority's records, to its Principal. This chain provides the technical evidence infrastructure that supports legal accountability: when an Agent takes an action with legal consequences, the responsible person or legal entity can be identified. Legal enforceability of this chain requires complementary contractual and regulatory frameworks (see Section 8.6).

> ***Note: Multi-Party Principal Arrangements****: Where multiple legal entities share accountability for an Agent's actions (e.g., in a joint venture, consortium, or temporary delegation), the Principal SHOULD be a single legal entity that represents the joint arrangement (e.g., the joint venture LLC). The internal multi-party approval process (e.g., requiring two executives to co-approve) happens at the organizational layer before the Governing Authority instructs the Agent. The Agent's Principal is the organization, not the individual approvers. The internal allocation of liability among the joint arrangement's members is governed by the organizational agreements of that entity and is outside the scope of this identity specification. This preserves the clean 1:1 Principal-to-Agent binding while accommodating multi-party commercial reality.* *Where the arrangement has no separate legal entity, the participating parties MUST designate one existing legal entity contractually as the accountable Principal; otherwise, the Agent is unsupported under this specification and MUST NOT receive an Identity Credential.*

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

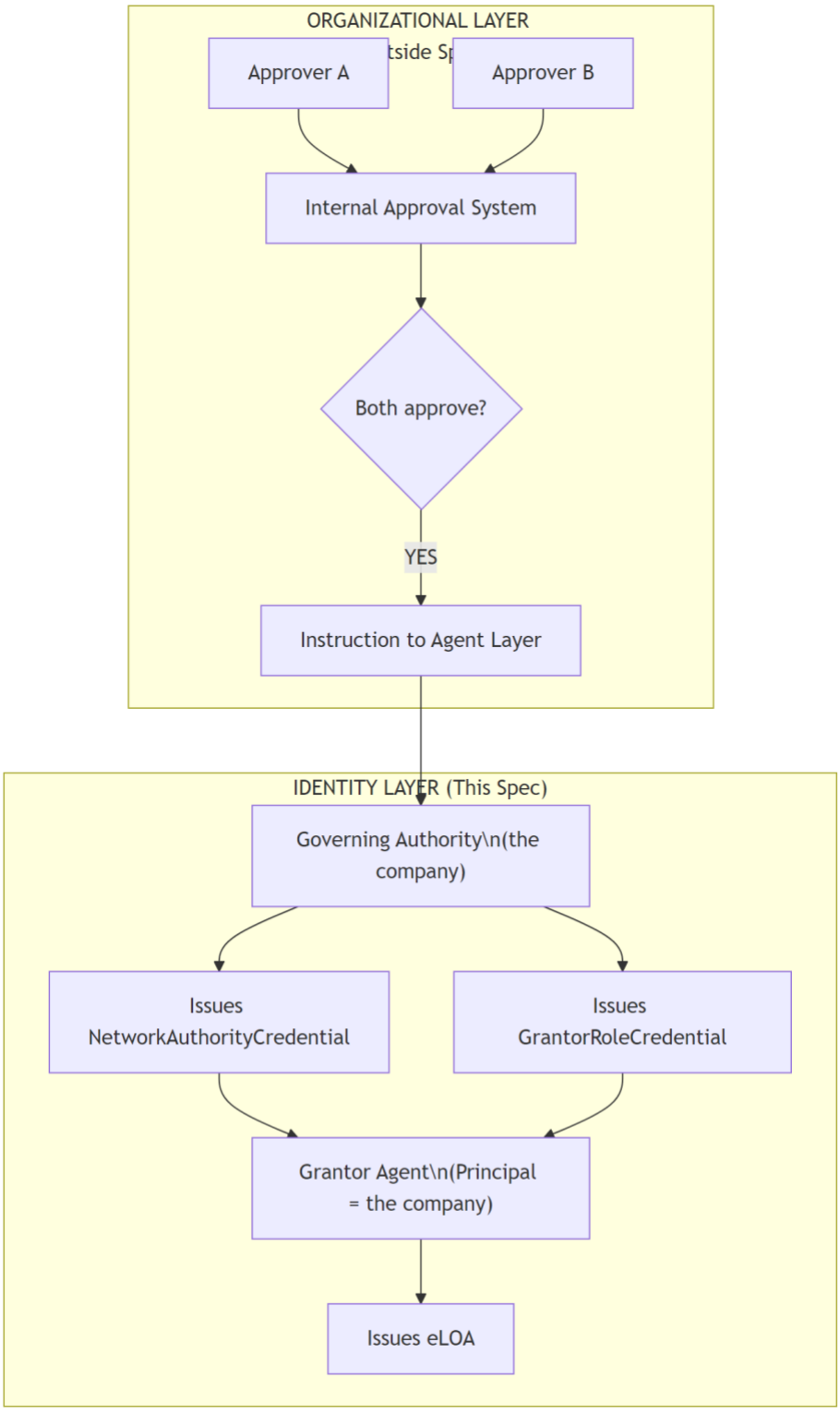


Figure 3: Accountability Chain — Multi-party principal arrangement with organizational and identity layers

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

4.2 Identity Structure

An Agent Identity comprises the following components:

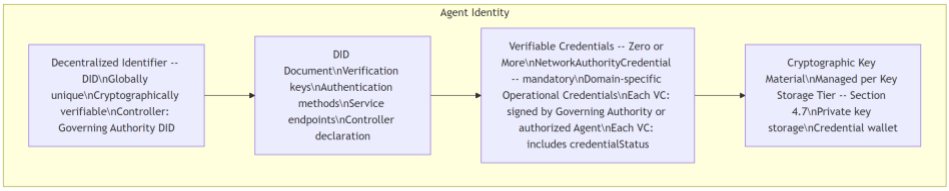


Figure 4: Agent Identity Structure — Components of a complete Agent Identity

4.3 DID Document Requirements

Every Agent DID Document MUST conform to the following requirements:

1. The DID Document MUST specify the Governing Authority's DID as `controller`.
2. The DID Document MUST include at least one `verificationMethod` using a cryptographic algorithm suitable for digital signatures (e.g., Ed25519).
3. The DID Document MUST enable the `authentication` relationship for at least one verification method.
4. The DID Document MUST be published to a DID registry operated by or designated by the Governing Authority. The registry MUST expose a publicly accessible resolution endpoint conforming to the W3C DID Resolution specification, enabling cross-organizational DID resolution.

