

ECZ-ID Service & Workload Passport Technical Reference

Technical semantics for service/workload identity, organisation/owner/operator responsibility, bindings, deployment/lifecycle state, evidence, Resolver representation and integration.

SERVICE & WORKLOAD PASSPORT · TECHNICAL REFERENCE

PRIMARY READER

Developer, architect,
security engineer,
integration engineer.

SECONDARY READER

Platform engineers,
automated-system
designers,
cloud/platform
engineering teams and
technical evaluators.

WHEN TO READ THIS

You understand the
product and need to
integrate or interpret
Service & Workload
Passport proof
precisely.

RELEASE

2026.10 · 16 October
2026

WHAT YOU WILL KNOW AFTER READING

You can identify the canonical object, supported
relationships/bindings, state/evidence semantics,
machine route, lifecycle and safe failure
behaviour.

WHAT THIS DOCUMENT DOES NOT ESTABLISH

This is not an internal schema dump and does not
promise fields or evidence that the current
canonical implementation does not expose.

NEXT Read the Operations Guide for lifecycle changes, stale context, duplicate prevention and troubleshooting.

CANONICAL OBJECT

What exactly does a Service & Workload Passport represent?

The canonical object is a service / workload / runtime object - not a provider-native workload ID, cloud resource ID, cluster/namespace reference, service account, container ID, runtime endpoint or credential used to access it.

CANONICAL IDENTITY

Persistent ECZ-ID representing the intended service/workload object across supported context changes.

EXTERNAL REFERENCES

Provider-native workload IDs, cloud account/project/subscription references, clusters, namespaces, service accounts, containers, runtime/deployment/service references and documentation are bindings/context.

OBJECT BOUNDARY

A new workload deployment does not automatically mean a new Service/workload identity; a genuinely different service/workload/runtime object should not inherit the same identity merely because it replaces the old one.

STABLE

ECZ-ID where object continuity holds.

MUTABLE

Bindings, deployment context, operator relationship, state and evidence.

IDENTITY GRAPH

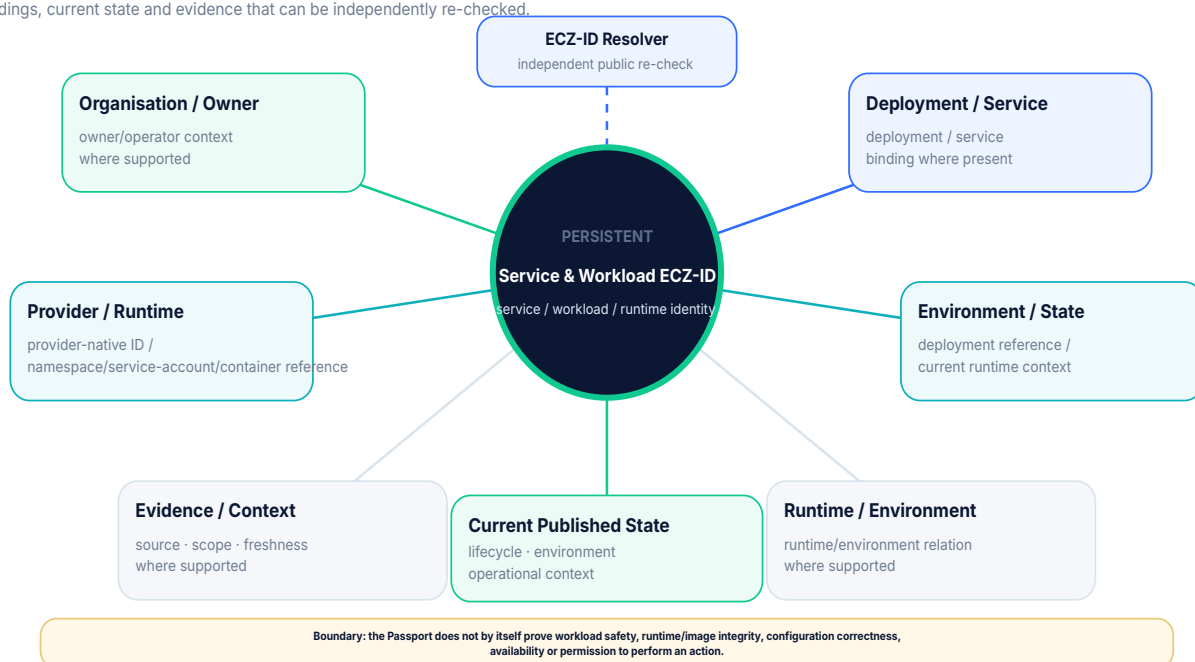
The Service & Workload ECZ-ID at the centre of bounded context

