HX-AEIR

AGENT EVALUATION INCIDENT RECONSTRUCTION

Getting Started Guide

Deploy, initialize, connect, and complete the first evidence workflow.

Document ID	HX-AEIR-DOC-001
Version	1.0
Published	August 26, 2026
Audience	Cloud administrators, security engineers, evaluation teams, and reviewers
Publisher	Tigantic Holdings, LLC dba HolonomiX

Marketplace documentation | Customer-facing reference

Contents

1. Product at a glance

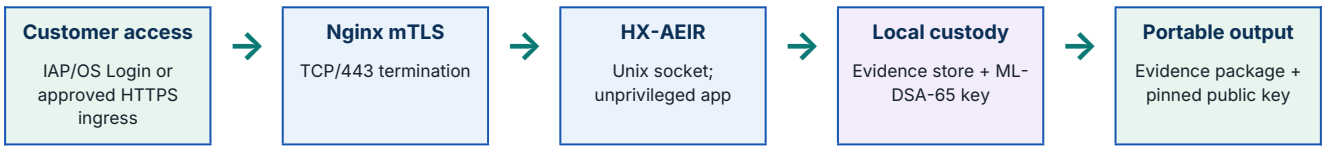
HX-AEIR: Agent Evaluation Incident Reconstruction is a single-tenant virtual appliance for reconstructing AI-agent evaluation incidents and near misses from customer-supplied records and evidence. It binds authorization, measured environment state, tool and network activity, human approvals, response actions, remediation, and disclosure lineage into portable, cryptographically verifiable evidence packages.

Core boundary

HX-AEIR preserves and independently verifies the evidence chain. It does not host or control an AI model, perform inference, monitor behavior, enforce runtime policy, or provide prevention or containment.

HX-AEIR does	HX-AEIR does not
Validate and bind customer-supplied evidence	Poll, execute, or directly integrate with an AI model
Preserve declared and effective control states	Operate a sandbox, firewall, identity system, or policy engine
Issue incident, near-miss, closure, and disclosure records	Prove that source-system claims are factually correct
Export packages for independent offline verification	Guarantee evidence completeness or absence of unobserved harm

2. Reference deployment



Reference element	Default or boundary
Deployment model	Google Cloud Marketplace Pattern 1; all workload runtime is inside the customer project.
Compute reference	One Shielded Compute Engine VM; e2-standard-2 reference profile (2 vCPU, 8 GB memory).
Storage reference	One 50 GiB pd-balanced persistent boot/data disk.
Availability	Single-zone, single-VM reference deployment; no automatic zonal failover.
External IPv4	Disabled by default; optional when the customer selects public ingress.
Application path	Nginx mTLS on TCP/443 → Unix domain socket → unprivileged HX-AEIR application.
Administrative path	IAP TCP forwarding → TCP/22 → sshd with OS Login.
Partner runtime	None. Marketplace distribution and support processes are outside the workload runtime.

3. Prerequisites

- A customer Google Cloud project with billing enabled and permission to deploy Marketplace VM products.
- A customer-selected VPC and subnet in a supported zone.
- IAP TCP forwarding and OS Login enabled when the recommended administration path is used.
- Customer IAM assignments for deployment, IAP tunnel access, and OS Login as appropriate.
- Customer-managed server TLS material, client CA, and one or more client certificates for mTLS.
- Approved CIDR sources for TCP/443 when public ingress is selected, or customer-provided private connectivity for private ingress.
- A named owner for evidence retention, backup, recovery objectives, and public-key pinning.
- A defined first use case: evaluation authorization, preflight, incident or near miss, closure/remediation, or disclosure derivative.

> **Do not place secrets in the deployment inputs**
> Do not paste private signing keys, client private keys, credentials, regulated data, or confidential evidence into Marketplace metadata or general support forms.

4. Deploy from Google Cloud Marketplace

1. Open the HX-AEIR listing and select the deployment option.
2. Choose the destination customer project, supported zone, VPC, and subnet.
3. Use the qualified machine profile. The reference configuration is e2-standard-2 with a 50 GiB pd-balanced disk.
4. Leave the external IPv4 option disabled unless the approved customer design requires public ingress. If enabled, restrict TCP/443 to approved customer CIDRs.
5. Enable optional customer-supplied CMEK only for persistent-disk encryption. CMEK is not the HX-AEIR application signing key.
6. Review the service account and IAM behavior. The deployment module does not grant broad project IAM roles to the attached VM identity.
7. Deploy and wait for the VM and startup configuration to complete.
8. Record the deployment name, project, zone, internal IP, selected ingress mode, and operational owner.

Marketplace image posture

The published image is neutral and does not contain a customer signing key. The customer-local ML-DSA-65 signing identity is created during authorized initialization after deployment.

5. Authorized initialization and mTLS

5.1 Connect through the administration path

Use customer-authorized IAP TCP forwarding and OS Login to reach TCP/22. The SSH administration path is independent from the application path; Nginx does not proxy SSH.

