ARTICLE 12 - RECORD-KEEPING

Designing AI logs that produce verifiable evidence

A long-form technical and governance guide for high-risk AI system record-keeping, audit trails, and Decision Ledger evidence architecture.

This document is written for compliance, legal, security, and engineering teams that need more than application logs. It explains how to structure Article 12 record-keeping around automatic event capture, signed decision records, artifact provenance, and independent verification.

Primary audience

AI governance leads, compliance officers, counsel, auditors, and engineering owners.

Recommended use

Forward as an internal review packet, attach to procurement responses, or use as a blueprint for logging requirements.

Evidence primitives

Signed decision records, canonical JSON payloads, Ed25519 public key verification, SHA-256 artifact fingerprints.

Scope boundary

Engineering reference only. It supports compliance readiness but is not a legal opinion.

Source page	certifieddata.io/eu-ai-act/article-12-record-keeping
PDF path	/eu-ai-act/article-12-record-keeping.pdf
Version	1.0 - generated from long-form page content
Updated	May 2026

Record-keeping

Decision Ledger

Artifact provenance

Independent verification

EXECUTIVE SUMMARY

Article 12 is not just a logging requirement. It is an evidence requirement.

For high-risk AI systems, logs must support traceability over the lifetime of the system. In practice, that means the organization needs to answer who or what produced a decision, which version of the system was used, what data or reference source was involved, what outcome was returned, and whether the record can still be trusted after the fact.

Traditional application logs are useful for operations, but they are rarely sufficient for governance review. They may be mutable, distributed across systems, hard to correlate to a specific AI decision, or dependent on privileged access to the production environment. A compliance-grade record-keeping architecture should produce evidence that can be exported, inspected, and independently verified.

Recommended framing for compliance buyers

Use the public long-form page as the source of truth for SEO, cross-links, and product education. Use this branded PDF as the forwardable internal artifact for legal, compliance, security, and procurement review. The PDF should be regenerated when the page copy changes.

NEED	WEAK PATTERN	BETTER ARTICLE 12 EVIDENCE PATTERN
Traceability	Generic logs with timestamps and text strings.	Structured decision records with actor, entity, output, reason codes, artifact references, and policy context.
Integrity	Database rows that privileged users can edit silently.	Canonical payloads hashed with SHA-256 and signed with Ed25519, with key IDs and public verification paths.
Reviewability	Only engineers can reconstruct what happened.	Compliance officers and auditors can inspect exported records without production access.
Lifecycle linkage	Runtime logs disconnected from training data and model artifacts.	Decision records link back to certified datasets, model artifacts, reference data, prompts, or generated outputs.

CONTENTS

What this guide covers

1. ARTICLE 12 IN PLAIN LANGUAGE

High-risk AI systems should automatically record events over their lifetime.

Article 12 of the EU AI Act focuses on record-keeping for high-risk AI systems. The core engineering obligation is that the system must technically allow automatic recording of events, often referred to as logs, throughout the system lifetime. The logging capability should support traceability appropriate to the intended purpose of the system.

Orientation excerpt

High-risk AI systems shall technically allow for the automatic recording of events over the lifetime of the system. Logging capabilities should ensure a level of traceability of the system's functioning that is appropriate to the intended purpose. Consult the official Regulation text and counsel for authoritative interpretation.

For implementation teams, the important word is not merely "logs." The important outcome is traceability. A record-keeping design should help a reviewer understand the functioning of the AI system when a decision, recommendation, classification, agent action, or materially relevant model output is challenged later.

Questions an Article 12-ready log should help answer

- Which AI system, model, agent, or workflow produced the output?
- When was the system operating, and which version or configuration was active?
- Which input, reference database, dataset, prompt, tool call, or artifact was used?
- What result was returned, and what human-readable or machine-readable reason was recorded?
- Was the event routine, exceptional, overridden, escalated, or security-relevant?

- Can an auditor detect if the record was altered after issuance?
- Can the record be verified without trusting the application that created it?

Important boundary

CertifiedData primitives can contribute evidence integrity, artifact provenance, and signed decision logging. They do not replace risk management, legal classification, fundamental rights review, model evaluation, cybersecurity controls, access governance, or retention policy decisions.

2. EVIDENCE MODEL

Start by defining the unit of record.

An Article 12 logging architecture should not begin with a storage technology. It should begin with the unit of record: the smallest event that must be reviewable later. For many systems, that unit is not every model inference. It is every high-risk decision, recommendation, action, override, escalation, or lifecycle event that can materially affect a person, organization, asset, or regulated process.

Runtime decision records

Capture the AI action or decision event itself: actor, timestamp, subject, output, reason codes, rationale summary, model or workflow version, policy context, and correlation IDs.

Artifact lifecycle records

Capture the provenance of datasets, model artifacts, prompts, reference databases, generated outputs, and supporting documentation used by the AI system.