The graph should be read as relationships and evidence around a persistent identity – not as a universal trust score.

SERVICE & WORKLOAD PASSPORT PROOF MODEL

What a Service & Workload Passport can prove and contextualise

A persistent service/workload identity with bounded organisation/owner/operator responsibility, provider/runtime/deployment bindings, current state and evidence that can be independently re-checked.



GRAPH READING LAW

Each edge has its own scope. A deployment/service reference binding does not prove operator authority; organisation/owner/operator responsibility does not prove workload safety; evidence does not become a blanket “trusted” verdict.

MACHINE CONSUMER

Inspect only fields/relationships relevant to the decision.

HUMAN REVIEWER

Read source, scope and freshness before drawing a conclusion.

PARENT, ORGANISATION AND OPERATOR RELATION

Responsibility is represented as context, not universal assurance

Where present, a Parent/organisation relationship or owner/operator context links the Workload object to the organisation claiming responsibility for it.

CAN ESTABLISH

The record publishes the shown organisation/Parent or organisation/owner/operator responsibility context.

CANNOT ESTABLISH BY ITSELF

That every deployment, capability, authority/policy context, dependency or third-party statement has been independently verified.

AUTHORITY NOTE

Supported authority/delegation may be represented where present, but identity or organisation/owner/operator responsibility is not blanket authority for every runtime action, tool/API invocation or deployment decision.

USE RELATIONSHIP

As bounded accountability context.

USE EVIDENCE

To understand support, source and scope.

RE-CHECK

When organisation/owner/operator responsibility is material.

DO NOT INFER

Certification, workload safety or deployment integrity beyond evidence.

BINDINGS

How the persistent identity connects to changing provider and runtime context

Bindings can associate the Service & Workload ECZ-ID with supported external surfaces and identifiers.

CLOUD CONTEXT

Cloud account, project or subscription reference where supported.

RUNTIME IDENTITY

Provider-native workload ID, namespace, service account, container or workload reference where supported.

DEPLOYMENT

Deployment, service or runtime reference where supported.

ENVIRONMENT

Cluster, namespace, runtime or environment relationship where supported.

AUTHORITY CONTEXT

Supported authority/delegation or policy context where present and evidenced.

OTHER SUPPORTED BINDING

Additional provider, runtime, deployment, service or workload references represented by the canonical system.

SEMANTICS

A binding says the ECZ-ID record is associated with the external surface according to current evidence/state. It does not make that surface the identity authority root.

DEPLOYMENT, AUTHORITY/POLICY CONTEXT/CAPABILITIES AND CURRENT STATE

Separate persistent identity from mutable workload and runtime context

Where supported, current state can carry point-in-time lifecycle and deployment family context while the Service & Workload ECZ-ID remains stable.

STATE MAY CHANGE

Deployment, declared authority/policy context/capabilities where supported, runtime/environment relationship, deprecation/withdrawal context, operator relationship, bindings and evidence may change over time.

IDENTITY MAY PERSIST

The same intended service/workload object can retain the ECZ-ID across those supported contextual changes.

FRESHNESS

Do not cache a workload trust conclusion indefinitely. Re-resolve when current deployment, operator, runtime, authority/policy context/capabilities or binding context matters.

