

Troubleshoot the Hermes Agent connector

Use this runbook to trace a Hermes failure to the signed host, pinned route, endpoint credential, provider request, stream, operator binding, or delegated child. End with one explicit decision: correct, wait, retry a read within policy, rec

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

Pause new model, tool, session, job, operator, and delegation mutations when the provider outcome is uncertain. Preserve the original Action ID, idempotency key, route assignment, endpoint ID, provider handle, and durable state before changing either deployment.

Trust the right error surface

Identify the surface before interpreting a code:

Surface	Evidence shape	Trust decision
Pre-envelope host rejection	Unsigned <code>schema.envelope.invalid</code> or <code>envelope.decode.invalid</code> HTTP error	Trust only the controlled TLS path; no valid envelope exists to sign
Connector-host response	Signed AIP error envelope with a <code>ProtocolError</code>	Verify host signature, sender, recipient, and correlation
Durable Action lifecycle	<code>ActionResult</code> plus events, chunks, and typed links	Read this even when the original HTTP exchange was interrupted
Process startup	Error text and non-zero exit	Compare exact configuration, files, storage, and admission; no protocol code exists yet

The host can also return unsigned `connector_host.response_signing` when it cannot sign a response. Treat that as an integrity failure and read the authoritative Action state through the central gateway.

Automate on `code`, `category`, `retryable`, `uncertain_outcome`, remote status, and durable state. Message text is diagnostic, not a stable interface. The Hermes connector does not populate provider request ID or provider operation fields at this revision.

Choose the first check

Symptom	First check	Safe first action
Capability is absent	Authenticated tenant catalogue and enabled binding	Correct discovery or admission; do not call the host directly
Host returns 401	<code>connector_host.authentication</code> versus <code>connector.hermes_agent.authentication</code>	Repair only the failed trust layer
Host returns 403	Signed route, credential revision, approval, tenant, or provider code	Re-resolve the exact authority boundary; do not bypass it
<code>/health</code> works and <code>/ready</code> fails	Drain, lease, runtime, and connector fields	Follow the failing readiness owner
Basic health works but protected reads fail	Endpoint ID, token, Hermes policy, and model route	Keep model and tool traffic paused
Read gets 429, 5xx, transport, or deadline failure	Read-only contract, retry flag, and budget	Retry only through the same governed Action policy
Chat, mutation, or operator gets an ambiguous failure	<code>uncertain_outcome</code> plus AIP and provider state	Reconcile; do not submit a replacement
Stream closes or stops publishing	AIP chunks, cursor, terminal flag, and provider state	Resume evidence collection; do not infer completion
Operator returns <code>outcome_unknown</code>	Source Action and durable binding	Preserve the binding; never replay the start or command
Delegation does not settle	Exact child Action, owner, tenant, queue, and lifecycle result	Wait on the same child or fail closed
Session, response, run, or job is missing	Owning endpoint and its local state	Treat absence as a state-location problem, not no-effect proof

Diagnose trust and routing in order

Stop at the first failed layer:

1. `connector_host.authentication` means the request signer, configured

gateway DID, sender, principal kind, or trust domain did not match. Repair the signed gateway path.

2. `connector_host.replay` means the message failed the host replay boundary.

Preserve the rejected envelope and investigate duplicated or altered identity; do not weaken the replay policy.

3. `connector_host.route_mismatch` means the signed assignment does not match

this connector type, version, instance, replica, endpoint, manifest, tenant, binding, or fence. Discard it and resolve a current route.

4. `connector_host.credential_revision_denied` or

`connector_host.credential_revision_unavailable` means the route-pinned credential policy denied or could not read the revision. Restore that policy; do not select another credential silently.

5. `connector_host.approval_authorization_conflict` or

`connector_host.approval_authorization_unavailable` means imported approval authority conflicts with durable state or cannot be verified. Preserve the Action and approval.

6. `connector.hermes_agent.tenant_mismatch` means the verified AIP tenant does

not own the endpoint selected by the capability ID. Repair admission and binding; input cannot override the tenant.

7. `connector.hermes_agent.missing_credential` means a protected route has no

endpoint API key. Replace the correct owner-only secret and host process.