Illustrative DID Document:

```
{
  "@context":                                                ["https://www.w3.org/ns/did/v1"],
  "id":                                                              "did:example:agent-087",
  "controller":                                              "did:example:member-enterprise",
  "verificationMethod":                                                                    [
    {
      "id":                                                   "did:example:agent-087#key-1",
      "type":                                                  "Ed25519VerificationKey2020",
      "controller":                                                  "did:example:agent-087",
      "publicKeyMultibase":                "z6Mkf5rGMoatrSj1f4CyvuHBeXJELe9RPdzo2PKGNCKVtZxP"
    }
  ],
  "authentication":                                                                 ["#key-1"],
  "assertionMethod":                                                                 ["#key-1"]
}
```

> ***Note:*** *This DID Document contains two distinct `controller` fields with different meanings. The top-level `controller` field (`did:example:member-enterprise`) designates the **Governing Authority** as the entity that governs this DID -- it holds the right to update, rotate keys, or deactivate the DID. The `verificationMethod.controller` field (`did:example:agent-087`) designates the **Agent itself** as the entity that operates the key for day-to-day signing. The Governing Authority does not use the Agent's key directly; it can replace it via emergency rotation using its controller authority, but routine signing is performed by the Agent.*
>
> ***Note:*** *The DID method, cryptographic algorithm, and key representation shown above are illustrative examples. Implementations SHOULD select DID methods and cryptographic primitives appropriate for their deployment environment and security requirements.*

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

4.4 Agent Type Classification

This specification defines the following Agent types within a legal authorization workflow. Each type has a distinct credential profile and operational scope:

Agent Type	Role	Required Credentials	Operational Scope
Grantor Agent	Issues authorizations on behalf of the asset owner	NetworkAuthorityCredential + GrantorRoleCredential	Create, sign, and transfer Operational Credentials
Requestor Agent	Receives and presents authorizations	NetworkAuthorityCredential + RequestorRoleCredential	Receive, store, and present Operational Credentials
Sentinel Agent	Verifies credentials and enforces access policies	NetworkAuthorityCredential + SentinelRoleCredential	Verify presentations, check revocation, validate operational data
Orchestrator Agent	Manages the lifecycle of other Agents	NetworkAuthorityCredential + OrchestratorRoleCredential	Bootstrap, monitor, rotate, and decommission other Agents

Each Governing Authority operates its own Orchestrator Agent(s) and Sentinel Agent(s). An Orchestrator cannot manage agents belonging to a different Governing Authority. Cross-organizational agent verification requires cross-federation trust establishment (Section 4.5).

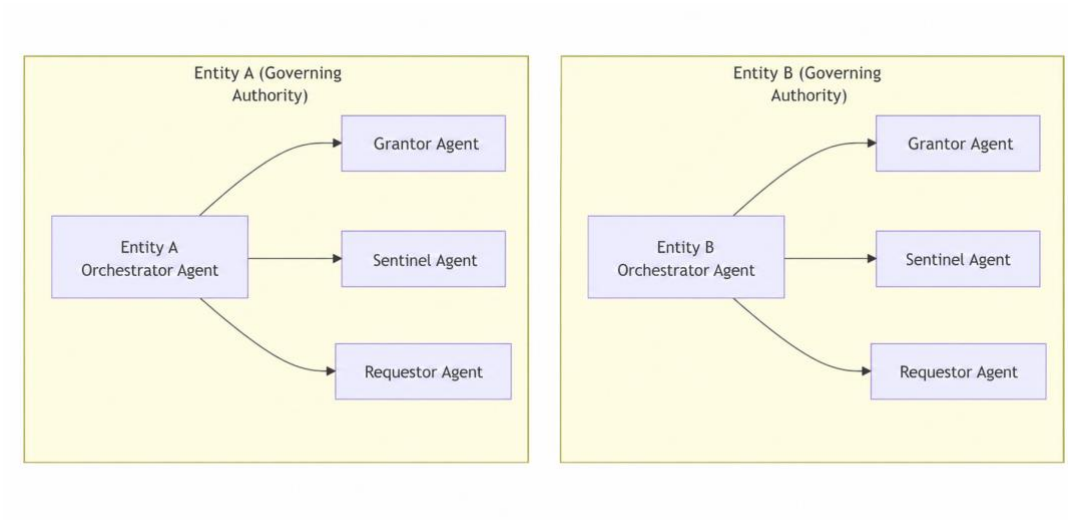


Figure 5: Agent Type Classification — Governing Authority organizational model

4.5 Cross-Federation Trust Model

Real-world deployments involve cross-organizational interactions where a Requestor Agent from one Governing Authority presents credentials to a Sentinel belonging to a different Governing Authority. This section defines the mechanism for establishing mutual trust between federations.

Trust Anchor Exchange: Governing Authorities establish mutual trust by exchanging their DIDs and public keys through a bilateral agreement or by publishing them to a shared trust registry. This is analogous to how Certificate Authorities operate in TLS -- participants pre-install trusted root credentials.

Sentinel Trust List: Each Sentinel maintains a list of trusted Governing Authority DIDs. When a Sentinel receives a credential from an unfamiliar issuer, it checks whether that issuer's Governing

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Authority appears on its trust list. Credentials from unrecognized Governing Authorities MUST be rejected.

Cross-Federation Verification Protocol: When a Requestor Agent from Operator A presents a Verifiable Presentation to Operator B's Sentinel, the following steps occur:

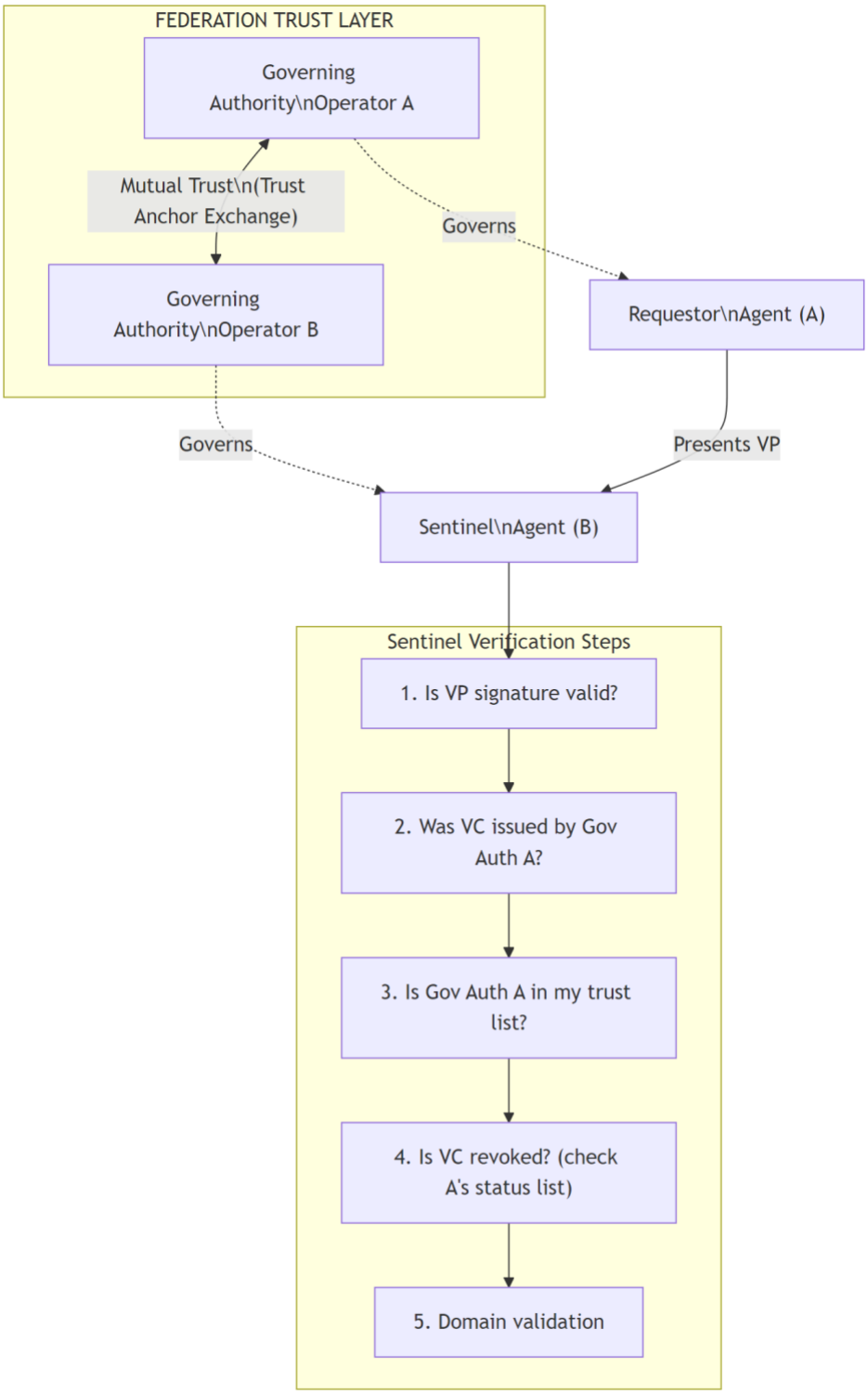


Figure 6: Cross-Federation Verification Protocol — Sentinel verification steps for cross-organizational credentials

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Status List Cross-Access: Each Governing Authority's BitstringStatusList endpoint MUST be publicly accessible so that Sentinels from other organizations can perform revocation checks against the issuing authority's status list.

