

ECZ-ID MCP Passport Technical Reference

External integration reference for representing, resolving and safely consuming MCP Passport proof.

MCP PASSPORT · TECHNICAL REFERENCE

PRIMARY READER

Developer, architect, security engineer, integration engineer and automated-system designer.

SECONDARY READER

AI platform engineering, IAM/security operations and technical evaluators.

WHEN TO READ THIS

You understand the product and need the exact technical object, proof model and safe integration pattern.

RELEASE

2026.10 · 15 October 2026

WHAT YOU WILL KNOW AFTER READING

You can integrate with MCP Passport identity, bindings, state, evidence and Resolver without over-inferring absent or stale data.

WHAT THIS DOCUMENT DOES NOT ESTABLISH

This reference is not an internal schema dump and does not create permissions, credentials or guarantees that are not represented in the record.

NEXT

Use the Operations guide for lifecycle changes, troubleshooting and support escalation.

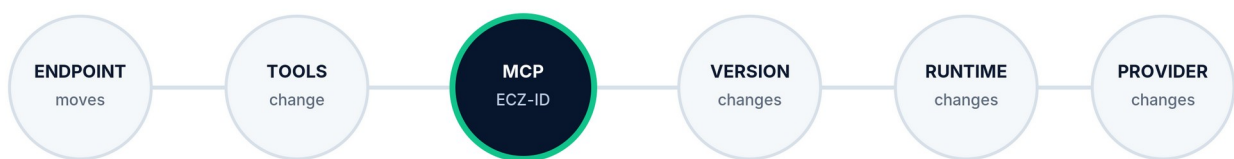
CANONICAL OBJECT

What the MCP ECZ-ID represents

An MCP Passport represents the intended logical MCP server or tool surface - not merely one transient URL, process, container or provider-native identifier.

One MCP identity across changing endpoints, tools and deployments

The server or tool surface can evolve while the canonical MCP ECZ-ID remains independently referenceable.



Persistent identity anchors a changing MCP surface

Endpoints, tool manifests, versions and provider context can change without treating the URL itself as the identity.

STABLE IDENTIFIER

The MCP ECZ-ID is the persistent reference. Endpoint, runtime, provider, tool manifest and version are contextual surfaces that can change.

PARENT AND OPERATOR RELATIONSHIP

Responsibility context is explicit and bounded

Where supported, the MCP record can relate the object to a Parent/organisation or operator context.

USE IT FOR

Confirming the expected responsible organisation/operator and understanding the published relationship around the MCP object.

DO NOT INFER

Universal assurance, tool safety, ownership beyond the published relationship, or permission for any caller to invoke tools.

MACHINE RULE

If your integration requires a specific responsibility relationship and it is absent, the requirement is not established.

SUPPORTED BINDING CLASSES

Connect stable identity to changing technical surfaces

Bindings are external references associated with the canonical MCP ECZ-ID according to the available state/evidence.

ENDPOINT

Current MCP server endpoint or well-known location where supported.

MANIFEST / TOOLS

Tool manifest, capability surface or machine-readable description where supported.

REPOSITORY / PACKAGE

Source, release, package or deployment references where supported.

PROVIDER ID

Cloud/provider-native service or workload identifier as a binding - never authority root.

AGENT RELATION

Agent↔MCP relationship context where present and useful to the interaction.

OTHER EVIDENCE

Other supported machine identities or evidence references.

STATE AND FRESHNESS

Point-in-time truth needs a freshness policy

The published state is useful only when consumers know whether it is fresh enough for the consequence they are about to allow.

CURRENT STATE

Parse the state that is actually returned. Do not convert a previous successful resolution into a permanent conclusion.

CHANGE CONTEXT

Where trust epochs/change evidence are available, compare current context with the previously observed state. Treat change as information to evaluate.

FRESHNESS ALGORITHM

Resolve → evaluate required fields/evidence → compare against your policy freshness threshold → re-resolve at the point of consequence when needed.

EVIDENCE SEMANTICS

Evidence is bounded by source, scope and freshness

Evidence can support claims about operator context, bindings, state or other MCP properties. It does not become an unlimited “trusted” verdict merely by being present.

SAFE

“This evidence reference supports the shown endpoint/tool relationship at the stated scope and freshness.”

UNSAFE

“Evidence exists, therefore this MCP server and all tools are safe and every claim is verified.”

ABSENCE RULE

If required evidence is absent, treat it as absent. Do not infer it from a tool manifest, vendor name, marketplace listing or prior cached result.

AUTHORITY AND INVOCATION

Identity is not tool permission

MCP Passport can support context for an authorization decision, but it does not itself grant an agent permission to call a tool.

Re-check the MCP surface before connection or consequence

Use MCP Passport proof as input to local policy - not as a universal safe/unsafe verdict.

**BOUNDARY**

MCP Passport identifies and contextualises the server/tool surface; it does not grant tool permission or certify behaviour.

