Hermes Agent connector

Use the Hermes Agent connector when an AIP application needs typed access to one or more Hermes API servers. The connector exposes health, discovery, chat, responses, structured runs, sessions, jobs, and governed operator work through endpo

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

The connector runs in its own `aip-host-hermes-agent` process. Product-neutral `getaip-server` routes admitted actions to that host; it does not embed Hermes product code, endpoint credentials, or model-provider credentials.

At a glance

Property	Reviewed value
Connector ID	`hermes-agent`
Standalone process	`aip-host-hermes-agent`
AIP workspace version	`2.0.0`
Reviewed AIP source	`d7cce13d1d555644d04a4d73c66c95b113737635`
Pinned Hermes source	`7426c09beee73bdff94d916015bac71384f6bc92`
Connector profile	`aip.connector.hermes_agent.v1`
Capability shape	`cap:hermes_agent:<endpoint_id>:<operation>`
Capability count	35 per endpoint
Effective standalone endpoint limit	14 under the fixed 512-capability host ceiling
Provider authentication	Per-endpoint Bearer token for every operation except basic health
Streaming	Chat, Responses API, run events, and session chat
First-class delegation	Disabled by default; explicit policy and allowlists required

The endpoint-file parser accepts between 1 and 1000 descriptors, but that is not the deployable catalog limit. The common standalone host rejects a manifest above 512 capabilities. At 35 capabilities per endpoint, 14 endpoints produce 490 capabilities and 15 produce 525. The reviewed binary exposes no setting that raises this ceiling.

These numbers describe source limits, not a scale qualification. Manifest byte size, provider capacity, storage, model service, and deployment policy may require a smaller boundary.

Choose this connector when

The connector fits an integration that needs to:

- discover models, skills, toolsets, and the Hermes API surface;
- submit synchronous or streaming chat and Responses API work;
- start, observe, approve, stop, and recover structured Hermes runs;

- keep conversations in persisted Hermes sessions;
- manage and trigger Hermes jobs through governed capabilities;
- use Hermes as a long-running AIP operator with durable approval and

cancellation state;

- route a first-class AIP delegation to a named Hermes endpoint under explicit

capability and scope allowlists.

Use another integration boundary when the task needs an arbitrary Hermes URL or method, direct access to a model provider, an ungoverned autonomous agent, or endpoints owned by different AIP tenants in one standalone instance.

Do not adopt the reviewed connector unchanged for a different Hermes revision. Re-audit the exact API-server routes and event shapes, then qualify the promoted connector and provider artifacts together.

Understand the deployment boundary

One standalone connector instance owns one AIP tenant and one immutable manifest containing 35 capabilities for each endpoint:

1. An operator configures one or more named Hermes endpoints in an

owner-controlled JSON file.

2. Every endpoint is bound to the standalone host's verified tenant. An

explicit different tenant causes startup to fail.

3. The host builds 35 endpoint-qualified capabilities for each descriptor.
4. Registry admission binds that exact manifest and artifact to the connector

instance and its replicas.

5. An authenticated caller submits a native AIP action to `getaip-server`.
6. The registry selects the tenant-owned Hermes instance and a healthy replica.
7. The host validates the action and maps its endpoint and operation to one

fixed Hermes method and path.

8. The per-endpoint credential is added inside the connector process, and the

result or stream returns through the native AIP lifecycle.

Adding a descriptor changes the manifest and therefore requires a new admitted connector version. Action input cannot select an unconfigured base URL, move an endpoint to another tenant, or supply a provider credential.

Choose a capability family

Each configured endpoint receives the same five families.

Family	Count per endpoint	Supported outcome
Health and discovery	6	Check health; list models, API capabilities, skills, and toolsets
Chat and responses	6	Run synchronous or streaming chat and Responses API work; read or delete a stored response
Structured runs and operator	6	Start, inspect, stream, approve, stop, or govern a durable operator run

Family	Count per endpoint	Supported outcome
Sessions	9	List, create, read, update, delete, inspect messages, fork, or chat in a persisted session
Jobs	8	List, create, read, update, delete, pause, resume, or run a durable job
Total	**35**	Complete published surface for one endpoint

The capability ID contains the normalized endpoint ID. Endpoint IDs are lowercase ASCII letters, digits, hyphens, or underscores after normalization. They are stable routing identities, not display labels.

Four operations are experimental in the reviewed manifest: skill discovery, toolset discovery, stored-response lookup, and stored-response deletion. The other 31 are marked stable. Stability does not replace provider qualification.

Keep four execution roles separate

Role	What executes	Authority boundary
Direct Hermes capability	The connector calls one fixed Hermes API route	The native AIP action contract and endpoint binding are authoritative
Hermes operator capability	A durable connector workflow starts or resumes a structured Hermes run	AIP approval, run binding, leases, and terminal provider state control settlement
First-class AIP delegation	The runtime routes a child action to `agent:hermes_operator:<endpoint_id>`	Delegation policy, scope and capability allowlists, approval, and native child lifecycle are authoritative
AIP MCP inside Hermes	Hermes uses deployment-configured MCP tools to call AIP	MCP authentication and tool allowlists are separate deployment responsibilities

The Rust connector does not write Hermes MCP configuration. Enabling operator delegation does not automatically give Hermes an AIP tool path, and configuring AIP MCP inside Hermes does not automatically authorize first-class delegation.