8. `connector.hermes_agent.authentication` means Hermes returned 401 or

403. Verify the endpoint ID, token, API-server policy, and intended deployment.

The remaining common host codes identify availability or execution rather than another credential layer:

Code	Decision
<code>connector_host.not_ready</code>	Wait for drain, lease, and connector readiness under the original Action policy
<code>connector_host.capacity</code>	Honor bounded backoff; do not bypass the admitted replica
<code>connector_host.execution</code>	Read the durable Action and underlying typed connector error before deciding
<code>connector_host.response_signing</code>	Isolate the host response and use central durable state; authenticity is incomplete

After a trust repair, require host readiness and route one protected models read through `aipd`. A direct Hermes request does not prove tenant, route, host-signature, or credential-revision enforcement.

Separate discovery, endpoint, and provider routing

Observation	Owner	Decision
Tenant catalogue omits the capability	Admitted version, implementation support, or binding	Correct the signed catalogue or use a published capability
Catalogue has it but no eligible replica exists	Lease, health, topology, capacity, or binding	Restore a ready admitted replica
Host reports route mismatch	Assignment versus immutable host identity	Resolve again; never edit the signed route
Capability selects the wrong endpoint ID	Application discovery choice	Stop and select the intended endpoint-qualified capability
Provider output belongs to another Hermes home or model route	Endpoint origin and deployment state	Treat as an isolation or endpoint-replacement incident
Basic health passes but protected model read fails	Provider auth or upstream model boundary	Repair the protected path before enabling model work

The connector never takes a base URL, endpoint ID, or token from Action input. Changing the endpoint descriptor or pointing an existing ID at a different Hermes home is a deployment change, not request recovery.

Interpret provider failures by operation safety

The connector retains the endpoint ID in its diagnostic and exposes the remote HTTP status when one was received. It does not expose the provider response body as structured redacted details.

Provider observation	Connector code	Read-only decision	Unsafe-operation decision
HTTP 401 or 403	<code>connector.hermes_agent.authentication</code>	Correct endpoint auth before a new read	Correct authority; do not replay accepted work
HTTP 429 or 500–599	<code>connector.hermes_agent.remote_temporary</code>	Runtime retry may be eligible within policy	Outcome is uncertain and automatic retry is disabled
Other non-success status	<code>connector.hermes_agent.remote_rejected</code>	Correct request or provider state	Follow the operation owner; some provider operations are non-atomic
Request, body, or SSE decode failure	<code>connector.hermes_agent.transport</code>	Runtime retry may be eligible within policy	Outcome is uncertain and automatic retry is disabled
Trusted execution deadline	<code>connector.hermes_agent.deadline_exceeded</code>	Runtime retry may be eligible within policy	Provider work may continue; reconcile policy
Runtime cancellation	<code>connector.hermes_agent.cancelled</code>	Read the original Action	Model, tool, message, or job effects can predate cancellation
AIP chunk publication failure	<code>connector.hermes_agent.stream_publication</code>	Preserve published chunks and source state	Provider work and missing AIP chunks are uncertain

Only a read-only Hermes capability can make a temporary or transport invocation failure retryable. Unsafe invocations mark 429, server, transport, decode, and deadline failures as uncertain and not retryable.

A connector-level `remote_rejected` does not override operation-specific behavior. For example, a session fork can change source and child rows before returning a title-validation `400`. Always consult the state owner before repeating a provider mutation.

Look up base connector codes

These codes are produced by the normal Hermes invocation path:

Code	Category and automatic retry	Decision
<code>connector.hermes_agent.configuration_or_input</code>	Permanent, false	Correct endpoint, URL, capability, identifier, input, or local client configuration
<code>connector.hermes_agent.tenant_mismatch</code>	Auth, false	Repair the verified tenant-to-endpoint binding
<code>connector.hermes_agent.missing_credential</code>	Auth, false	Install the endpoint key and replace the host process
<code>connector.hermes_agent.authentication</code>	Auth, false	Correct the selected endpoint token or provider access
<code>connector.hermes_agent.remote_temporarily</code>	Temporary, read-only conditional	Retry a read within policy; reconcile unsafe work
<code>connector.hermes_agent.remote_rejected</code>	Permanent, false	Correct provider state, then apply the operation-specific recovery rule
<code>connector.hermes_agent.transport</code>	Transport, read-only conditional	Retry a read within policy; reconcile unsafe work
<code>connector.hermes_agent.stream_publication</code>	Connector, false, uncertain	Preserve provider and AIP stream evidence; do not restart the operation
<code>connector.hermes_agent.cancelled</code>	Temporary, false	Inspect provider effects and the original Action
<code>connector.hermes_agent.deadline_exceeded</code>	Temporary, read-only conditional	Retry only a read; reconcile unsafe work
<code>connector.hermes_agent.operator_outcome_unknown</code>	Temporary, false, uncertain	Keep the source binding and do not replay the operator
<code>connector.hermes_agent.operator_state</code>	Temporary, false	Restore the exact PostgreSQL state and resolve contention or corruption
<code>connector.hermes_agent.operator_policy</code>	Permanent, false	Correct policy, approval, allowlist, binding, or idempotency collision
<code>connector.hermes_agent.error</code>	Connector, false	Retain the wrapped error and escalate an unexpected connector path