DID Registry Interoperability: Each Governing Authority may operate its own DID registry. Cross-federation resolution is achieved through one of two models:

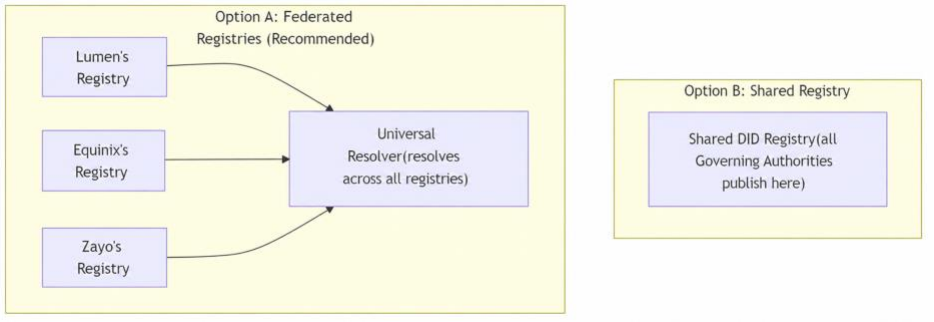


Figure 7: DID Registry Interoperability — Shared vs. Federated registry models

4.6 Governing Authority Qualification

A Governing Authority MUST meet the following minimum criteria:

Criteria	Rationale
Must be an identifiable legal entity	Accountability chain requires a legally addressable party
Must operate or designate a DID registry with a publicly accessible resolution endpoint	Agents need a discoverable place to publish DID Documents; cross-federation resolution requires public access
Must maintain a BitstringStatusList endpoint	Revocation infrastructure must be operational
Must maintain heartbeat publishing at intervals no greater than the MRI	MRI compliance requires continuous liveness signals
Must designate a human administrator	Human oversight per EU AI Act Art. 14

Additional qualification requirements may be established by the federation's governance charter.

4.7 Key Storage Requirements

The authority an Agent can exercise is bounded by the security of its key storage. This section defines a three-tier model that links key storage security to credential authority.

Tier	Key Storage	Agent Types	Authority Level
Tier 3 -- HSM	Hardware Security Module (FIPS 140-3 Level 3+)	Governing Authority Orchestrator, high-value Grantor Agents	Full -- may issue credentials with no restrictions
Tier 2 -- TPM/TEE	Trusted Platform Module or Trusted Execution Environment	Production Sentinel Agents, standard Grantor/Requestor Agents	Standard -- may participate in all operational workflows
Tier 1 -- Software	Encrypted keystore (e.g., OS keychain, KMS-backed)	Dev/test agents, low-value Requestor Agents	Limited -- may only hold credentials with short expiration (24 hours or less) and reduced operational scope

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

The `NetworkAuthorityCredential` schema includes a `keyStorageTier` field (see Section 6.2.1) that Sentinels MAY use to enforce key storage requirements during verification. A Governing Authority SHOULD issue weaker credentials (shorter TTL, narrower capabilities) to Agents with lower key storage tiers.

4.8 Open Agent Interconnection (OAI) Model

Implementations SHOULD keep each layer (e.g. identity, messaging, tool invocation, and business operations) separated so that a change in one layer does not require redesign of the others.

Layer	Description	Implementation examples	Use cases	Constraints
Identity layer	Answers who the Agent is, what it is authorized to do, and whether the credential is current	DID; VC	Agent DID Documents, NetworkAuthorityCredential, Operational Credentials, Verifiable Presentations, status lists.	Answers who the Agent is, what it is authorized to do, and whether the credential is current.
Agent to Agent messaging layer	Secure agent-to-agent messaging. (e.g. Present-proof, issue-credential, challenge/response, secure agent transport.)	DIDComm; A2A	Present-proof, issue-credential, challenge/response, secure agent transport.	Carries proofs and messages. It does not define business semantics by itself.
The tool access layer	Tool invocation and runtime access to capabilities.	MCP	Model-initiated calls to tools, services, or enterprise functions.	Protected MCP calls SHOULD be mediated by a Sentinel before tool execution.
Workflow interaction layer	Business operations	LSO APIs; Network API	Ordering, provisioning, assurance, cross-connect workflows.	Executed only after the identity, authorization, and policy layers have approved the operation.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

4.9 Sentinel Functional Responsibilities

A Sentinel is an Agent that operate a logical enforcement function.

Function	Responsibility	Implementation Notes
Enforcement Boundary	Validate credentials and policies before protected resources, APIs, or MCP tools are invoked.	The Sentinel marks the transition from untrusted or partially trusted agent interaction to authorized business execution.
Trust Authority Scope	Verify issuer trust, credential status, schema version, expiration, key-storage tier, and domain policy.	The Sentinel does not issue Identity Credentials and does not expand authority beyond the credentials and policies it validates.
Rules Engine Integration	Evaluate domain-specific constraints such as facility, cabinet, port, bandwidth, order type, or commercial threshold.	The Sentinel MAY delegate rule evaluation to an external rules engine, but it remains accountable for fail-closed decisioning and audit output.
MCP and Runtime Integration	Authorize protected tool invocation before an MCP runtime or agent framework executes the requested function.	The Sentinel SHOULD bind the authorization decision to the tool call, request parameters, nonce, and session context.
Audit and Evidence	Emit immutable audit events for challenge issuance, presentation receipt, policy version, decision, revocation status, override, and downstream API correlation.	Audit events SHOULD be exportable to compliance, SIEM, ticketing, and dispute-resolution systems.

5. Agent Identity Lifecycle

5.1 Lifecycle Overview

The Agent Identity Lifecycle defines the complete sequence of states and transitions that an Agent identity undergoes from creation to retirement. The Governing Authority maintains control at every stage.