For a delegated action, the connector supplies the exact native child-action contract to the stable `aip_call` tool. The result resolver then reads the native durable AIP lifecycle. Hermes final prose is never accepted as the authoritative business result.

Protect endpoint credentials and tenants

Basic `GET /health` is the only operation that does not require endpoint authentication. Every other published operation requires the endpoint's Bearer token. The standalone binary reads an optional API-key file for each descriptor; invoking a protected operation without a configured key fails before a useful provider result can be returned.

Secret values are excluded from serialized endpoint configuration and debug output. The default HTTP client:

- accepts only configured `http` or `https` base URLs;
- does not follow redirects, so it will not forward a Bearer token to a new

origin;

- bounds connection and total request time;
- maps provider and transport failures to typed connector errors.

Use HTTPS and private service discovery outside a controlled loopback network. Give each tenant or trust domain independent Hermes and model-provider service accounts. Do not place endpoint keys, model credentials, or MCP tokens in AIP action input.

Apply operation safety by capability

The 35 operations do not share one retry or approval rule.

- Read-only discovery, status, event, session-read, and job-read operations are

low risk.

- Chat, response creation, run start, session changes, and most job changes are

medium risk because they can consume models or change provider state.

- Operator execution, destructive deletes, and immediate job execution are

high risk.

- Response deletion, run approval and stop, operator execution, session

deletion, and all state-changing job controls require AIP approval.

- Destructive, transaction-declaring, and operator operations require an AIP

idempotency key.

Do not interpret the AIP idempotency requirement as universal provider exactly-once behavior. The focused capability and operator pages define where the connector can replay a stored result, where it can resume provider state, and where an uncertain outcome requires reconciliation.

Treat streams and cancellation explicitly

The connector maps bounded Hermes SSE to native AIP stream chunks. It rejects oversized or malformed frames, limits the number of accepted events, and does not infer success merely because a connection closed.

Direct streaming operations end according to their operation-specific terminal events. Runtime cancellation stops local stream processing. Structured operator cancellation is stronger: the connector durably records intent, sends the Hermes stop request when a run ID is known, and polls for a terminal state. Only provider-confirmed `cancelled` becomes an AIP cancelled result. A racing completion or failure remains authoritative; an unconfirmed stop becomes outcome unknown.

Govern operator and delegation work

The operator capability uses structured runs rather than stateless chat. A durable profile-state record owns the start claim, run ID, stream progress, pending approval, resume command, cancellation intent, and terminal result. This prevents a restart from blindly starting the same Hermes run again.

A failure after Hermes may have accepted a start but before the run ID is stored becomes outcome unknown and is not automatically replayed. Approval resume commands are generation-fenced so an old response cannot approve a later prompt.

First-class delegation is off by default. Enabling it requires explicit capability and scope allowlists. Deployments that require delegated approval must provide the runtime approval store; the standalone host also installs an authoritative result resolver backed by native action and lifecycle stores.

Do not expose recursive `aip_delegate` to the model-facing Hermes tool set. The external AIP orchestrator owns the delegation graph and its cycle, depth, tenant, approval, and child-action invariants.

Deliberate exclusions

The reviewed connector does not provide:

- arbitrary HTTP forwarding or dynamic endpoint selection from action input;

- a cross-tenant endpoint set in one standalone instance;
- provider or model credentials inside protocol messages;
- redirect following for authenticated provider requests;
- default first-class delegation or unrestricted delegation policy;
- authoritative delegated results from model prose;
- connector-managed Hermes MCP profile writes;
- native AIP transactions or inbound Hermes webhooks;
- a guarantee that all 1000 syntactically accepted descriptors can be admitted;
- current production readiness for an unqualified artifact or provider.

Evidence status

Evidence	What it supports	What it does not support
Source and 35-route matrix	Implemented endpoint-qualified contract at the reviewed commit	Execution of an unbuilt artifact
Frozen conformance source	A deterministic 13-check gate, with transactions and inbound webhooks not applicable	A pass for the current artifact without a retained run
Controlled fleet source	Registration and selected fixture-backed calls in the standalone topology	Real Hermes or model-provider compatibility
Historical two-endpoint report	The exact older artifacts and seven recorded assertion groups	Current revision, rebuilt images, or another model route

Repeat deterministic, standalone-image, fleet, and isolated-live qualification for the exact connector, Hermes, model-provider, MCP, and deployment identities before publishing a readiness claim.

Plan the next task

Continue according to the task you need to complete:

- configure one endpoint and prove a read-only call before enabling mutations;
- bind endpoint credentials and every descriptor to the standalone host tenant;
- select one of the five capability families and read its exact contract;
- review the dedicated operator, delegation, cancellation, and MCP guidance

before enabling those higher-risk paths;

- use the operations guidance for deployment, health, recovery, and rollout;
- complete exact-artifact qualification before making a compatibility or

readiness claim.

The global references below own the shared fleet, MCP, and evidence rules.

Related documentation

- Connector documentation
- Pinned upstream baselines
- Connector fleet API
- Use AIP through MCP

- Historical Hermes Agent isolated-live evidence