The category describes the emitted error. The Action and provider-state owner remain authoritative for replay safety.

Resolve stream failures

Parsed stream results have three additional non-retryable codes:

Code	Meaning	Decision
<code>connector.hermes_agent.stream_truncated</code>	The configured event limit ended consumption before terminal state	Read retained AIP chunks and provider state; do not rerun unsafe generation
<code>connector.hermes_agent.stream_incomplete</code>	The provider stream closed without an explicit terminal event	Treat execution as nonterminal or uncertain until another state owner settles it
<code>connector.hermes_agent.stream_failed</code>	Hermes sent a recognized terminal failure event	Retain the failure and inspect prior model, tool, message, and delivery effects

`stream_publication` differs from those result codes. It means a parsed provider frame could not be committed through AIP lifecycle storage. The provider stream may have continued while the durable AIP sequence stopped.

Use the original Action events and the last verified combined AIP cursor. Provider SSE IDs, event names, chunk IDs, and `run_id` values are not resume cursors. Closing or losing a follow connection does not cancel the Action.

For chat and Responses streams, do not start a second model turn merely to recover missing text. For structured run events, poll the retained provider run while it exists. For operator recovery, use the durable binding and bound run ID. Require a durable AIP result and provider terminal state where the operation defines one.

Resolve operator failures

The operator has durable result codes in addition to the base connector mapping:

Code	Meaning	Decision
<code>connector.hermes_agent.operator_start_rejected</code>	Hermes returned a definitive 4xx before accepting the start	Correct the request and create new reviewed work only after the original Action is terminal
<code>connector.hermes_agent.operator_failed</code>	The bound provider run reported terminal failure	Retain output and error; inspect tool and child effects
<code>connector.hermes_agent.operator_outcome_unknown</code>	Start response, run binding, or governed timeout left provider outcome unknown	Never replay POST /v1/runs; reconcile the retained binding
<code>connector.hermes_agent.approval_outcome_unknown</code>	Provider may have applied an approval response before its result was stored	Do not resend the choice; inspect the same run and approval generation
<code>connector.hermes_agent.cancellation_outcome_unknown</code>	Stop did not reach a confirmed provider terminal state	Keep cancellation intent and poll or escalate the same run

`operator_state` covers PostgreSQL read, compare-and-set, serialization, or contention failure.

`operator_policy` covers invalid policy, denied delegation, Action-ID collision, or missing binding. Correct the owning state or policy; do not bypass it with a different Action ID.

A bound `run_id` is still provider-local and volatile. Hermes restart or retention can make it return 404 `run_not_found`. That does not prove the model or a tool performed no effect. An unbound start is deliberately finalized as `outcome_unknown` and is never replayed automatically.

Follow the [governed operator lifecycle](#) for the source Action, two approval loops, restart, and cancellation decisions.

Resolve delegated-result failures

First-class delegation accepts only the exact durable child result. Use these codes to identify the failed invariant:

Code	Category and retry	Decision
<code>connector.hermes_agent.delegated_result_not_ready</code>	Temporary, true	Continue waiting on the same child within the original deadline
<code>connector.hermes_agent.delegated_execution_not_observed</code>	Policy, false	Hermes finished without creating the expected child; do not invent a result
<code>connector.hermes_agent.delegated_action_contract_mismatch</code>	Policy, false	Isolate the route; the child contract differs from the recorded request
<code>connector.hermes_agent.delegated_identity_unverified</code>	Auth, false	Restore transport-authenticated owner and <code>action:write</code> authority
<code>connector.hermes_agent.delegated_identity_mismatch</code>	Auth, false	Isolate the child created by the wrong Hermes principal
<code>connector.hermes_agent.delegated_identity_integrity</code>	Auth, false	Preserve the Action and trusted identity context; fail closed
<code>connector.hermes_agent.delegated_tenant_mismatch</code>	Auth, false	Isolate cross-tenant execution and preserve evidence
<code>connector.hermes_agent.delegated_tenant_expired</code>	Auth, false	Renew authority through policy; do not rewrite stored membership
<code>connector.hermes_agent.delegated_result_integrity</code>	Connector, false	Preserve queue and lifecycle records that disagree or lack a terminal result
<code>connector.hermes_agent.delegated_result_resolver_missing</code>	Connector, false	Restore the host's authoritative queue and lifecycle resolver

