Dify connector configuration

This reference describes the product settings accepted by aip-host-dify at the reviewed source revision. A complete process combines these settings with the shared connector-host identity, storage, routing, signing, lifecycle, and optional

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

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

Configuration identity

Property	Value
Binary	`aip-host-dify`
AIP workspace version	`1.0.0`
Reviewed AIP source	`97be86e9efedf07ecf1783b03800f683f107fb04`
Connector ID	`dify`
Connector profile	`aip.connector.dify.v1`
Pinned Dify source	`f8d47616c15d0959f604f6a5e2e1d32d3108991b`
Frozen operation catalogue	33 app operations and 46 Knowledge operations

Environment variables and flags contain non-secret values or file paths. API keys, database credentials, and signing seeds belong in bounded owner-controlled files.

Dify-specific settings

Environment variable	Command-line flag	Required	Default	Meaning
`AIP_DIFY_BASE_URL`	`--base-url`	Yes	None	Fixed Dify origin shared by this host
`AIP_DIFY_API_KEY_FILE`	`--api-key-file`	No	Per-app files	Legacy key shared intentionally by all configured apps
`AIP_DIFY_APPS_FILE`	`--apps-file`	Conditional	Empty app set	JSON array of app descriptors
`AIP_DIFY_KNOWLEDGE_FILE`	`--knowledge-file`	Conditional	Empty Knowledge set	JSON array of Knowledge credentials
`AIP_DIFY_MAX_RESPONSE_BYTES`	`--max-response-bytes`	No	`8388608`	Maximum blocking response or complete event stream

At least one app or Knowledge descriptor is required. A host may serve only apps, only Knowledge, or both domains.

Configure application descriptors

The app file is a JSON array. Each object rejects unknown fields.

```
JSON
[
  {
    "id": "support-chat",
    "name": "Support chat",
    "mode": "chat",
    "description": "Published customer-support application",
    "api_key_file": "/run/secrets/dify-support-chat"
  }
]
```

Field	Required	Constraint
`id`	Yes	Non-empty, not `knowledge`, no control characters, valid inside a capability ID
`name`	Yes	Non-empty
`mode`	Yes	`workflow`, `completion`, `chat`, `advanced-chat`, `agent-chat`, or `agent`
`description`	No	Public discovery text
`api_key_file`	Conditional	Required unless the legacy global app key is configured

App IDs must be unique. The file may contain at most 1,000 descriptors and at most 4 MiB of JSON. Those are parser ceilings, not a promise that such a set fits the global admitted-manifest capability limit.

If `AIP_DIFY_API_KEY_FILE` is set, every app must omit `api_key_file`. If the global key is absent, every app must name its own file. Mixed or missing sources fail startup.

Configure Knowledge credentials

The Knowledge file is also a JSON array with unknown fields denied.

```
JSON
[
  {
    "id": "main-workspace",
    "name": "Main knowledge workspace",
    "description": "Reviewed workspace Knowledge API boundary",
    "api_key_file": "/run/secrets/dify-main-knowledge"
  }
]
```

Field	Required	Constraint
`id`	Yes	Non-empty, no control characters, valid inside a capability ID
`name`	Yes	Non-empty
`description`	No	Public discovery text
`api_key_file`	Yes	Owner-controlled workspace Knowledge API key

Knowledge IDs must be unique. The file shares the 4 MiB read bound and has a 1,000-descriptor parser ceiling. Each descriptor publishes 46 capabilities, so manifest validation normally constrains the practical count first.

The global application key never supplies a Knowledge credential.

Configure provider origin and keys

The Dify base URL must be an origin:

- HTTPS is required except for explicit loopback HTTP;
- credentials, path prefix, query, and fragment are rejected;
- authenticated redirects are disabled;
- provider connection establishment has a 10-second timeout.

The base URL does not identify a Dify source commit or image. Bind provider artifact identity and compatibility evidence outside this string.

Every API-key file is read as non-empty UTF-8 with a 16 KiB bound. Descriptor and configuration files may be public only when they contain no key material.

Set the response-size limit

`AIP_DIFY_MAX_RESPONSE_BYTES` accepts an integer from `1` through `67108864`. Omission uses `8388608` bytes.

The connector checks declared content length and accumulated stream bytes. The same active limit applies to blocking JSON, text, binary data before base64 projection, and the complete SSE response.

Increasing this setting does not increase the common native-envelope request limit, JSON encoder bound, decoded upload limit, per-frame SSE limit, or event count.

Apply relevant shared host settings

The global connector-host reference owns the complete matrix. These groups are required for Dify behavior.