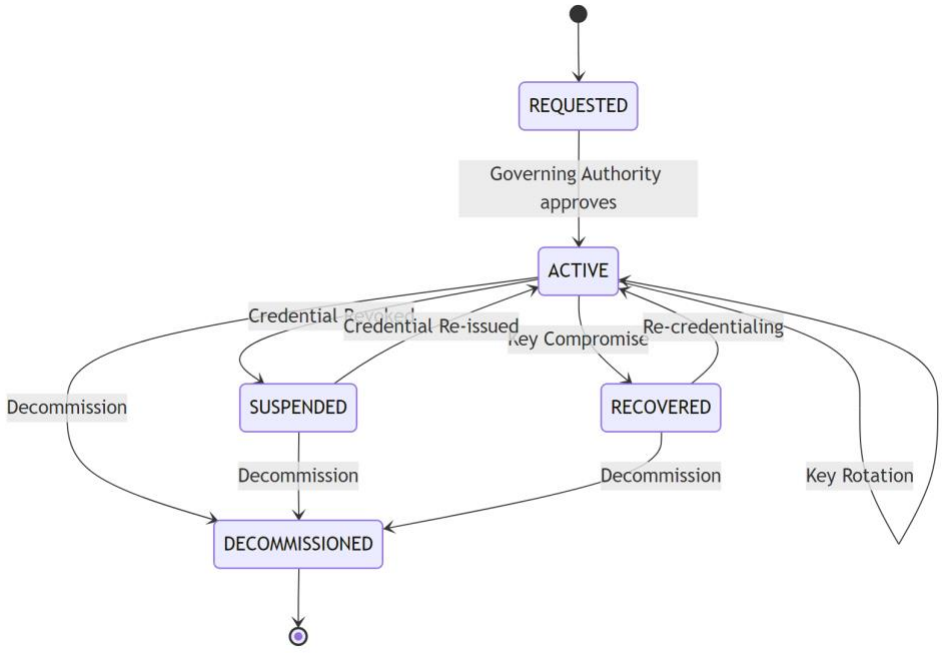


Figure 8: Agent Identity Lifecycle — State transitions from creation to retirement

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

5.2 State Definitions

REQUESTED: The Agent has generated key material in a secure environment conforming to its designated key storage tier (Section 4.7) and published a DID Document to the registry, but has not yet received any credentials from the Governing Authority. The Agent cannot participate in any operational workflow.

ACTIVE: The Agent holds a valid NetworkAuthorityCredential and any additional role-specific credentials. It is authorized to participate in operational workflows according to its credential scope.

SUSPENDED: One or more of the Agent's credentials have been revoked, but its DID has not been deactivated. Suspension is per-credential, not per-identity: the Agent remains ACTIVE for operations covered by its remaining valid credentials and is SUSPENDED only with respect to operations requiring the revoked credential(s). The Governing Authority may re-issue credentials to return the Agent to fully ACTIVE state.

RECOVERED: The Governing Authority has forcibly rotated the Agent's keys due to a suspected compromise. The Agent's DID Document has been updated with new key material. The Agent must re-request credentials before returning to ACTIVE state.

DECOMMISSIONED: The Agent's DID has been deactivated on the registry, and all associated credentials have been revoked. This state is terminal. The DID remains in the registry for audit trail purposes but cannot be used for any new operations.

5.3 Bootstrapping Protocol

The bootstrapping protocol creates a new Agent identity under the governance of the Governing Authority. The protocol follows these steps:

1. **Key Generation**: The Agent generates a cryptographic key pair in a secure environment conforming to its designated key storage tier (Section 4.7).
2. **DID Publication**: The Agent constructs a DID Document designating the Governing Authority as `controller` and publishes it to the Governing Authority's DID registry.
3. **Authorization Request**: The Agent sends a DIDComm `present-proof` message to the Governing Authority, proving control of the DID's private key.
4. **Verification**: The Governing Authority resolves the Agent's DID from the registry, verifies the `controller` field points to itself, and validates the proof of key control. If any verification step fails, the bootstrapping request is rejected, the DID remains in REQUESTED state, and the Governing Authority MUST log the rejection reason.
5. **Credential Issuance**: The Governing Authority issues a NetworkAuthorityCredential (and any applicable role credentials) to the Agent via DIDComm `issue-credential`. The `keyStorageTier` field is set based on the Agent's attested key storage environment.
6. **Storage**: The Agent stores the received credentials in its secure credential wallet.

The Agent is now in ACTIVE state.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

5.4 Key Rotation

Key rotation replaces an Agent's cryptographic key material while preserving its DID and credentials.

Planned Rotation: The Agent generates a new key pair, constructs a DID Update transaction signed by the current key, and publishes it to the registry. Existing credentials remain valid; new proof presentations use the new key.

Emergency Rotation (Recovery): Triggered by the Governing Authority upon documented evidence of a key compromise. Except where immediate action is necessary to prevent imminent harm, the Governing Authority MUST document the reason for the emergency rotation. The Governing Authority signs a DID Update on behalf of the Agent using its controller authority, forcibly replacing the key material. The Agent enters RECOVERED state.

5.5 Decommissioning

Decommissioning permanently retires an Agent identity. It is a two-step process to ensure no residual authority remains:

1. **Authority Revocation**: The Governing Authority revokes all Verifiable Credentials issued to the Agent by updating the relevant BitstringStatusList entries. This immediately prevents the Agent from proving authorization at any Sentinel.
2. **Identity Deactivation**: The Governing Authority signs a DID Update marking the Agent's DID as deactivated on the registry. This prevents the DID from being used for any future authentication.

The order of operations is deliberate: authority is removed before identity, ensuring that there is no window during which the Agent has an active identity but could exploit residual credentials.

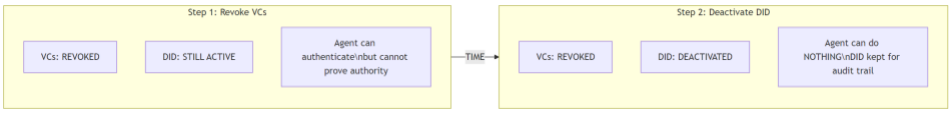


Figure 9: Decommissioning Protocol — Two-step authority-then-identity retirement

> ***Note: Corporate Change Scenarios****: In the event of a corporate acquisition, merger, or organizational restructuring, the Governing Authority may decommission the acquired organization's Agent fleet and re-bootstrap new Agents under the new Principal, or re-issue NetworkAuthorityCredentials with an updated* `principalRef` *field. In this case Governing Authority MUST document the reason and inform in advance all the effected entities. This approach preserves the accountability chain while accommodating changes in organizational ownership.*

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

6. Credential Architecture

6.1 Credential Design Principles

All Agent credentials in this specification conform to the W3C Verifiable Credentials Data Model and adhere to the following design principles:

1. **Single Issuer for Identity Credentials**: All Identity Credentials MUST be issued exclusively by the Governing Authority. Operational Credentials MAY be issued by authorized Agents within domain workflows, subject to the delegation depth limits in Section 6.5. An Agent's authority to issue Operational Credentials is derived from its verified Identity Credential.
2. **Revocable by Default**: Every credential MUST include a `credentialStatus` field referencing a BitstringStatusList entry, enabling real-time revocation.
3. **Schema Versioned**: Every credential MUST include a `schemaVersion` field. Verifying parties (Sentinels) MUST reject credentials with outdated schema versions.
4. **Selective Disclosure Enabled**: All credentials SHOULD be signed using a signature scheme that supports selective disclosure, enabling privacy-preserving verification.

6.2 Core Credential Types

6.2.1 NetworkAuthorityCredential

The foundational credential that establishes an Agent as a recognized member of the federation. Every Agent MUST hold this credential to participate in any operational workflow.

