AIP security policy

If you believe you found a vulnerability in AIP, email with the subject prefix [SECURITY]. Do not open a public issue or publish credentials, customer data, exploit payloads, or affected deployment identifiers before remediation has been co

SECTION	Project and Releases
TYPE	Project policy
PROTOCOL	AIP 1.0
SOURCE	SECURITY.md

Send the smallest safe report that lets the maintainers understand the issue. You do not need a complete exploit or severity score before reporting a credible concern.

What to report here

Use this process for a suspected security weakness in AIP-owned material, including:

- native protocol validation, signatures, identity, authorization, replay,

approval, or audit behavior;

- runtime, storage, transport, MCP, A2A, or webhook implementations;
- connector SDK, connector fleet, admission, registry, host, or control-plane

behavior;

- one of the six maintained product connectors or its standalone host;
- generated AIP schemas, release automation, published images, attestations,

or source-boundary inventories;

- documentation or examples when following them could expose secrets, weaken a

security boundary, or misrepresent a required control.

Report uncertainty. If it is unclear whether a defect belongs to AIP, an upstream product, or a dependency, send the relevant boundary information and let the maintainers route it.

What belongs elsewhere

Use the owning project's security process for a defect that exists only in an upstream product or third-party dependency. Also report it to AIP when the AIP integration exposes the defect in a new way, bypasses an expected control, or ships an affected artifact.

Use these routes for other concerns:

- report harassment, retaliation, or other community behavior through the

private process in the code of conduct;

- propose an ordinary code or documentation correction through the

contribution workflow;

- use the deployment owner's incident process for an outage, abuse event, or

compromised account that does not demonstrate an AIP vulnerability.

The AIP project is not an emergency service. If people are in immediate danger, contact the appropriate local emergency service or platform safety channel.

Version and asset scope

The selected source proves the following version facts:

Target	Evidence at the reviewed revision	Reporting guidance
AIP `v1.0.0`	One source tag exists	Report suspected vulnerabilities
Reviewed mainline source	Workspace version remains `1.0.0` after the tag	Report issues against the exact commit
Earlier versions, forks, and modified images	No support or backport schedule is published in the selected source	Include the exact identity; response may require reproduction on an AIP-owned target

This table records source evidence, not a promise to backport every fix. The project has not published a fixed end-of-life schedule, patch cadence, or supported-version matrix beyond the version information above.

A protocol version alone does not make a deployment secure. Identity-provider configuration, credentials, storage, network policy, connectors, upstream services, and exact image digests remain deployment-owned boundaries.

Contain immediate exposure first

If a credential, signing key, token, database URL, customer payload, or other secret is exposed, begin the authorized deployment's containment procedure before expanding the investigation:

1. stop the unsafe operation when doing so will not increase harm;
2. revoke or rotate the exposed credential through its owning system;
3. restrict access to affected logs, artifacts, and accounts;
4. preserve the minimum records needed to understand the event;
5. report the suspected AIP weakness and state which containment steps occurred.

Do not send a live secret in the report. Redact it while preserving its type, location, scope, and relevant prefix or hash. If ordinary email is unsuitable for the remaining evidence, send a minimal initial report and request a safer transfer method. The project does not publish a PGP key in the selected source.

For active exploitation, use the deployment owner's incident-response process in parallel. A vulnerability report does not replace containment, customer communication, legal review, or regulatory obligations owned by the affected operator.

What to include

Include what is known and safe to share:

- the affected AIP crate, binary, transport, profile, connector, image, schema,

or documentation path;

- the exact source commit, release tag, image digest, package version, and

deployment topology when available;

- prerequisites, configuration, trust boundary, and attacker position;
- minimal reproduction steps and a redacted request, payload, trace, or log;

- observed behavior and the behavior you expected;
- likely impact on confidentiality, integrity, availability, authorization,

tenant isolation, audit evidence, or supply-chain integrity;

- whether credentials, customer data, model output, or external provider state

may be affected;

- whether exploitation is active or public and which mitigations are already

known;

- a safe contact method and whether you want public credit.