The retryable not-ready code means the same child has not settled. It never authorizes creation of another child. Model prose is not a delegated result.

Diagnose sessions, responses, runs, and jobs

Several provider resources are local to one Hermes endpoint:

Symptom	State owner	Decision
Stored response returns 404	Endpoint response SQLite or in-memory fallback	Verify the owning endpoint and original Action; do not repeat generation
Session route returns session_db_unavailable through provider 503	Endpoint state.db	Restore that database; mutation outcome still requires state inspection
Session chat succeeded but messages are absent	Model result versus transcript persistence	Keep the result, inspect logs, and do not claim a durable append
Session fork returns title-validation 400	Source and child session rows	Inspect both rows because the provider fork is not atomic
Structured run returns run_not_found	In-process run maps and retention	Preserve AIP evidence; do not recreate the run
Job mutation times out or is cancelled	Profile-local scheduler store	Read the job before retrying; persistence may have won the race
Job exists but no output was delivered	Scheduler claim, execution output, and delivery target	Separate stored schedule from run and delivery evidence

A successful host readiness probe covers only basic endpoint health. It does not open state.db, prove response-store durability, inspect scheduler claims, or retain structured runs.

Decide whether to retry

Apply this order and stop at the first unmet condition:

1. verify the signed response or read the durable Action result;
 2. confirm the exact endpoint-qualified capability and operation;
 3. determine whether the capability is read-only;
 4. for unsafe work, inspect its Action and provider-owned state; do not retry
- when the outcome is uncertain or no terminal no-effect evidence exists;
5. require connector implementation retry support and retryable: true;
 6. respect attempt, elapsed-time, deadline, and backoff budgets;
 7. let the runtime retry the same governed read rather than calling Hermes

directly.

The connector has no generic provider transaction ledger that proves not_committed for every Hermes model, tool, session, or job operation. Absence of a provider handle is therefore not permission to repeat unsafe work.

Resolve startup failures

Startup failures occur before a signed AIP error. Inspect the first process error and compare it with the exact configuration. Common owners are:

- empty, oversized, malformed, or unknown-field endpoint JSON;
- invalid or duplicate normalized endpoint IDs;
- non-HTTP(S) endpoint URLs or tenant mismatch;

- missing, unsafe, empty, or oversized endpoint secret files;
- invalid operator-policy fields, bounds, or allowlist relationships;
- PostgreSQL connectivity, role, schema, or recovery failure;
- a manifest above the 512-capability host ceiling;
- host identity, digest, endpoint, revision, or registry admission mismatch.

Correct one owner at a time. After every change, require `/ready`, registry lease agreement, and one protected model read. Do not combine endpoint, credential, database, artifact, and registry changes in one diagnostic restart.

Verify the resolution

Close the incident only when:

- the signed error or startup failure no longer occurs on the corrected path;
- `/ready`, the lifecycle lease, and tenant discovery agree;
- a protected read reaches the intended endpoint and credential;
- the original Action is terminal or remains explicitly paused as uncertain;
- every stream, operator, approval, cancellation, and delegated child has a justified state under its original identity;
- session, response, run, and job evidence comes from the owning endpoint;
- no endpoint key, raw sensitive payload, model credential, or personal data

entered the incident record.

Escalate with pinned source revisions, artifact and manifest digests, tenant-safe identities, timestamps, signed code and category, retry and uncertainty flags, and remote status. Include Action and provider state, stream cursor, operator binding, child integrity evidence, readiness fields, metric snapshots, and redacted diagnostics.

Related documentation

- [Errors and retry decisions](#)
- [Hermes authentication and endpoints](#)
- [Hermes configuration](#)
- [Recover the Hermes Agent connector](#)
- [Hermes streaming cursors and cancellation](#)