Field	Description
`issuer`	Governing Authority DID
`credentialSubject.id`	Agent DID
`credentialSubject.agentType`	One of: GRANTOR, REQUESTOR, SENTINEL, ORCHESTRATOR
`credentialSubject.principalRef`	Opaque reference linking to the human/organization Principal
`credentialSubject.capabilities`	Array of authorized capability strings
`credentialSubject.keyStorageTier`	One of: TIER_1_SOFTWARE, TIER_2_TPM, TIER_3_HSM (see Section 4.7)
`schemaVersion`	e.g., "1.1"
`issuanceDate`	ISO 8601 timestamp
`expirationDate`	ISO 8601 timestamp
`credentialStatus`	BitstringStatusList reference

6.2.2 Domain-Specific Operational Credentials

In addition to Identity Credentials, Agents may hold or issue domain-specific Operational Credentials. These are legal or commercial authorization credentials issued by a Grantor Agent to a Requestor Agent, asserting that the Requestor is authorized to perform specific actions or access specific resources. The Operational Credential's trust is derived from the issuing Grantor Agent's verified Identity Credential and the Governing Authority's role as the trust anchor.

Business Context: Operational Credentials enable automated verification of authorizations that historically required manual document review. By encoding authorization data as machine-readable, cryptographically verifiable data elements, verifying systems can automatically validate that the authorization matches their operational requirements -- eliminating manual review delays and reducing error rates.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Representative Use Case -- Electronic Letter of Authorization (eLOA):
In data center cross-connection workflows, a physical interconnection order requires a Letter of Authorization from the asset owner. Today, this involves a PDF or paper form emailed to the data center operator, where an employee manually reads and verifies the details. With Operational Credentials, this becomes:

1. **Company A's Grantor Agent** issues a digital eLOA (an Operational Credential) to **Company B's Requestor Agent**, containing the same authorization data -- but as structured, signed, machine-readable fields.
2. **Company B's Requestor Agent** presents this credential to **Equinix's Sentinel Agent**.
3. **Data Center's Sentinel Agent** verifies the signature, checks revocation status, and validates port/cabinet data against its own inventory -- all automatically, in seconds.

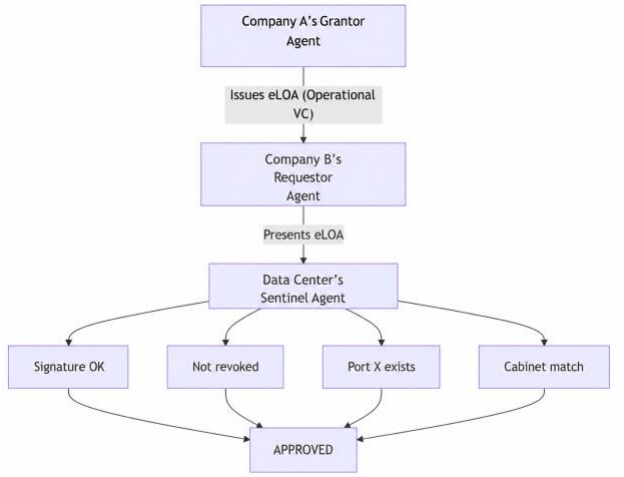


Figure 10: eLOA Operational Credential Workflow — Automated cross-connect authorization

The credential data includes structured fields such as `facilityCode`, `cabinetId`, `portId`, and `mediaType` -- each individually verifiable and each optionally hidden through selective disclosure.

6.3 Credential Verification Protocol

When an Agent presents a credential to a Sentinel for verification, the following checks MUST be performed, in order:

1. **Proof Integrity**: Verify the cryptographic proof on the Verifiable Presentation.
2. **Nonce Binding**: Verify that the presentation is bound to the Sentinel-provided challenge nonce (replay resistance).
3. **Revocation Check 1**: Query the local revocation cache for the credential's BitstringStatusList index.
4. **Session Establishment**: Complete the session key exchange.
5. **Revocation Check 2**: Re-query the local revocation cache (atomic TOCTOU prevention).
6. **Schema Version Check**: Verify the credential's `schemaVersion` is current.
7. **Expiration Check**: Verify the credential has not expired.
8. **Domain Validation**: Perform any domain-specific validation (including multi-credential requirements per Section 6.5).

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

If any check fails, the Sentinel MUST reject the presentation and terminate the session. The system is designed to fail closed: any uncertainty results in rejection.

> ***Note: TOCTOU Prevention Rationale****: The session key exchange in step 4 is deliberately placed between the two revocation checks. The dual revocation check (steps 3 and 5) creates a time window around the key exchange to detect revocations that occur during the cryptographic operations. If all checks were performed first and the session established afterward, a revocation occurring during the key exchange would go undetected. By sandwiching the key exchange between two revocation checks, the protocol guarantees that no session is established using a credential that was revoked at any point during the verification process.*

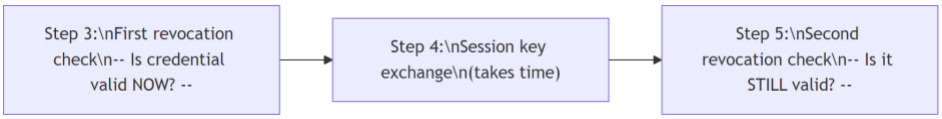


Figure 11: TOCTOU Prevention — Dual revocation check surrounding session key exchange

6.4 Revocation Infrastructure

Credential revocation follows a hybrid push-and-pull model:
Primary Channel -- Push-Based:

1. The Governing Authority updates the master BitstringStatusList and publishes the updated list to a content-addressable data store, generating a new content identifier (CID).
2. The Governing Authority signs the CID with a JSON Web Signature (JWS) and publishes a notification message to a secure publish-subscribe channel.
3. All subscribed Sentinels receive the notification, verify the JWS signature, retrieve the updated list from the content-addressable store, and update their local cache.

Fallback Channel -- Pull-Based:

The Governing Authority MUST expose an HTTPS endpoint serving the current BitstringStatusList. When the push channel is unavailable, Sentinels SHOULD poll this endpoint at `MRI/2` intervals (every 2.5 minutes). A successful pull MUST reset the heartbeat-missed counter.

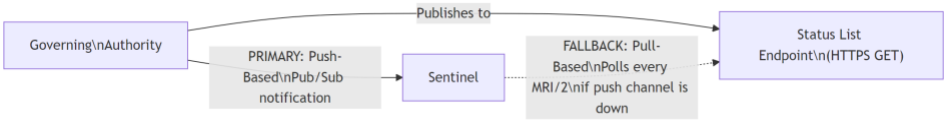


Figure 12: Revocation Infrastructure — Push-based primary with pull-based fallback

Liveness Guarantees:

- **Cold Start**: A Sentinel MUST perform a full resynchronization from the content-addressable store before accepting any connections after startup.
- **Heartbeat**: The Governing Authority MUST publish periodic signed heartbeat messages at an interval no greater than the MRI value (i.e., no more than 5 minutes). Where a Governing Authority outage triggers degraded or fail-closed mode, the Governing Authority MUST publicly announce the degraded status and the related causes. If a Sentinel misses a heartbeat, the fail-closed policy in Section 6.4.2 applies.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

6.4.1 Revocation Authority and Status-List Governance

- **Operational Credential Status:** Status-list entries for any Credentials MUST be allocated and maintained by a credential-status service operated by, or under the control of, the issuing Grantor Agent's Governing Authority.
- **Revocation Initiation and Authority:**
    1. The Governing Authority CAN revoke any Credentials issued under its authority and it remains the ultimate revocation authority.
    2. In addition to the Governing Authority a Grantor Agent MAY also revoke Credentials
    3. A Grantor Agent MUST only revoke Credentials that it issued

6.4.2 Revocation Resilience

The fail-closed behavior below preserves the five-minute MRI as a hard Level 2 conformance requirement while distinguishing new authorization decisions from the continuity of already-authorized sessions:

Scenario	Behavior
Revocation status information is less than one MRI old	The Sentinel MAY continue processing while attempting immediate synchronization. The authorization decision MUST be bound to the authenticated status-list version and timestamp used.
Revocation status information reaches the MRI	The Sentinel MUST reject all new credential presentations that require revocation verification. No grace period beyond the MRI is permitted for Level 2 conformance.
Connectivity or status service remains unavailable beyond the MRI	Fail closed for all new authorization decisions. Existing sessions MAY complete only under a separately defined session-continuity policy that does not create a new authorization decision.
Connectivity restored	Full authenticated resynchronization and status-list freshness validation are REQUIRED before normal operation resumes.

