

Chatwoot connector configuration

This reference describes the configuration accepted by `aip-host-chatwoot` at the reviewed source revision. A complete process combines six Chatwoot-specific settings with the shared connector-host identity, storage, routing, lifecycle, and o

SECTION	Connectors
TYPE	Connector
PROTOCOL	AIP 1.0
SOURCE	docs/connectors/chatwoot/reference/configuration.md

All product settings are read at startup. Only the shared non-secret credential-revision policy is re-read before provider side effects.

Configuration identity

Property	Value
Binary	<code>aip-host-chatwoot</code>
AIP workspace version	<code>1.0.0</code>
Reviewed AIP source	<code>97be86e9efedf07ecf1783b03800f683f107fb04</code>
Connector ID	<code>chatwoot</code>
Connector profile	<code>aip.connector.chatwoot.v1</code>
Pinned Chatwoot source	<code>8818d276b954ac4f84cffd8915c99f40e43804ed</code>
Full capability surface	150 catalogue operations plus three composites

Environment variables and flags contain non-secret values or filesystem paths. API tokens, webhook secrets, database credentials, and signing material belong in bounded owner-controlled files.

Chatwoot-specific settings

Environment variable	Command-line flag	Required	Default	Meaning
<code>AIP_CHATWOOT_BASE_URL</code>	<code>--base-url</code>	Yes	None	Fixed Chatwoot origin
<code>AIP_CHATWOOT_ACCOUNT_ID</code>	<code>--account-id</code>	Yes	None	Account inserted into every account-scoped route
<code>AIP_CHATWOOT_API_TOKEN_FILE</code>	<code>--api-token-file</code>	Yes	None	Owner-only API-token file
<code>AIP_CHATWOOT_WEBHOOK_SECRET_FILE</code>	<code>--webhook-secret-file</code>	No	Outbound-only host	Owner-only HMAC-secret file
<code>AIP_CHATWOOT_MAX_RESPONSE_BYTES</code>	<code>--max-response-bytes</code>	No	<code>16777216</code>	Maximum buffered provider response
<code>AIP_CHATWOOT_OPERATIONS_FILE</code>	<code>--operations-file</code>	No	All 150 catalogue operations	JSON suffix allowlist

The API-token and webhook-secret readers accept at most 16 KiB. Both files must pass the common secret-file type, ownership, permission, size, UTF-8, and non-empty checks.

Provider origin and account

The provider base URL must be an origin:

- HTTPS is required except for explicit loopback HTTP;
- credentials, a path prefix, query, and fragment are rejected;
- authenticated redirects are disabled;
- the connection timeout is 10 seconds;
- the overall provider request timeout is 120 seconds.

The account ID must contain 1 to 256 bytes, must not be blank after trimming, and must not contain control characters. It is inserted for `{account_id}` in every catalogue path and is not exposed as Action input.

Response-size setting

`AIP_CHATWOOT_MAX_RESPONSE_BYTES` accepts an integer from 1 through 134217728 bytes. Omission uses 16777216 bytes.

The connector checks a declared `Content-Length` and also stops while streaming when accumulated bytes exceed the active limit. An oversized response becomes `connector.chatwoot.response_too_large` and is non-retryable. For an invocation, the connector marks the external outcome uncertain because the provider may have completed before the response was rejected.

Operation allowlist file

The optional operations file is a JSON array of exact catalogue suffixes. It is a non-secret configuration file with a 128 KiB size bound.

```
JSON
[
  "account.get",
  "agent.list",
  "conversation.list",
  "message.list"
]
```

Rules:

- every value must match one of the 150 compiled suffixes;
- an unknown suffix fails startup;
- an empty array fails startup;
- duplicate suffixes collapse to one operation;
- omission enables all 150 catalogue operations;
- the three connector-level composites remain available and are not filtered

by this file.

The allowlist changes the discovered immutable manifest. Advance the connector version and admission material when its effective set changes.

Relevant shared host settings

The global connector-host reference owns the complete shared matrix. The following groups are required to understand Chatwoot behavior.

