

Qualify the Hermes Agent connector

Use this page to decide whether one exact Hermes Agent connector artifact has enough evidence for a bounded release claim. It separates source-defined checks, deterministic provider fixtures, standalone-host behavior, path probes, and actual

SECTION	Connectors
TYPE	Connector
PROTOCOL	AIP 1.0
SOURCE	docs/connectors/hermes-agent/qualification/README.md

Current exact-artifact result: not established by this page. The pinned source contains test and qualification entry points, but this documentation review did not execute them. A historical isolated-live report applies to a different AIP artifact and cannot qualify the current standalone host.

Use evidence terms consistently

Term	Required evidence
Implemented	The behavior and contract exist in identified source
Conformant	The exact artifact passed the applicable conformance scenarios with retained results
Controlled-qualified	The exact artifact passed a named deterministic topology and failure matrix
Path-verified	One identified native, MCP, chat, or operator path produced its expected observations
Isolated-live verified	The exact artifact produced reviewed observations against identified Hermes and model-provider artifacts
Production-observed	Deployment-specific records show the behavior under the stated production boundary

Source presence proves implementation only. A test function or smoke binary proves that a gate exists, not that it passed for the release under review.

Bind every claim to one identity set

A qualification result is invalid unless its evidence binds all applicable values:

Identity class	Required values
AIP source	Commit, tree or complete dirty-source snapshot, and clean or dirty status
Connector artifact	Immutable image ID, component, version, source label, endpoint, and architecture
Contract	Manifest digest, 35-capability catalogue digest, schemas, and implementation-support map
Hermes upstream	Commit, tree, immutable image ID, API configuration, and state migrations
Model route	Provider, model identifier, credential scope, and relevant generation settings
Harness	Source digest, runner image, configuration digest, and selected cases
Runtime	Container engine, databases, trust policy, endpoint set, and network topology
Run	UTC interval, unique run ID, operator or CI identity, and evidence-root digest

The pinned source identities for this review are:

Source	Commit	Git tree
AIP	97be86e9efedf07ecf1783b03800f683f107fb04	2d83a382a312c601ecbbf5ed03f8af2b0218bdae
Hermes Agent	7426c09beee73bdf94d916015bac71384f6bc92	583423a976e761d3f076d2cf7f7859c13e40cc29

These source identities do not identify a built image or a completed run. A mutable tag, branch, filename, or successful command exit is not an immutable artifact identity.

Evaluate five independent gates

Gate	Source-owned entry point	What it can prove	What it cannot prove
Q1: connector conformance	<code>tests/frozen_conformance.rs</code>	Deterministic Hermes connector semantics through public connector and runtime boundaries	Container, fleet, or actual Hermes behavior
Q2: standalone image	<code>verify-product-images.sh</code>	Exact image identity, runtime user, endpoint, labels, inspect, history, and SBOM	Provider compatibility, image signature, vulnerability policy, or admission
Q3: controlled product fleet	<code>run-product-qualification.sh</code> and <code>verify-product-fleet.sh</code>	Admission, selected operations, approval, replay, routing, fail-closed behavior, and recovery against fixtures	Actual Hermes or model-provider compatibility
Q4: bounded path probes	Five Hermes smoke binaries	The exact native, MCP, chat, operator, or delegation path selected by a retained run	Broad conformance or unobserved capabilities
Q5: isolated live	<code>aipd-hermes-operator-smoke</code> plus a campaign manifest	Selected behavior against identified Hermes and model-provider artifacts	Arbitrary models, every capability, scale, or production readiness

Record a result for every gate required by the release claim. One gate cannot silently substitute for another. A release may omit an external-provider gate only when its claim states that exclusion.

Review deterministic conformance evidence

The frozen driver returns 13 scenario results. It exercises:

- authenticated actor, tenant, correlation, and opaque credential projection;
- input and delegated-output schema rejection;
- concurrent replay suppression, input-collision rejection, and stable delivery identity;
- retry classification, backoff, exhaustion, and dead-letter settlement;
- pre-dispatch, in-flight, and late cancellation boundaries;
- ordered chunks, one terminal chunk, and bounded stream truncation;
- read retry fields, unsafe uncertainty, and secret redaction;
- independent approval authority, quorum, retained evidence, and one resume;
- delegation receipts, correlation, and removal of raw secrets;
- restart recovery, an outcome-unknown start, and cursor-based reconnect.