Do not collect unrelated customer records, broaden access, establish persistence, or prove impact by changing external state. A narrowly scoped report is preferable to a more damaging demonstration.

How the project handles a report

The maintainers use the following process:

1. A maintainer acknowledges the report when practical and establishes the

next communication step. No fixed response-time SLA is published.

2. Access is limited to people needed to assess the issue. Anyone with a

material conflict should recuse from the decision.

3. The report is checked against an AIP-owned revision or artifact in an

isolated, non-customer environment when possible.

4. The reviewer identifies affected components, versions, trust boundaries,

deployment assumptions, and credible impact.

5. Immediate mitigations are separated from the durable correction. Operators

may need a configuration restriction before a fixed artifact exists.

6. A correction is reviewed with the relevant negative tests, schemas,

conformance behavior, documentation, and release evidence.

7. Affected operators or upstream owners are coordinated when their action is

required for safe remediation.

8. The project determines whether to publish an advisory, release note,

mitigation notice, corrected artifact, or a combination of them.

The exact order may change to contain active harm. Receipt of a report does not confirm a vulnerability, severity, affected version, or remediation outcome.

Confidentiality and coordinated disclosure

The project limits report details to a need-to-know basis, but cannot promise anonymity or absolute confidentiality. Disclosure may be required for safe remediation, affected-operator coordination, upstream reporting, repository host action, or legal obligations.

The project has not published a fixed embargo period. Reporter and maintainers should coordinate a disclosure date based on exploitability, active abuse, availability of mitigations, affected operators, upstream dependencies,

and the time needed to distribute a verified correction.

A public advisory may describe:

- affected and unaffected versions or artifact digests;
- security impact and required attacker conditions;
- mitigations, fixed versions, and verification steps;
- relevant deployment assumptions and remaining limitations;
- reporter credit when the reporter requests it and disclosure is safe.

Do not treat silence or a delayed response as permission to publish secrets, customer data, or another party's protected information. A reporter may ask for a status update or propose a disclosure date through the original thread.

Responsible research expectations

When evaluating AIP security:

- use systems and accounts you own or are explicitly authorized to test;
- minimize access, requests, data retention, and provider-side effects;
- stop when testing reaches customer data, another tenant, a live credential,

or a risk of material service disruption;

- do not use social engineering, physical intrusion, denial of service,

destructive actions, persistence, or data exfiltration;

- keep exploit material private while remediation is being coordinated;
- follow applicable law and the terms governing the tested environment.

The selected source does not define a bug bounty, monetary reward, contractual safe harbor, CVE-assignment authority, or immunity from third-party terms. This policy does not create those promises.

Good-faith reporting is welcome even when the issue is incomplete, cannot be reproduced, or is ultimately classified as a non-security defect. Deliberately false, coercive, or retaliatory reporting remains subject to the code of conduct.

Maintainer responsibilities

Maintainers handling a report are responsible for:

- protecting report data and avoiding unnecessary reproduction on live data;
- distinguishing protocol, implementation, connector, upstream, artifact, and

deployment ownership;

- documenting the affected identity and evidence behind a security claim;
- ensuring that a mitigation does not silently weaken another trust boundary;
- updating tests, schemas, documentation, release notes, and artifacts that

would otherwise preserve the defect or an unsafe instruction;

- avoiding production-ready or fully secure claims that the evidence does not

establish;

- crediting a reporter only with their consent.

A passing source gate, vulnerability scan, signature, or conformance suite is one evidence layer. It does not prove that an arbitrary deployment, upstream service, identity system, or future configuration is secure.

Verify a remediated release

When a fix is published, identify and verify the exact corrected artifact. Do not rely only on a mutable image tag, local build ID, or version string.

The reviewed release workflow is designed to publish immutable multi-platform image digests with SBOMs, provenance, keyless Cosign signatures, and build attestations. The presence and verification of those artifacts must still be confirmed for the release that claims the correction.

Use the release note to identify the change, the artifact reference to verify what was published, and the security architecture to reassess the affected boundary.

Related documentation

- Security model
- Implementation status
- Release artifacts
- Contributing to AIP
- AIP code of conduct