Identity and route binding

Setting	Role
<code>AIP_CONNECTOR_HOST_PUBLIC_ENDPOINT</code>	Externally reachable native endpoint ending in <code>/aip/v1/messages</code>
<code>AIP_CONNECTOR_HOST_TYPE_ID</code>	Admitted Chatwoot connector type
<code>AIP_CONNECTOR_HOST_VERSION_ID</code>	Immutable admitted version
<code>AIP_CONNECTOR_HOST_INSTANCE_ID</code>	Logical one-account instance
<code>AIP_CONNECTOR_HOST_REPLICA_ID</code>	Concrete process identity
<code>AIP_CONNECTOR_HOST_TENANT_ID</code>	Verified tenant owned by the instance
<code>AIP_CONNECTOR_HOST_MEMBERSHIP_ID</code>	Stable membership assertion
<code>AIP_CONNECTOR_HOST_EXTERNAL_ACCOUNT_ID</code>	Deployment record for the Chatwoot account
<code>AIP_CONNECTOR_HOST_EXTERNAL_ACCOUNT_SYSTEM</code>	Set to <code>chatwoot</code> with the external-account ID
<code>AIP_CONNECTOR_HOST_ARTIFACT_DIGEST</code>	Exact running OCI digest
<code>AIP_CONNECTOR_HOST_SECRET_PROVIDER_REF</code>	Non-secret provider reference

The external-account ID and system are an optional pair. A Chatwoot deployment should set them and validate the ID against `AIP_CHATWOOT_ACCOUNT_ID` outside the binary; the reviewed code does not compare those two settings.

Storage, signing, gateway, and lifecycle

Setting	Required or default
<code>AIP_CONNECTOR_HOST_BIND</code>	Default <code>0.0.0.0:8081</code>
<code>AIP_CONNECTOR_HOST_DATABASE_URL_FILE</code>	Required owner-only PostgreSQL URL
<code>AIP_CONNECTOR_HOST_SIGNING_SEED_FILE</code>	Required owner-only Ed25519 seed
<code>AIP_CONNECTOR_HOST_CONTROL_ENDPOINT</code>	Required fixed lifecycle endpoint
<code>AIP_CONNECTOR_HOST_CONTROL_DID</code> or <code>_FILE</code>	Exactly one control-plane DID source
<code>AIP_CONNECTOR_HOST_GATEWAY_PRINCIPAL_ID</code>	Required trusted gateway principal
<code>AIP_CONNECTOR_HOST_GATEWAY_DID</code> or <code>_FILE</code>	Exactly one gateway DID source
<code>AIP_CONNECTOR_HOST_TRUST_DOMAIN</code>	Required shared trust domain
<code>AIP_CONNECTOR_HOST_TLS_CA_FILE</code>	Optional private-PKI root
<code>AIP_CONNECTOR_HOST_MAX_IN_FLIGHT</code>	Default <code>32</code>
<code>AIP_CONNECTOR_HOST_LEASE_TTL_MS</code>	Default <code>30000</code>
<code>AIP_CONNECTOR_HOST_HEARTBEAT_MS</code>	Default <code>10000</code>
<code>AIP_CONNECTOR_HOST_DRAIN_TIMEOUT_MS</code>	Default <code>30000</code>
<code>AIP_CONNECTOR_HOST_HEALTH_TIMEOUT_MS</code>	Default <code>2000</code>
<code>AIP_CONNECTOR_HOST_CONTROL_TIMEOUT_MS</code>	Default <code>5000</code>

Shared credential ID, issuer, scopes, revision, and revision-policy settings are optional as a complete unit. They fence route authorization; they do not replace the required Chatwoot token file.

Choose outbound-only or webhook-enabled mode

Outbound-only

Omit `AIP_CHATWOOT_WEBHOOK_SECRET_FILE`. The process creates no `/webhooks/chatwoot` route, and shared central event delivery is not required for startup.

The API token remains required because discovery health and outbound capabilities use the account API.

Webhook-enabled

Set:

```
TEXT
AIP_CHATWOOT_WEBHOOK_SECRET_FILE=/run/secrets/chatwoot-webhook-hmac
AIP_CONNECTOR_HOST_EVENT_ENDPOINT=https://aipd.example.test/aip/v1/connector-events
```

The webhook secret enables `POST /webhooks/chatwoot`. The central event endpoint becomes mandatory; startup fails if the publisher is absent.