CONSUMER PATTERN

Read current state and evidence before a consequential deploy, start, update or invocation decision.

BOUNDARY

Current state is not a guarantee of code behaviour, vulnerability status or future safety.

EVIDENCE AND BOUNDED PROVENANCE

Evidence supports bounded claims - it is not magic provenance

Evidence can support claims about service/workload identity, responsibility, bindings, runtime/environment context and other supported relationships according to source, scope and freshness.

CLOUD CONTEXT EVIDENCE

Can support association to a cloud account/project/subscription or provider context where represented.

RUNTIME IDENTITY EVIDENCE

Can support association to a provider-native workload ID, namespace, service account, container or workload reference where represented.

DEPLOYMENT EVIDENCE

Can support bounded deployment relationships where represented.

RESPONSIBILITY EVIDENCE

Can support organisation/owner/operator context where represented.

PROVENANCE BOUNDARY

Do not infer signed-build provenance, reproducibility, artifact integrity, SBOM completeness or cryptographic attestation unless the specific evidence explicitly supports it.

AUTHORITY AND ADOPTION BOUNDARY

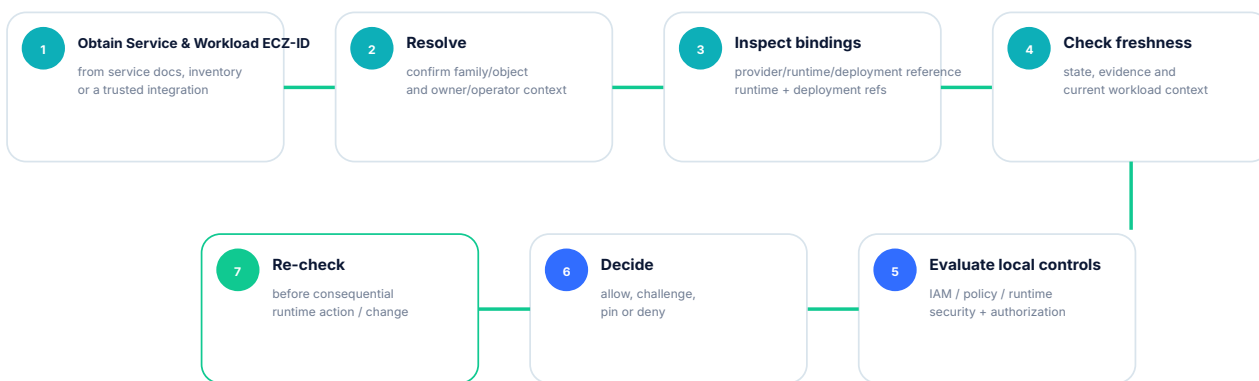
Identity is not authority to deploy, invoke or operate a workload

Some interactions require separate evidence and policy that a service/workload may be deployed, invoked, connected to, scaled, modified or otherwise operated in a particular environment.

PRE-USE VERIFICATION

Verify identity and context before a consequential service/workload action

Use Passport proof as one input; keep workload authentication, IAM/policy, runtime/image integrity and vulnerability/configuration review separate from identity proof.



Boundary: identity proof does not replace workload authentication, IAM/authorisation, runtime/image integrity, vulnerability management, deployment approval or runtime policy.

If required proof is absent, stale or mismatched, challenge or deny rather than infer.

DECISION BOUNDARY

Evaluate workload authentication, IAM/authorization, container/image integrity, vulnerability/configuration, deployment approvals, change control and runtime restrictions separately from service/workload identity.

PASSPORT CONTRIBUTION

Which service/workload is this and what bounded owner/operator, provider, runtime, state and evidence context is current?

LOCAL CONTROLS

Is this exact action permitted under IAM, policy, runtime and security controls for this environment?

RESOLVER REPRESENTATION

Public proof and machine-readable resolution

Resolver is the public, read-only proof surface. It is designed for independent re-check by humans and machines.

HUMAN USE

Inspect ECZ-ID, family/object, organisation/Parent/owner/operator context, current state, bindings and evidence where present.