Transactions and webhooks are explicitly not applicable in this driver. The source test accepts only a passing report with 13 checks, but that assertion is not a result artifact. Retain the report, selected binary identity, process status,

timestamps, and redacted diagnostics from the exact run.

The connector also has focused operator tests for:

- exactly one Hermes run under concurrent replay;
- rejection when one Action ID is reused with changed input;
- generation-bound and idempotent approval resume;
- no replay of an unbound start after process restart;
- cancellation confirmed by terminal Hermes state;
- server-assigned child Action IDs and exact native child contracts;
- delegated-result contract, tenant, and transport-owner verification;
- delegation that remains disabled until explicitly allowlisted.

Those tests strengthen the deterministic gate. They do not replace the frozen report, standalone topology, or real provider observations.

Review the standalone image gate

The image verifier builds `aip-host-hermes-agent` with the common host Dockerfile. It runs the help surface under read-only, capability-dropped, and no-new-privileges controls, then requires:

- an immutable `sha256` image ID;
- runtime user `10001:10001`;
- entrypoint `/usr/local/bin/aip-connector-host`;
- exact source revision, version, and component labels;
- retained image inspect, full history, and SPDX JSON SBOM artifacts.

Review the Hermes image artifacts rather than relying on the shared six-host summary. An SBOM is an inventory. It is not an image signature, vulnerability decision, license decision, provenance attestation, or provider qualification.

Review the controlled product-fleet gate

The source-owned product fleet uses deterministic upstream fixtures. Its Hermes baseline requires:

- exactly 35 admitted capabilities for the identified connector version;
- one `models` read and one `jobs_list` read through the gateway;
- one approval-governed `job_create` with stable Action and idempotency

identities;

- replay of the completed mutation through the same governed lifecycle;
- fixture audit records for `GET /v1/models`, `GET /api/jobs`, and

`POST /api/jobs`;

- authorization and idempotency markers at the fixture boundary.

The outer matrix includes the Hermes host in graceful restart, `SIGKILL`, expired-lease fail-closed, restart recovery, shared PostgreSQL outage, and database recovery cases. It also checks that the Hermes runtime database has an independent owner and the required runtime tables.

This gate exercises three Hermes capability IDs against a controlled fixture. It does not exercise the other 32 capabilities, a real Hermes deployment, model inference, streams, sessions, operator work, first-class delegation, or

provider-version drift. Label it controlled product-fleet qualification.

Review each bounded path probe

Use a path probe only for the claim it actually observes:

Entry point	Required retained observation
<code>aip-hermes-smoke</code>	Direct endpoint health plus AIP handshake and endpoint-qualified health actions
<code>aip-hermes-chat-smoke</code>	One endpoint health check and one non-empty chat result through the in-process AIP path
<code>aipd-hermes-chat-smoke</code>	Deployed manifest exposure, native handshake, health, and exact prompted chat content
<code>aipd-hermes-mcp-smoke</code>	MCP tool discovery plus structured health and exact prompted chat content
<code>aipd-hermes-operator-smoke</code>	Independent approval, operator stream evidence, exactly one child action, scope denial, and exact delegated result

Record the endpoint set, tool and capability IDs, prompt marker, Action IDs, approval identity, provider handles, terminal state, and raw result references. Do not combine separate runs into one apparently continuous result.

A chat marker proves the selected route returned expected content once. It does not prove model quality, deterministic generation, every transport, provider durability, or the other capability families.

Review isolated-live evidence

A current isolated-live campaign needs actual Hermes and model-provider artifacts, not fixture-compatible routes. Use at least two independently named Hermes endpoints when the claim includes endpoint isolation.

The operator campaign should verify:

1. unauthenticated profile access fails at the intended boundary;
2. every endpoint discovers only its allowlisted AIP tools;
3. an operator Action enters pending approval before execution;
4. an independent authenticated principal approves the immutable request;
5. structured run chunks contain the expected tool start and completion;
6. exactly one protocol-assigned child Action performs the downstream work;
7. the child result matches an independently read expected record;
8. an out-of-scope delegation fails closed;
9. the allowed delegation returns the same child Action and durable result;
10. cancellation, restart, or uncertainty cases required by the claim settle

under their original identities.

Hermes must not receive credentials for the downstream business system. The authoritative delegated result comes from the native AIP lifecycle, not model prose.