Offline or Degraded Deployment Profiles: A deployment MAY define a separate offline profile using short-lived credentials. However, any profile that accepts credentials when authenticated revocation information is older than the MRI is non-conformant with Level 2 and MUST NOT be used for live legal or commercial authorization workflows that claim the five-minute revocation guarantee.

6.5 Credential Composition Patterns

This section defines the rules for credential delegation, multi-credential presentations, and conditional authorization.

6.5.1 Delegation Depth Limits

Operational Credential delegation depth MUST NOT exceed 1 (direct issuance only). A Grantor Agent may issue an Operational Credential to a Requestor Agent, but the Requestor Agent MUST NOT re-issue or delegate that credential to a third agent.



Figure 13: Delegation Depth Limits — Allowed vs. forbidden delegation chains

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

With a maximum depth of 1, circular delegation chains are structurally impossible. If future versions extend the maximum depth, a `delegationChain` field in the credential containing Agent DID history SHALL enable loop detection.

6.5.2 Multi-Credential Presentations

Workflows requiring authorization from multiple parties are composed by presenting multiple independent credentials within a single Verifiable Presentation. This preserves the simple binary validity model for individual credentials while supporting complex multi-party workflows.
Example -- Cross-Connect requiring A-side and Z-side authorization:

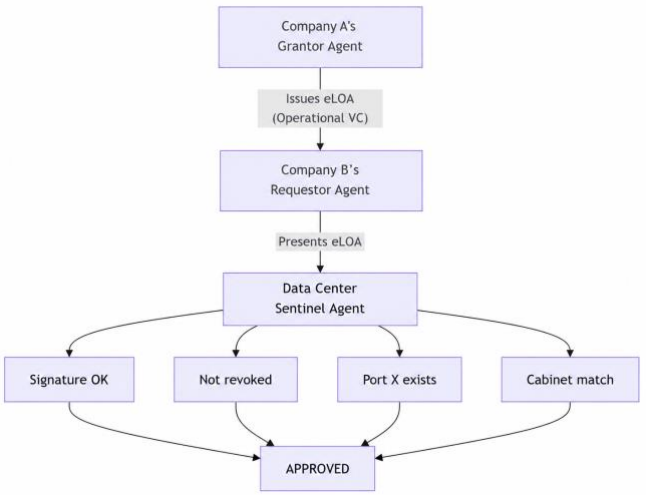


Figure 14: Multi-Credential Presentations — Dual-sided authorization workflow

The Sentinel's domain validation logic (Section 6.3, step 8) is where multi-credential requirements are enforced. The standard provides the mechanism (Verifiable Presentations can contain multiple VCs); the Sentinel's policy determines which credentials are required for a given operation.

6.5.3 Conditional Credentials

Operational Credentials MAY include conditional bounds in their `credentialSubject` fields. Sentinels validate these bounds during domain validation:

```
{
  "credentialSubject":                                                    {
    "maxBandwidthGbps":                                                  10,
    "serviceType":                                                "dark-fiber-lease",
    "requiresNocApproval":                                               true
  }
}
```

The Sentinel compares requested parameters against credential bounds and rejects requests that exceed the authorized scope.

6.6 Human Approval and Override Patterns

Detailed human approval, manual override, and escalation patterns are provided in Appendix B.3.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

7. Privacy and Selective Disclosure

7.1 Privacy Principles

Agent identity interactions are designed to minimize information disclosure. Two fundamental privacy guarantees apply:

Zero DID Correlation: When an Agent presents a credential to an external agent, the Agent's DID (the `credentialSubject.id` field) MUST NOT be disclosed. The Agent proves it holds a valid, unexpired credential without revealing its specific identity. This prevents Sentinels from building correlation profiles across multiple interactions. This requirement applies only for Agent to Agent interactions crossing Trust Boundaries.

Minimum Necessary Disclosure: Only the credential fields required by the Sentinel's verification policy are disclosed. All other fields are cryptographically hidden. The set of disclosed fields is determined by the Sentinel's disclosure policy, not by the Agent.

7.2 Selective Disclosure Mechanism

Selective disclosure is achieved through the use of cryptographic signature schemes that support zero-knowledge proof generation from signed credentials. The process operates as follows:

1. The Sentinel issues a challenge containing a nonce and a disclosure policy specifying which credential fields are required.
2. The Agent generates a zero-knowledge proof that demonstrates (a) it holds a credential signed by a recognized Governing Authority, (b) the required fields satisfy the disclosure policy, and (c) the proof is bound to the Sentinel's nonce -- all without revealing any fields not specified in the policy.
3. The Sentinel verifies the proof, extracts the disclosed fields, and proceeds with domain validation.

7.3 Privacy Limitations

This specification acknowledges the following privacy limitation:

Traffic Analysis: Zero-knowledge-proof-based privacy prevents identity correlation at the credential layer but does not prevent de-anonymization through traffic or timing analysis. An adversary monitoring network communications can potentially profile an Agent based on payload size, communication frequency, and connection patterns.

Recommended Mitigation: Implementations SHOULD consider randomized payload padding and traffic shaping (introducing random delays) to reduce the effectiveness of traffic analysis, accepting the associated performance trade-off.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

8. Regulatory Alignment

8.1 Alignment Framework

This specification is designed to facilitate compliance with the following regulatory frameworks. Compliance with any specific regulation remains the responsibility of the implementing organization; this specification provides the technical mechanisms that support compliance.

8.2 European Union Artificial Intelligence Act (EU AI Act, Regulation 2024/1689)

The EU AI Act, which entered into force on August 1, 2024, introduces tiered requirements for AI systems based on risk classification. Key provisions relevant to Agent Identity include:

EU AI Act Requirement	This Specification's Response
Transparency (Art. 50): AI systems interacting with humans must disclose their AI nature	Agent Identity credentials include an `agentType` field that identifies the holder as a software agent rather than a human actor. Implementing organizations must ensure appropriate disclosure at the interaction layer.
Traceability (Art. 12): High-risk AI systems must support logging and audit	Every Agent action is traceable through its DID to its Governing Authority and Principal. All credential lifecycle events are logged. DIDComm receipts provide non-repudiable transaction records.
Human Oversight (Art. 14): High-risk AI systems must enable human oversight	The Human-in-the-Loop (HITL) governance framework provides escalation pathways from automated Agent actions to human reviewers. The Principal, Governing Authority, and Agent accountability chain ensures a human is always identifiable.
Risk Management (Art. 9): Providers must implement risk management systems	The credential architecture enables granular risk management through scoped authorizations, real-time revocation, and fail-closed verification.
GPAI Obligations (Art. 51-56): Additional requirements for general-purpose AI models	Agents built on GPAI models must comply with the GPAI Code of Practice (see Section 2.1, Informative References). The `agentType` and `capabilities` fields in the NetworkAuthorityCredential support GPAI transparency requirements.