Identity and route

Setting	Role
`AIP_CONNECTOR_HOST_PUBLIC_ENDPOINT`	Reachable native endpoint ending in `/aip/v1/messages`
`AIP_CONNECTOR_HOST_TYPE_ID`	Admitted Dify connector type
`AIP_CONNECTOR_HOST_VERSION_ID`	Immutable admitted version
`AIP_CONNECTOR_HOST_INSTANCE_ID`	Logical descriptor and task-store boundary
`AIP_CONNECTOR_HOST_REPLICA_ID`	Concrete process identity
`AIP_CONNECTOR_HOST_TENANT_ID`	Verified tenant owned by the instance
`AIP_CONNECTOR_HOST_MEMBERSHIP_ID`	Stable membership assertion
`AIP_CONNECTOR_HOST_ARTIFACT_DIGEST`	Exact host artifact digest
`AIP_CONNECTOR_HOST_SECRET_PROVIDER_REF`	Non-secret secret-provider reference

An optional external-account ID and system must appear together. Dify can contain several app and workspace identities, so deployment policy must define what that single external-account reference means.

Storage, signing, and lifecycle

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

Shared credential ID, issuer, scope, revision, and revision-policy settings are optional as one complete unit. They fence route authorization and do not replace Dify API-key files.

Stream callback

Streaming publication can use the common fixed callback endpoint and its explicit network policy:

Setting	Default
`AIP_CONNECTOR_HOST_CALLBACK_ENDPOINT`	None
`AIP_CONNECTOR_HOST_CALLBACK_ALLOW_HTTP`	`false`
`AIP_CONNECTOR_HOST_CALLBACK_ALLOW_PRIVATE`	`false`

Configure callback behavior according to the selected gateway topology. The product configuration does not make a callback mandatory for non-streaming operations.

Use the effective limits

Boundary	Reviewed value
Native AIP request body, common default	4 MiB
App or Knowledge descriptor file	4 MiB
App descriptors, parser ceiling	1,000
Knowledge descriptors, parser ceiling	1,000
API-key file	16 KiB
Provider JSON body encoder	8 MiB
Decoded upload file	32 MiB
Encoded upload field in capability schema	48 MiB
Query properties	128
Values in one query array	256
Body properties	512
Rendered path parameter	512 bytes
End-user value	256 bytes
Provider response, default	8 MiB
Provider response, maximum	64 MiB
SSE frame	1 MiB
SSE events per response	10,000
Concurrent credential health probes	32

The effective request ceiling is the smallest applicable limit. The 8 MiB JSON encoder and 48 MiB encoded upload schema do not enlarge the common 4 MiB native envelope limit.

Understand readiness

Product health checks every configured credential:

Credential	Probe
App	Authenticated `GET /v1/parameters`
Knowledge	Authenticated `GET /v1/datasets?limit=1`

The connector runs at most 32 checks concurrently and reports ready only after all succeed. Host readiness also requires durable storage, valid lease, non-draining lifecycle state, and common host readiness.

Readiness does not exercise app invocation, every mode-supported operation, mutation approval, stream termination, cancellation, binary response, upload, or provider compatibility beyond these reads.

Know startup and reload ownership

Value	Read
Provider origin	Startup
App and Knowledge descriptor files	Startup
Every Dify API-key file	Startup
Response limit	Startup
Common identity, topology, endpoints, and lifecycle limits	Startup
Profile-state task-store scope	Constructed at startup from instance ID
Credential-revision policy file	Startup, then before provider side effects

Change startup-owned values through a replacement replica and a new configuration or admission revision. Editing a file does not update the running manifest, key map, mode, task-store scope, or response limit.

Diagnose configuration failures

Failure	First decision
At least one app or knowledge credential is required	Supply a reviewed descriptor domain
Invalid Dify base URL	Correct the origin; do not weaken scheme, credential, path, or redirect policy
Unsupported app mode	Use one of the six exact mode strings
App ID `knowledge`	Rename the app because that prefix belongs to Knowledge routing
Duplicate descriptor ID	Make local routing identities unique
Missing or ambiguous app key	Choose either per-app files or one intentional global source
Invalid credential UTF-8 or size	Repair the owner-controlled file without printing it
Descriptor count exceeds 1,000	Split the deployment before manifest construction
Manifest admission exceeds its capability limit	Reduce descriptors or split immutable instances
Response bound is outside 1-64 MiB	Select a value inside the compiled range
Health fails for one credential	Identify its descriptor and probe; do not rotate unrelated keys

Related documentation

- Dify connector overview
- Authentication and credential scopes
- Connector host configuration
- Connector host lifecycle
- Environment variables