Suggested event taxonomy

EVENT CLASS	EXAMPLES	WHY IT MATTERS
Operating period	Service start, service stop, deployment window, batch job run.	Shows when the system was active and which operating context applied.
Decision or output	Approval, denial, ranking, recommendation, risk score, agent action.	Links system functioning to the user-visible or business-visible effect.
Reference data access	Use of a reference database, ruleset, embedding index, dataset, prompt library.	Supports traceability between output and data context.
Artifact version	Model version, dataset fingerprint, configuration hash, prompt hash.	Prevents ambiguity about what actually produced the event.
Human intervention	Override, escalation, approval, rejection, review note.	Shows where human oversight affected the outcome.
Exception or risk signal	Out-of-distribution flag, confidence threshold breach, policy failure, security event.	Helps identify situations that may present risk or require substantial modification.

The strongest evidence model connects runtime decisions to artifact provenance. A decision record should be able to reference a certified dataset, a model artifact hash, a policy version, a prompt fingerprint, a reference database snapshot, or a generated output certificate. This is how record-keeping becomes lineage rather than a pile of logs.

3. DECISION LEDGER SCHEMA

A record should be structured before it is signed.

Machine-verifiable record-keeping depends on a stable payload shape. The exact schema will vary by domain, but the record should separate identity, event context, decision content, references, integrity metadata, and verification metadata.

```
{
  "record_id": "dec_01hv...",
  "timestamp": "2026-05-04T12:00:00Z",
  "actor": {
    "type": "ai_system",
    "system_id": "credit-risk-workflow",
    "model_version": "risk-model-2026-04-18"
  },
  "entity": {
    "type": "application",
    "correlation_id": "case_7f32..."
  },
  "decision": {
    "outcome": "manual_review_required",
    "reason_codes": ["income_variance", "thin_file"],
    "rationale_summary": "Application requires human review because two risk signals exceeded
policy thresholds."
  },
  "references": {
    "dataset_hash": "sha256:...",
    "model_artifact_hash": "sha256:...",
    "policy_version": "underwriting-policy-2026-03",
    "certificate_url": "https://certifieddata.io/verify/..."
  },
  "integrity": {
    "payload_hash": "sha256:...",
    "previous_record_hash": "sha256:...",
    "canonicalization": "RFC8785-JCS",
    "signature_algorithm": "Ed25519",
    "key_id": "cd-key-2026-01",
    "signature": "base64url..."
  }
}
```

FIELD GROUP	PURPOSE	COMPLIANCE VALUE
`record_id`	Stable identifier for the event.	Allows audit references, exports, appeals, and internal case review.
`timestamp`	ISO-8601 event time.	Supports chronological reconstruction of system functioning.
`actor`	System, model, agent, service, or human actor involved.	Links the event to the responsible technical component.
`entity`	Subject of the decision, using a privacy-preserving correlation ID where possible.	Supports traceability without overexposing personal data in governance exports.
`decision`	Outcome, reason codes, and rationale summary.	Transforms raw logging into reviewable decision evidence.
`references`	Dataset, model, policy, reference database, prompt, or certificate references.	Connects runtime events to artifact lineage.
`integrity`	Hash, canonicalization method, signature algorithm, key ID, and signature.	Allows independent verification and tamper detection.

4. CERTIFIEDDATA PRIMITIVE MAPPING

CertifiedData contributes verifiable evidence, not a legal shortcut.

The CertifiedData stack is useful for Article 12 because it separates three concerns that are often mixed together: the decision record, the artifact provenance record, and the independent verification surface.

1

Signed decision record

Decision Ledger captures a high-risk AI decision or lifecycle event as a structured payload and signs it.

2

Certified artifact reference

Datasets, model artifacts, AI outputs, or manifests can be fingerprinted and certified separately.

3

Public verification path

Auditors can verify hashes, signatures, and key IDs without depending on dashboard access.

How the primitives fit together

CERTIFIEDDATA PRIMITIVE	WHAT IT PROVIDES	ARTICLE 12 CONTRIBUTION
decision-record	A signed, machine-readable record of a decision or event, including actor, outcome, explanation, entity, timestamp, and integrity metadata.	Directly supports traceability of system functioning and review of specific decisions.
signed-certificate	Ed25519-signed certificate over an artifact's SHA-256 fingerprint, algorithm specification, and metadata.	Lets logs reference the exact dataset, model artifact, prompt, manifest, or AI output involved.
canonicalization	Deterministic JSON canonicalization, such as RFC 8785 JCS, before signing.	Lets third parties reproduce the exact signed bytes and avoid signature-equivalent ambiguity.
signing-key-registry	Public Ed25519 verification keys, including retired keys where needed for historical verification.	Supports independent verification even after key rotation.
public-registry	Public or controlled verification URLs for artifacts and selected evidence records.	Gives reviewers a stable path to verify provenance and integrity without production access.
manifest-certification	Batch certification for groups of related artifacts from a structured manifest.	Supports CI/CD and AI Bill of Materials workflows where multiple artifacts must be linked.