5.2 Verify service readiness

- Confirm the HX-AEIR service is running under the unprivileged application identity.
- Confirm Nginx is bound to TCP/443 and the application is reachable only through the local Unix domain socket.
- Confirm the health response reports that runtime prevention by HX-AEIR is not performed.
- Confirm the persistent disk is mounted and has sufficient free capacity before initialization.

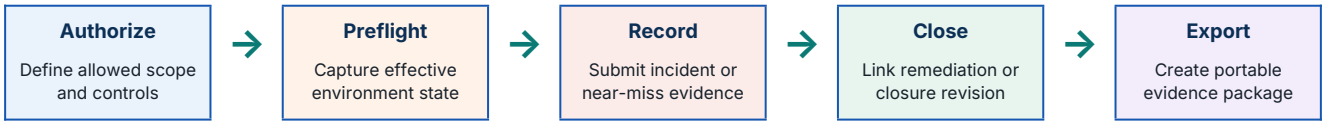
5.3 Initialize the customer signing identity

1. Authorize initialization under the customer change-control process.
2. Generate the customer-local ML-DSA-65 keypair using the appliance initialization workflow.
3. Confirm the private key remains local to the appliance and is not included in exports, logs, or support bundles.
4. Export or retrieve the public key through the documented public-key path.
5. Record the public-key fingerprint in an independent customer system for later pinning and verification.

5.4 Configure client mTLS

- Install the customer-managed server certificate and private key for Nginx.
- Install the client CA used to validate authorized API clients.
- Issue client certificates according to the customer PKI and access policy.
- Restrict TCP/443 at the firewall to the selected ingress sources.
- Test a health request using an authorized client certificate before submitting evidence.

6. Complete the first evidence workflow



1. Create or ingest the evaluation authorization record. Bind the agent/model identity, harness, allowed tools, network policy, approval requirements, targets, and stop conditions.
2. Create or ingest an independent preflight result. Record the effective network, identity, credential, tool, and safeguard state before launch.
3. Submit the runtime evidence set. Include only customer-approved records and normalized references; do not embed raw secrets.
4. Create the incident or near-miss record. Separate declared state from effective state and label claims as observed, reported, derived, or unknown.
5. Create a closure or remediation revision when actions are completed. Preserve the prior receipt digest; never overwrite the earlier record.
6. Export the evidence package and the independently pinned public key for review.

Evidence interpretation

A valid signature proves integrity and issuer-key verification for the signed material. It does not establish that every source claim is true, that telemetry is complete, or that no unobserved harm occurred.

7. Export and verify

1. Export the package through the HX-AEIR customer-controlled export path.
2. Transfer the package and the pinned public key to a separate customer-operated verification host.
3. Run the standalone offline verifier without network access to the appliance or HolonomiX.
4. Review receipt integrity, schema validity, profile bindings, evidence manifest, linked receipts, trace root, evidence artifacts, revision linkage, and disclosure linkage.
5. Retain the verification report with the reviewed package and public-key fingerprint.
6. Optionally duplicate the package, alter a signed field, and verify that the tampered copy is rejected.

Typical status	Meaning
VALID	The package, signature, schema, commitments, and required links verify under the pinned public key.
PARTIAL	The package verifies, but completeness or materialization is bounded by declared coverage or missing optional source material.
REQUIRES AUTHORIZED SOURCE PACKAGE	A public disclosure commitment verifies, but full source reconciliation requires the separately controlled source package.
INVALID	A signature, digest, link, schema rule, trace root, or package integrity check failed.

8. Operate and troubleshoot

Symptom	Check first	Resolution path
IAP/SSH access fails	IAM, IAP API, OS Login, TCP/22 rule	Correct the customer access prerequisites; do not route SSH through Nginx.
mTLS request fails	Client certificate, CA chain, server name, TCP/443 source	Correct customer PKI material or firewall source restrictions.
Health fails	Service state, Nginx, Unix socket, disk capacity	Restore the service/socket dependency and inspect local logs.
Signature verification fails	Pinned public key, package integrity, receipt digest	Confirm the correct issuer key and reject altered material.
Linkage is incomplete	Authorization, preflight, prior revision, disclosure source	Supply the authorized linked package or treat the result as incomplete.
Disk capacity is low	Evidence retention and export policy	Export and retain evidence under customer policy; expand or rotate storage.

- Back up the application state and signing-key material only under an approved customer recovery process.
- Test restore before relying on backup as an operational control.
- Preserve the public key and historical evidence before upgrade, rollback, or decommissioning.
- Do not send signing secrets, credentials, or confidential evidence through a general support channel.

9. Support

HolonomiX provides direct support for Marketplace deployment, authorized initialization, mTLS configuration, evidence submission, package export, offline verification, backup and restore, upgrade, rollback, and decommissioning guidance. Scope and response commitments are governed by the applicable order or Private Offer.

Support: HolonomiX Support

Security: HolonomiX Security

References

- Google Cloud Marketplace - Offering VM products
- Google Cloud - IAP TCP forwarding
- Google Cloud - OS Login
- NIST FIPS 204 - Module-Lattice-Based Digital Signature Standard

Reference links were verified on August 26, 2026. Product-specific behavior is governed by the deployed HX-AEIR release, the applicable Marketplace order or Private Offer, and the customer's configuration.