The retained 11 July 2026 report describes a historical pass for two endpoints. It used a different AIP artifact, has no complete machine-readable result or cleanup record, and cannot establish a current pass. Preserve it as `historical` and review its exact limits in the dedicated record.

Account for all 35 capabilities

Maintain one generated or machine-validated row per published capability:

Field	Required value
Capability ID	Exact endpoint-qualified admitted identifier
Contract evidence	Schema, method, route, risk, approval, retry, streaming, and transaction mapping
Deterministic evidence	Scenario and result artifact, or an explicit gap
Controlled-fleet evidence	Case and artifact, or <code>not_run</code>
Native and MCP evidence	Path probe and artifact, or <code>not_run</code>
Isolated-live evidence	Provider identities, case, and artifact, or <code>not_run</code>
Failure evidence	Auth, validation, rejection, uncertainty, cancellation, restart, and recovery cases exercised
Side-effect control	Action, idempotency, approval, provider handle, reconciliation, and cleanup as applicable
Review decision	Accepted scope, limitation, owner, reviewer, and time

Every row needs contract evidence. Live mutation of all operations is not a default requirement and may create unsafe model, session, job, or operator effects. Select live cases by unique risk and behavior, then publish every unobserved capability as a gap.

Retain a complete evidence set

A current qualification record should contain:

1. immutable AIP source, host image, manifest, schema, Hermes, model-route, harness, dependency, topology, and run identities;
2. a gate manifest naming every required case and invariant;
3. raw process exits, structured per-case results, and redacted diagnostics;
4. admission, catalogue, implementation-support, assignment, and route snapshots;
5. image inspect, history, SBOM, and separate supply-chain decisions;
6. native envelopes, MCP tool metadata, provider requests, and correlations without credentials or sensitive prompt content;
7. durable Action, approval, idempotency, lifecycle, chunk, operator-binding, child-action, and provider-state observations;
8. failure and recovery ordering, including process, lease, database, and provider events;
9. a generated 35-row capability coverage matrix;
10. checksums for every retained artifact and one evidence-root checksum;
11. verified cleanup, residual-resource inventory, and retained exceptions;
12. a summary that states scope, result, exclusions, reviewer, and UTC time.

A summary file's presence is not a pass. Review every referenced artifact and ensure that later recovery did not hide an earlier failed invariant.

Assign results without ambiguity

Result	Use when
passed	Every required invariant passed for the exact identity set, artifacts are complete and redacted, gaps match the claim, and cleanup was verified
failed	A required invariant failed, identity mismatched, evidence was corrupted or leaked, or cleanup violated the campaign contract
blocked	A required dependency or authority was unavailable before the invariant could be evaluated
not_run	The gate was not executed for this identity set
historical	A prior result belongs to a different source, artifact, topology, or evidence standard

Do not collapse `blocked`, `not_run`, or `historical` into `passed`. One gate can pass while the broader release claim remains unestablished.

Publish only the supported claim

Acceptable result language names the evidence boundary:

At the recorded time, the identified AIP, Hermes Agent, and model-route artifacts passed the listed deterministic, standalone-fleet, and selected isolated-live cases in the retained topology. The coverage matrix lists every unobserved capability and excluded production property.

Do not publish `fully compatible`, `all 35 capabilities live verified`, `exactly once`, or `production-ready` unless the retained campaign defines and proves each phrase under the stated deployment.

Reviewer checklist

- ☐ Every source, image, manifest, Hermes, model-route, harness, and run identity is immutable and mutually consistent.
- ☐ Required gates are explicit; no fixture result is labeled live.
- ☐ The 35-row coverage matrix matches the admitted catalogue.
- ☐ Frozen-conformance and standalone-fleet reports are complete.
- ☐ Native and MCP path results name their exact route and endpoint.
- ☐ Operator and delegation evidence binds approval, child identity, lifecycle state, provider handle, and authoritative result.
- ☐ Cancellation and uncertain outcomes are reconciled without replay.
- ☐ Image inventory is not presented as signature, scan, or provenance.
- ☐ Artifacts are redacted, checksummed, access-controlled, and retained for the claim lifetime.
- ☐ Cleanup and residual resources are independently recorded.
- ☐ Final wording states time, scope, result, gaps, and exclusions.

Related documentation

- [Conformance and qualification](#)
- [Connector fleet qualification](#)
- [Live-product qualification](#)
- [Historical Hermes Agent isolated-live result](#)
- [Hermes Agent capability index](#)
- [Release artifacts and verification](#)