Related event settings are:

Environment variable	Default	Meaning
<code>AIP_CONNECTOR_HOST_EVENT_ENDPOINT</code>	None	Fixed central event ingress
<code>AIP_CONNECTOR_HOST_EVENT_ALLOW_HTTP</code>	false	Permit HTTP only for an explicitly controlled deployment
<code>AIP_CONNECTOR_HOST_EVENT_ALLOW_PRIVATE</code>	false	Permit a private destination explicitly
<code>AIP_CONNECTOR_HOST_TLS_CA_FILE</code>	None	Optional CA for event-delivery TLS

Network exceptions without an event endpoint fail configuration. The event destination is host-allowlisted, authenticated by the host signer, and pinned to the admitted event channel.

Request and payload limits

Boundary	Reviewed value
Native AIP request body, common default	4 MiB
Provider JSON body encoder	16 MiB
Query properties	128
Items in one query array	256
JSON body properties	512
Multipart fields	512
Multipart files	20
One decoded multipart file	25 MiB
All decoded multipart fields and files	64 MiB
One multipart field	1 MiB
One rendered path parameter	512 bytes
Provider response, default	16 MiB
Provider response, configurable maximum	128 MiB
Operations file	128 KiB
API token or webhook secret	16 KiB
Webhook request body	1 MiB
Webhook freshness window	300 seconds
Webhook replay identities	10000 per instance-scoped state set

The effective Action request ceiling is the smallest applicable boundary. A 16 MiB provider JSON encoder does not increase the common 4 MiB native envelope limit.

Webhook acceptance and event delivery

The webhook route requires:

- `X-Chatwoot-Delivery` with a non-empty delivery identity;
- `X-Chatwoot-Timestamp` parsed as an integer;
- `X-Chatwoot-Signature` in `sha256=<hex>` form;
- an original UTF-8 body of at most 1 MiB;
- HMAC verification over timestamp, a dot, and the exact body;
- a timestamp within five minutes of the host clock.

After authentication, instance-scoped durable profile state fences the delivery identity. The host maps a non-looping message to a deterministic AIP event, appends it to the durable event log, and enqueues it on `chatwoot-webhooks`.

A successful JSON response has this shape:

```
JSON
{
  "accepted_events": 1,
  "central_delivery": "centrally_acknowledged"
}
```

`central_delivery` is:

Value	Meaning
<code>centrally_acknowledged</code>	Central AIP acknowledged the durable batch before the response
<code>durably_queued</code>	The batch is durable locally and the background publisher owns delivery
<code>null</code>	No event was produced, such as for a connector-marked loop

An HTTP `200` with `durably_queued` does not prove downstream consumption. Storage failure returns `503` before successful acceptance.

Readiness behavior

The Chatwoot product health probe performs an authenticated `GET /api/v1/accounts/`. A successful response supports reachability and credential validity for that account route.

Host readiness also depends on durable storage, admitted identity, lifecycle lease, and common host state. It does not exercise all enabled operations, webhook delivery, approval, mutation safety, or provider compatibility beyond the health read.

Startup and reload boundaries

Value	Read
Provider origin and account	Startup
API token	Startup
Webhook secret and route mode	Startup
Response limit	Startup
Operation allowlist	Startup
Common identity, topology, endpoints, and lifecycle limits	Startup
Credential-revision policy file	Startup, then before provider side effects

Change any startup-owned value through a replacement replica and an admission or configuration revision. Do not edit a running process in place and assume its manifest, token, route, or webhook mode changed.

Configuration failures

Failure	Cause or decision
<code>invalid Chatwoot base URL</code>	Correct the origin; do not relax credentials, path, scheme, or redirect checks
<code>invalid Chatwoot connector configuration</code>	Check account length, response bound, operation suffixes, and non-empty allowlist
<code>configured Chatwoot credential is not valid UTF-8</code>	Repair the owner-only token file without printing it
Unknown operation suffix	Use an exact compiled suffix or advance the source and admission package
Webhook ingress requires event endpoint	Configure a valid central endpoint or remove the webhook-secret setting
Event destination policy rejects the endpoint	Correct TLS, host, and explicit network policy
<code>webhook.invalid_timestamp</code>	Supply an integer timestamp generated within the freshness window
<code>webhook.authentication_failed</code>	Verify exact raw body, timestamp, signature encoding, and secret ownership
<code>webhook.storage_unavailable</code>	Preserve the provider delivery ID and repair durable state before replay
<code>connector.chatwoot.response_too_large</code>	Reconcile provider state before changing limits or retrying a mutation

Related documentation

- [Chatwoot connector overview](#)
- [Authenticate and isolate a Chatwoot account](#)
- [Connector host configuration](#)
- [Connector host lifecycle](#)
- [Environment variables](#)