MACHINE USE

GET
https://api.ecocitizenz.com/api/p/{ecz_id}.json
and evaluate only the returned fields relevant to policy.

READ-ONLY

Resolver does not mutate the Service & Workload Passport. Changes flow through the authenticated/canonical control path and become visible when the public record is updated.

RE-CHECK

Use Resolver again when operator, provider/runtime, deployment or evidence freshness matters.

NOT FOUND / ABSENT

Treat required missing data as not established; do not fabricate a fallback service/workload identity.

MACHINE-CONSUMER CONTRACT

A safe integration pattern

Treat Resolver output as structured identity/proof context, not as an all-purpose “trust = true” signal.

1 · PARSE IDENTITY

Read the stable ECZ-ID and confirm Service & Workload Passport family/object.

2 · CHECK RESPONSIBILITY

Evaluate organisation/Parent/owner/operator context if policy requires it.

3 · CHECK BINDINGS

Compare provider/runtime/deployment/service references relevant to the service/workload.

4 · CHECK STATE/EVIDENCE

Evaluate required current state, evidence scope and freshness.

5 · APPLY LOCAL CONTROLS

Container/image integrity verification, vulnerability/configuration review, authority/policy context, runtime/environment policy and execution controls remain separate.

6 · FAIL SAFELY

If required proof is missing, stale or mismatched, challenge/deny/pin/re-resolve.

NO UNIVERSAL SCORE

Do not reduce the record to a single “trusted workload” boolean unless your own policy defines and justifies that decision from specific fields/evidence.

LIFECYCLE AND WORKLOAD CONTINUITY

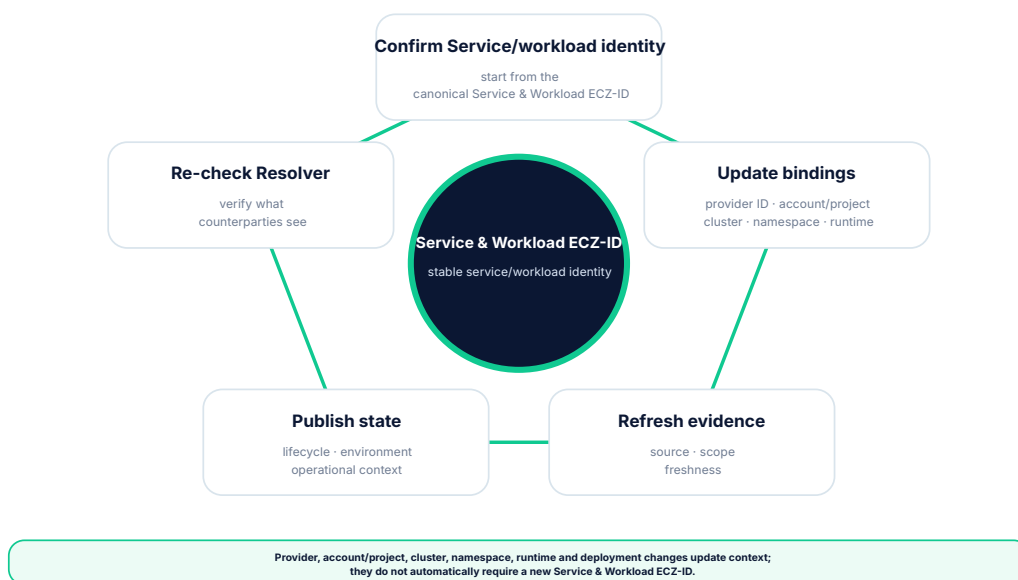
Keep the identity meaningful as provider and runtime context changes

A service/workload can move through active operation, major deployments, deprecation, provider/runtime migration, suspension, withdrawal and archival while retaining identity continuity where the object remains the same.

OPERATIONS LIFECYCLE

Operate one durable service/workload identity through change

Update supported provider, runtime, deployment, state and evidence context without fragmenting the canonical service/workload identity.