Practical result

An Article 12 record can say: this AI system produced this decision at this time, under this policy and model version, using these certified artifacts, and the record has not been silently altered since issuance.

5. PROVIDER AND DEPLOYER RESPONSIBILITIES

Logging architecture must be shared, but responsibilities are not identical.

Article 12 readiness usually spans provider-side technical design and deployer-side operational control. CertifiedData can help both sides by creating evidence that travels across organizational boundaries: a provider can certify artifacts and expose verification paths; a deployer can log runtime decisions and retain the records under its own controls.

Provider evidence package

- Document the system's logging capability and event taxonomy.
- Certify training datasets, synthetic datasets, model artifacts, manifests, or output sets where relevant.
- Publish or share verification keys and certificate verification instructions.
- Provide deployers with integration guidance for logging runtime events.

Deployer operating controls

- Decide which high-risk decisions and lifecycle events must be logged.
- Configure retention, access control, and export procedures.
- Link records to user-visible actions through correlation IDs.
- Maintain review procedures for overrides, incidents, appeals, and authority requests.

Neither side should treat record-keeping as an afterthought. If logging is added only after a regulator, auditor, customer, or impacted person asks a question, the organization is forced to reconstruct evidence from partial logs. Article 12-ready design captures evidence as part of normal system operation.

6. IMPLEMENTATION CHECKLIST

Move from requirement to operating evidence.

WORKSTREAM	CHECKLIST	EVIDENCE OUTPUT
System classification	Confirm whether the workflow is a high-risk AI system, a component, or an adjacent governance process. Capture legal basis and owner.	Classification memo, risk owner, system inventory entry.
Event taxonomy	Define decision events, lifecycle events, exception events, and reference-data events. Decide what does not need a signed record.	Article 12 event taxonomy and logging policy.
Record schema	Define required fields, optional fields, redaction rules, pseudonymous identifiers, and correlation IDs.	Versioned schema and sample records.
Integrity model	Choose canonicalization, hashing, signature algorithm, key IDs, key rotation policy, and verification URL format.	Verification specification and key registry.
Artifact linkage	Link decisions to certified datasets, model artifacts, prompt fingerprints, policy versions, and generated outputs where relevant.	Artifact registry references and certificate IDs.
Retention and access	Set retention windows, storage controls, export procedures, and access logging for the logs themselves.	Retention schedule, access policy, export runbook.
Audit workflow	Define who can request records, how verification is performed, how exceptions are reviewed, and how counsel is involved.	Audit response playbook and review worksheet.

Verification procedure

1. Export the decision record payload and integrity metadata.
2. Canonicalize the payload using the stated canonicalization scheme.
3. Compute the SHA-256 payload hash and compare it with the recorded hash.
4. Fetch the public Ed25519 verification key from the key registry using the record's key ID.
5. Verify the signature over the canonical payload.
6. Resolve any referenced certificate URLs and verify artifact hashes separately.
7. Document the verification result, reviewer, date, and any exceptions.

7. COMMON FAILURE MODES

What makes AI logs weak in review?

FAILURE MODE	WHY IT FAILS	MITIGATION
Mutable operational logs only	Privileged users can alter or delete records without a clear integrity signal.	Sign decision records and preserve hash-linked exports.
No artifact references	The log says a model acted, but not which model, dataset, policy, or reference source was used.	Link each decision to artifact fingerprints and certificate IDs.
Human-readable text only	Auditors cannot reliably query, compare, or automate verification.	Use machine-readable reason codes alongside rationale summaries.
No key rotation plan	Historical records become difficult to verify after signing keys change.	Maintain a public or controlled key registry with retired keys retained for verification.
Overcollection of personal data	Logs create privacy risk and make export harder.	Use pseudonymous entity IDs, references, and minimization rules.
Compliance claims without controls	A tool is presented as full compliance, but policy, retention, and governance remain undefined.	State what the evidence layer contributes and what remains the organization's responsibility.

8. INTERNAL REVIEW WORKSHEET

Use this worksheet before implementation sign-off.

Article 12 evidence scope

AI system / workflow	
Business owner	
Technical owner	
Compliance / legal reviewer	
High-risk classification basis	
Decision events to log	
Artifacts to certify or reference	
Retention period and storage owner	
Verification procedure approved?	Yes / No / Exceptions

Governance reviewer signature

Date

Engineering owner signature

Date

Build the evidence layer before the first review request arrives.

CertifiedData helps teams connect AI decision records, signed artifact certificates, public verification keys, and registry references into an audit trail that can be checked independently.

This is not a promise of automatic legal compliance. It is a practical way to make AI record-keeping more traceable, tamper-evident, and reviewable.

View Decision Ledger | Verify a record | Review the public registry