8.3 NIST AI Risk Management Framework (AI RMF 1.0, NIST AI 100-1)

The NIST AI RMF provides voluntary guidance for trustworthy AI systems. This specification aligns with NIST guidance as follows:

NIST AI RMF Function	This Specification's Response
GOVERN: Establish governance structures	Governing Authority model with centralized credential issuance and lifecycle management.
MAP: Identify and catalog AI risks	Agent Type classification and scoped credentials enable risk-appropriate authorization.
MEASURE: Assess and monitor risks	Maximum Revocation Interval (5-minute MRI), heartbeat liveness monitoring, and fail-closed Sentinel design provide continuous risk measurement.
MANAGE: Prioritize and respond to risks	Real-time revocation, emergency key rotation, and two-step decommissioning provide graduated risk response capabilities.

8.4 Digital Networks Act — DNA

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

The proposed EU Digital Networks Act introduces a modern, simplified, and harmonised legal framework for electronic communications networks and digital connectivity services across the European Union. In particular, the DNA aims to reduce regulatory fragmentation by introducing a **Single Passport authorisation** mechanism, allowing providers to notify in only one Member State and operate more easily across the EU.
This specification does not implement the DNA and does not grant any regulatory authorisation to provide electronic communications services. However, it supports a complementary technical layer for federated agent-based operations by enabling verifiable identity, scoped authorisation, trusted information exchange, and accountability across operators.

Digital Network act Requirement	This Specification's Response
Single Passport authorisation: providers should be able to rely on a simplified notification mechanism in one Member State to support pan-European operations.	W185 applies a similar trust-portability principle at the agent level: a recognised Agent Identity and its associated credentials may be accepted by multiple operators within a federation, subject to Sentinel verification and trust-domain policies.
Harmonised authorisation framework: the DNA seeks to reduce fragmentation in provider authorisation rules across Member States.	W185 harmonises how AI Agents prove identity, role, capabilities, credential validity, and authorisation scope across multi-operator workflows.
Structured information exchange: the DNA requires information flows between providers, national authorities, and EU-level coordination functions.	W185 requires federation-level communication of trust anchors, credential status, revocation information, policy versions, audit identifiers, and Governing Authority metadata.
Accountability of providers: the provider remains identifiable and accountable under the applicable regulatory framework.	Every Agent action is traceable to an Agent DID, a Governing Authority, and a Principal, providing the technical evidence layer for accountability.
Cross-border operations: the DNA supports the provision of digital networks and services across Member States with fewer administrative barriers.	W185 supports cross-operator and cross-domain agent recognition by defining trust domains, cross-federation trust exchange, and Sentinel-based verification.
Regulatory simplification and market scalability: the DNA aims to make the EU connectivity market more scalable and less fragmented.	W185 reduces operational friction by replacing manual verification of authorisations, documents, and agent permissions with machine-verifiable credentials.

Note: *The DNA operates at the regulatory and public-law level, while W185 operates at the technical, operational, and federation-governance level. The two frameworks are therefore complementary but not equivalent: the DNA may create or simplify the legal right for providers to operate, whereas W185 provides the technical evidence infrastructure that allows agent-based actions to be verified, audited, and attributed within a trusted federation.*

8.5 SOC 2 Type II Alignment

The Agent Identity lifecycle produces the access control evidence, audit logging, and identity management documentation required for SOC 2 Type II trust service criteria, particularly:

- **CC6.1 (Logical and Physical Access Controls)**: Credential-based access with scoped authorizations.
- **CC6.2 (System Access Registration and Authorization)**: Agent bootstrapping protocol with Governing Authority approval.
- **CC6.3 (System Access Removal)**: Two-step decommissioning protocol (authority revocation followed by identity deactivation).
- **CC7.2 (Monitoring of System Components)**: Sentinel heartbeat and fail-closed liveness monitoring.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

8.6 Legal Enforceability Considerations

This specification provides the **technical evidence infrastructure** that supports legal accountability. It does not, by itself, create legally binding relationships. Legal enforceability of agent actions requires three complementary layers:

1. **Technical Traceability** (provided by this specification): Agent to DID to Governing Authority to Principal. Every action is cryptographically signed, timestamped, and non-repudiable.
2. **Contractual Framework** (outside this specification): The federation's membership agreements, service level agreements, and terms of service must establish that participants are legally bound by their agents' actions. For example: *"Operator A agrees that actions taken by its credentialed agents are binding upon Operator A under this agreement."*
3. **Regulatory Recognition** (evolving): The EU AI Act (Art. 12, Art. 14) requires traceability and human oversight for high-risk AI systems. Emerging regulations such as the GPAI Code of Practice are establishing the basis for recognizing machine-generated evidence as legally sufficient. Regulatory recognition of these mechanisms is an ongoing process.

Implementing organizations SHOULD ensure that their federation agreements explicitly address the legal status of agent-issued Operational Credentials and the binding nature of cryptographic signatures generated by credentialed agents.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

9. Conformance Requirements

9.1 Conformance Levels

Implementations of this specification may claim conformance at one of two levels:

Conformance levels are normative implementation claims; deployment profiles and maturity levels in Sections 1.5 and 1.6 are adoption guidance. A single-operator pilot MAY operate under the Basic Trust Profile while claiming only Level 1 conformance, whereas a production multi-operator federation SHOULD implement Level 2 conformance together with the Full Sovereign Identity Profile.

Level 1 -- Core Agent Identity: The implementation supports agent DID creation, credential issuance and storage, and credential presentation with basic signature verification. This level is appropriate for proof-of-concept pilots and internal testing environments where no real authorization orders are processed.

Level 2 -- Full Agent Identity: The implementation supports all Level 1 capabilities plus selective disclosure (zero-knowledge proofs), real-time revocation with MRI enforcement, fail-closed Sentinel behavior with tiered resilience (Section 6.4.2), the complete agent lifecycle (including emergency key rotation and two-step decommissioning), key storage tiering (Section 4.7), and cross-federation trust (Section 4.5). This level is required for any deployment processing real authorization requests or handling live authorization data in production environments.

Level	Purpose	Target User
Level 1 -- Core	Get started quickly; validate the model works	Engineering teams running PoCs, hackathons, internal pilots
Level 2 -- Full	Production-grade, legally defensible	Production deployments handling real authorizations

9.2 Mandatory Requirements Summary

The following table summarizes the normative requirements of this specification:

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Requirement ID	Requirement	Level 1	Level 2
REQ-ID-01	Agent MUST possess a DID conforming to W3C DID v1.1	Required	Required
REQ-ID-02	Agent DID Document MUST designate Governing Authority as controller	Required	Required
REQ-ID-03	Agent MUST hold a NetworkAuthorityCredential issued by Governing Authority	Required	Required
REQ-ID-04	All credentials MUST include a `credentialStatus` field	Required	Required
REQ-ID-05	All credentials MUST include a `schemaVersion` field	Required	Required
REQ-ID-06	Revocation propagation MUST complete within 5-minute MRI	--	Required
REQ-ID-07	Decommissioning MUST follow two-step process (authority then identity)	--	Required
REQ-ID-08	All credential lifecycle events MUST be logged with DIDComm receipts	Required	Required
REQ-ID-09	Emergency key rotation MUST be executable by Governing Authority	--	Required
REQ-ID-10	NetworkAuthorityCredential MUST include `keyStorageTier` field	Required	Required
REQ-ID-11	Operational Credential delegation depth MUST NOT exceed 1	Required	Required
REQ-ID-12	DID registry MUST expose a publicly accessible resolution endpoint	--	Required
REQ-ID-13	Governing Authority MUST expose an HTTPS BitstringStatusList endpoint	--	Required
REQ-ID-14	Governing Authority MUST meet qualification criteria per Section 4.6	Required	Required
REQ-ID-15	Cross-domain implementations MUST define Trust Domains, Federation Boundaries, and Inter-Federation Trust Exchange procedures	--	Required
REQ-ID-16	Sentinels MUST enforce credential and policy validation before protected resource tool invocation	Required	Required
REQ-ID-17	Implementations SHOULD define auditable human approval, override, and escalation patterns for sensitive actions	Recommended	Required

10. List of contributing Memebers

The following members of the MEF participated in the development of this document and have requested to be included in this list.