ACTION-SPECIFIC AUTHORITY

Evaluate the relevant caller identity, delegation, tool policy, credentials and runtime controls for the actual invocation.

RESOLVER REPRESENTATION

Public proof and machine resolution

Resolver is the public, read-only proof surface. A customer or machine can independently re-check the MCP record without using the Dashboard control plane.

MACHINE-READABLE ROUTE

GET https://api.ecocitizenz.com/api/p/{ecz_id}.json

- Parse the stable ECZ-ID and Passport family/object identity.
- Check expected Parent/operator relationship if required.
- Read current published state.
- Evaluate endpoint/tool bindings and evidence relevant to the intended connection.
- Treat absent fields as not established.
- Re-resolve when your freshness policy requires it.

INTEGRATION PATTERN

A machine-safe pre-connection sequence

Use the Passport before connection or invocation when identity continuity, operator context or freshness matters.

1 RECEIVE

Obtain the MCP ECZ-ID from discovery, configuration, documentation or a trusted integration path.

2 RESOLVE

Fetch the current public Resolver record.

3 VALIDATE

Check family/object, operator, state and required bindings/evidence.

4 AUTHORIZE

Apply local authentication/authorization/to ol policy separately.

5 ACT

Connect, challenge or deny based on policy.

FAIL-SAFE BEHAVIOUR

If your policy requires a field or evidence item and it is absent, malformed, stale or mismatched, do not silently infer a positive result.

AGENT ↔ MCP RELATIONSHIPS

Reciprocal trust across machine actors

The agent and the MCP surface can each carry a resolvable identity. Policies can evaluate both sides and the relevant relationship/evidence rather than relying on one opaque self-assertion.

AGENT SIDE

Resolve Agent ECZ-ID, expected organisation responsibility, current state and action-specific authority context where relevant.

MCP SIDE

Resolve MCP ECZ-ID, operator context, endpoint/tool bindings, current state and evidence needed for the connection.

RELATIONSHIP

Reciprocal proof does not merge the two identities. Each remains its own canonical object with relationships evaluated at the interaction boundary.

SECURITY SIGNALS AND DANGEROUS CHAINS

Use warnings as bounded evidence

Some connected tooling may surface secret-exposure, dangerous-chain or other security indicators. The Passport can reference relevant evidence where supported, but the semantics remain bounded.

SUPPORTED USE

Use a current, scoped warning/evidence item to inform a specific policy decision or investigation.

DO NOT CLAIM

MCP Passport itself scans every secret, proves tools are harmless, or guarantees an invocation chain cannot become dangerous.

INTEGRATION LAW

Keep identity/proof semantics separate from the detector or policy engine that produced an evidence signal.

LIFECYCLE AND MUTATION MODEL

Change context, preserve identity

When the logical MCP surface remains the same, update its supported context instead of minting another ECZ-ID merely because infrastructure moved.

ENDPOINT MOVE

Update endpoint binding/evidence.

TOOL CHANGE

Update manifest/tool-surface evidence and re-check policy.

PROVIDER MOVE

Update provider-native bindings; preserve canonical MCP ECZ-ID.

OPERATOR CHANGE

Update supported organisation/operator relationship through the canonical control path.

DECOMMISSION

Publish/reflect the relevant lifecycle state; do not leave consumers with silently stale context.

FAILURE HANDLING

Do not fabricate identity when proof is missing

Integrations should distinguish “not found”, stale/missing evidence, mismatched family/object and transport/client failures.

- Not found: confirm the ECZ-ID and official Resolver route; do not manufacture a fallback identity.
- Wrong family/object: treat as mismatch, not success.
- Required operator/binding/evidence absent: requirement not established.
- Stale state/evidence: re-resolve or challenge according to policy.
- Network/client failure: distinguish from an authoritative “not found”.
- Malformed/unexpected data: fail safely when policy requires strict interpretation.

DEVELOPER PRINCIPLE

Be explicit about which fields your integration requires and what it does when they are absent.

DEVELOPER QUICK REFERENCE

The external contract in one page

Keep this model in mind when integrating MCP Passport into a client, gateway or policy path.

CANONICAL OBJECT

Logical MCP server/tool surface

STABLE IDENTIFIER

MCP ECZ-ID

PUBLIC PROOF

Resolver + machine-readable route

MUTATION

Dashboard/control path - not Resolver

NEVER INFER

Safety, blanket tool permission, absent evidence

SUPPORT

support@ecocitizenz.com

NEXT DOCUMENT

Operate the Passport over time

The technical integration is only useful if operational state, bindings and evidence remain current.

READ NEXT

ECZ-ID MCP Passport Operations, Lifecycle and Troubleshooting Guide

SHARED REFERENCE

ECZ-ID Resolver and Passport Verification Guide for safe human/machine interpretation.

SUPPORT

support@ecocitizenz.com