OBJECT CONTINUITY

Update supported bindings/state/evidence when provider/runtime/deployment context changes. Mint a distinct identity when the service/workload object boundary genuinely differs.

PRESERVE

Same intended service/workload object; context changes.

SEPARATE

Different service/workload object that replaces or forks the original.

PROVIDER-SPECIFIC BINDINGS

Provider and runtime identities are context, not canonical authority

Provider identifiers can be useful for navigation and proof, but ECZ-ID remains the canonical identity layer.

EXAMPLES OF PROVIDER CONTEXT

Provider-native workload IDs, cloud account/project/subscription references, cluster/namespace/service-account/container identifiers, runtime/deployment/service references and documentation URLs where supported.

ARCHITECTURE RULE

Provider-native identifiers become bindings/evidence/relationships - never the ECZ-ID authority root.

PORTABILITY

The Service & Workload ECZ-ID can remain referenceable even if the service/workload moves between providers, accounts/projects, clusters, namespaces or runtime environments.

USEFUL FOR

Cross-provider service/workload inventories, verification and migration.

STILL REQUIRED

Workload authentication, IAM/authorization, container/image integrity, vulnerability/configuration review and local runtime controls.

FAILURE HANDLING

Treat absence and ambiguity explicitly

Safe integrations distinguish not-found, missing-evidence, stale-context, wrong-object and transport/client errors.

NOT FOUND

Confirm ECZ-ID and distinguish missing record from network/client failure.

WRONG FAMILY/OBJECT

Do not treat another Passport family or service/workload as equivalent.

BINDING MISMATCH

Challenge if the provider/runtime/deployment/service binding does not match the intended service/workload.

EVIDENCE ABSENT

Treat required provenance/context as not established.

STATE STALE

Re-resolve or require fresh evidence according to policy.

WORKLOAD SECURITY / RUNTIME-POLICY FAILURE

Authentication, IAM/authorization, vulnerability/configuration or runtime-policy failure remains a separate control outcome.

FAIL CLOSED WHEN REQUIRED

If your decision requires a field/evidence and it is absent, stale or mismatched, do not infer it away.

WORKED INTEGRATION EXAMPLES

Apply the model to real provider and runtime patterns

Examples show how the same persistent identity can support different operational contexts without claiming more than the record proves.

PROVIDER OR CLUSTER MIGRATION

Same Service & Workload ECZ-ID; supported provider/runtime binding updates.

ACCOUNT/PROJECT/SUBSCRIPTION MOVE

Same service/workload identity; supported cloud-context bindings change.

MAJOR DEPLOYMENT

Same service/workload identity where object continuity is intended; deployment context updates.

ENTERPRISE SERVICE/WORKLOAD ACTION GATE

Resolve identity, operator/bindings/state/evidence, then apply local IAM/security/runtime policy.

SECONDARY RUNTIME/PROVIDER

A secondary provider/runtime identity is binding/context; canonical identity remains ECZ-ID.

WORKLOAD SUCCESSOR / FORK

A genuinely different service/workload should receive its own identity even if it replaces the original in a product stack.

EXAMPLE LAW

Examples illustrate the identity model; they do not promise every binding or evidence type is always present.

DEVELOPER QUICK REFERENCE

The external integration contract in one page

Use this page when implementing a verifier or integration around Service & Workload Passport proof.

OBJECT

Service / workload / runtime identity.

STABLE IDENTIFIER

ECZ-ID.

MACHINE ROUTE

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

KEY CONTEXT

Parent/operator · state ·
provider/runtime/deployment/service bindings ·
evidence.

MUST NOT INFER

Workload safety · configuration correctness ·
runtime/image integrity · availability ·
permission to act.

FAILURE RULE

Required absent/stale/mismatched proof →
challenge, deny, pin or re-resolve.

NEXT

Read the Operations, Lifecycle and Troubleshooting Guide before implementing workload-identity changes or production support workflows.