- DENSO
- Graphiant
- IometrixIrange
- MaiaEdge

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

- Orange
- Orchest
- Sage
- SecureIQLab
- Sparkle

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

Appendix A -- Glossary Cross-Reference

Term	Section Defined	Related Standards
Agent	3.1	NIST AI 100-1 Appendix A
Agent Identity	3.1	W3C DID v1.1 Section 1
Controller	3.1	W3C DID v1.1 Section 5.1.2
Credential Status Check	3.1	BitstringStatusList Section 2
DID	3.1	W3C DID v1.1 Section 3
Governing Authority	3.1	--
Grantor Agent	3.2	--
Identity Credential	3.1	W3C VC-DATA-MODEL-2.0 Section 4
MRI	3.1	--
Operational Credential	3.1	W3C VC-DATA-MODEL-2.0 Section 4
Principal	3.1	EU AI Act Art. 3(4)
Requestor Agent	3.2	--
Sentinel	3.1	--
VC	3.1	W3C VC-DATA-MODEL-2.0 Section 4
VP	3.1	W3C VC-DATA-MODEL-2.0 Section 6

Appendix B -- Glossary Cross-Reference

This appendix consolidates detailed guidance moved from the main body so that the normative sections remain concise while preserving the full technical content for implementation reference.

B.1 Key Storage Requirements

This section provides the detailed key storage requirements referenced by Section 4.7.
The authority an Agent can exercise is bounded by the security of its key storage. This section defines a three-tier model that links key storage security to credential authority.

Tier	Key Storage	Agent Types	Authority Level
Tier 3 -- HSM	Hardware Security Module (FIPS 140-3 Level 3+)	Governing Authority Orchestrator, high-value Grantor Agents	Full -- may issue credentials with no restrictions
Tier 2 -- TPM/TEE	Trusted Platform Module or Trusted Execution Environment	Production Sentinel Agents, standard Grantor/Requestor Agents	Standard -- may participate in all operational workflows
Tier 1 -- Software	Encrypted keystore (e.g., OS keychain, KMS-backed)	Dev/test agents, low-value Requestor Agents	Limited -- may only hold credentials with short expiration (24 hours or less) and reduced operational scope

The `NetworkAuthorityCredential` schema includes a `keyStorageTier` field (see Section 6.2.1) that Sentinels MAY use to enforce key storage requirements during verification. A Governing Authority SHOULD issue weaker credentials (shorter TTL, narrower capabilities) to Agents with lower key storage tiers.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

B.2 Open Agent Interconnection (OAI) Model

This section provides the non-normative Open Agent Interconnection model referenced by Section 4.8.
Implementations SHOULD keep each layer (e.g. identity, messaging, tool invocation, and business operations) separated so that a change in one layer does not require redesign of the others.

Layer	Description	Implementation examples	Use cases	Constraints
Identity layer	Answers who the Agent is, what it is authorized to do, and whether the credential is current	DID; VC	Agent DID Documents, NetworkAuthorityCredential, Operational Credentials, Verifiable Presentations, status lists.	Answers who the Agent is, what it is authorized to do, and whether the credential is current.
Agent to Agent messaging layer	Secure agent-to-agent messaging. (e.g. Present-proof, issue-credential, challenge/response, secure agent transport.)	DIDComm; A2A	Present-proof, issue-credential, challenge/response, secure agent transport.	Carries proofs and messages. It does not define business semantics by itself.
The tool access layer	Tool invocation and runtime access to capabilities.	MCP	Model-initiated calls to tools, services, or enterprise functions.	Protected MCP calls SHOULD be mediated by a Sentinel before tool execution.
Workflow interaction layer	Business operations	LSO APIs; Network API	Ordering, provisioning, assurance, cross-connect workflows.	Executed only after the identity, authorization, and policy layers have approved the operation.

B.3 Human Approval and Override Patterns

This section provides the detailed human approval, manual override, and escalation patterns referenced by Section 6.6.
Autonomous execution does not remove human control. Operators SHOULD define human-in-the-loop approval, manual override, and escalation patterns for commercial, financial, safety-critical, and regulatory-sensitive actions. These patterns can be implemented as additional credentials, approval tokens, policy decisions, or workflow tickets, but they MUST be auditable and bound to the Agent action they authorize.

- **Pre-Issuance Approval** applies before an Operational Credential is created. It ensures that a Grantor Agent cannot issue an authorization for sensitive, high-value, regulated, or unusual actions unless a human approval has already been obtained and recorded.
- **Inline Approval Checkpoint** applies during credential verification or workflow execution. It is used when the Agent presents a valid credential, but the requested action exceeds predefined policy thresholds, conflicts with business rules, or requires additional confirmation before execution can continue.
- **Manual Override / Break-Glass** applies in exceptional or urgent situations, such as incident response, service restoration, or regulatory obligations. It allows a controlled exception to normal policy, but only under strict conditions: the override must be short-lived, narrowly scoped, auditable, and subject to appropriate approval.
- **Escalation Path** applies when the system cannot make a safe automated decision. This includes cases such as revocation uncertainty, trust-domain mismatch, credential validation failure, data inconsistency, or policy ambiguity. In these cases, the Sentinel must fail closed

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.

and route the case to a human review process rather than allowing silent automated execution.

These control patterns may be implemented through the mechanisms listed in the following table.

Control Pattern	Typical Trigger	Required Mechanism	Failure and Audit Requirements
Pre-Issuance Approval	Sensitive, high-value, regulated, or unusual credential issuance	Signed approval token or workflow record bound to the proposed credential, approver, scope, and expiry	The credential MUST NOT be issued without valid approval. Approval and credential identifiers MUST be logged.
Inline Approval Checkpoint	A valid credential is presented, but the requested operation exceeds a policy or commercial threshold	Short-lived approval token or credential bound to the request parameters, nonce, session, and operation	Execution MUST pause or fail closed until approval is received. The decision and policy version MUST be logged.
Manual Override / Break-Glass	Incident response, service restoration, or regulatory emergency	Short-lived, narrowly scoped override token issued by an authorized approver; dual approval SHOULD be required for high-risk actions	The override MUST be denied if scope, expiry, approver, or justification is missing. A post-event review MUST be performed.
Escalation Path	Trust mismatch, validation failure, revocation uncertainty, inconsistent data, or policy ambiguity	Human-review case or workflow ticket containing the relevant evidence and policy context	No automated execution is permitted. Reviewer identity, evidence, decision, and outcome MUST be logged.

© Mplify Alliance 2026. All rights reserved. Any reproduction of this document, or any portion thereof, shall contain the following statement: "Reproduced with permission of Mplify Alliance." No user of this document is authorized to modify any of the information contained herein.